

GESELLSCHAFT
FÜR INFORMATIK



Arno Pasternak (Hrsg.)

Informatik für alle
18. GI-Fachtagung Informatik und Schule

16. - 18. September 2019
Dortmund

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) - Proceedings

Series of the Gesellschaft für Informatik (GI)

Volume P-288

ISBN 978-3-88579-682-4

ISSN 1617-5468

Volume Editors

Dr. Arno Pasternak

TU Dortmund

Otto-Hahn-Str. 14

D-44227 Dortmund

Germany

Arno.Pasternak@tu-dortmund.de

Series Editorial Board

Heinrich C. Mayr, Alpen-Adria-Universität Klagenfurt, Austria
(Chairman, mayr@ifit.uni-klu.ac.at)

Torsten Brinda, Universität Duisburg-Essen, Germany

Dieter Fellner, Technische Universität Darmstadt, Germany

Ulrich Flegel, Infineon, Germany

Ulrich Frank, Universität Duisburg-Essen, Germany

Michael Goedicke, Universität Duisburg-Essen, Germany

Ralf Hofestädt, Universität Bielefeld, Germany

Wolfgang Karl, KIT Karlsruhe, Germany

Michael Koch, Universität der Bundeswehr München, Germany

Thomas Roth-Berghofer, University of West London, Great Britain

Peter Sanders, Karlsruher Institut für Technologie (KIT), Germany

Andreas Thor, HFT Leipzig, Germany

Ingo Timm, Universität Trier, Germany

Karin Vosseberg, Hochschule Bremerhaven, Germany

Maria Wimmer, Universität Koblenz-Landau, Germany

Dissertations

Steffen Hölldobler, Technische Universität Dresden, Germany

Thematics

Andreas Oberweis, Karlsruher Institut für Technologie (KIT), Germany

© Gesellschaft für Informatik, Bonn 2019

printed by Köllen Druck+Verlag GmbH, Bonn



This book is licensed under a Creative Commons BY-SA 4.0 licence.

Vorwort

Seit inzwischen über 50 Jahren gibt es Informatik-Angebote in den Schulen. Aus den ersten Arbeitsgemeinschaften und Projekten erwuchs in den 70-Jahren das Schulfach Informatik. Mit der Reform der gymnasialen Oberstufe im Jahre 1972 wurde es in den Rang eines ‚normalen‘ Schulfaches gehoben, das von den Schülerinnen und Schülern in der gymnasialen Oberstufe gewählt werden konnte [KM72].

Folglich entwickelte sich die didaktische Forschung in der Schulinformatik, Schulbücher wurden erstellt, Lehrer ausgebildet und weiterqualifiziert. Zusätzlich entstanden in der Sekundarstufe I Lehrangebote von Arbeitsgemeinschaften bis hin zu Wahlpflichtangeboten. Eigentlich stellte sich nicht die Frage, ob Informatik in der Sekundarstufe I Schulfach werden sollte oder nicht, sondern nur wann und wie. Eine *Grundbildung Informatik* für alle Schülerinnen und Schüler war ein wichtiges bildungspolitisches Thema [Bo85, fPuTdS86, Ku90, vP86, Bi90].

Ausdruck dieser intensiven bildungspolitischen Diskussion um das Schulfach Informatik war u.A. der Start der Konferenzreihe *Informatik und Schule* mit der ersten Tagung vom 8. bis zum 10. Oktober 1984 in Berlin vor fast genau 35 Jahren. Dort formulierte *Peter Heyderhoff* in einem Grundsatzartikel *Didaktik der Schulinformatik* den Anspruch auf eine *Informatik für alle*:

„Ein Schüler, der nach 10 Schuljahren in die Berufsausbildung entlassen wird, muß verstehen, wie eine Problemlösung in Einzelschritten erarbeitet wird und welche intellektuellen Fähigkeiten des Menschen durch Rechner sinnvoll unterstützt werden können. . . . In seiner Berufssituation als Betroffener und als Anwender muß er in fachlicher und sozialer Kompetenz zum Einsatz technischer Medien Stellung beziehen können [He84, S.66].“

Er schliesst seine Analyse mit:

„Die Herausforderung durch die Informationstechnik führt zu der Notwendigkeit, Informatikunterricht in allen Bildungsbereichen einzuführen. Dies ist mittlerweile eine unumstrittene und allgemein akzeptierte Erkenntnis und muß konsequenterweise dazu führen, daß Informatikunterricht sehr bald von einem Spezialgebiet und Wahlfach zu einem festen Bestandteil des Pflichtfachbereichs in der allgemeinbildenden Schule wird [He84, S.71].“

Diese schon 1984 berechnigte Forderung nach einer *Informatik für alle* ist in Deutschland bis heute nicht realisiert.

In der Euphorie der Anfangsjahre der Schulinformatik und der ersten INFOS-Konferenz ist vermutlich allen Beteiligten entgangen, dass diese Position nicht von allen im Bildungsbereich Tätigen geteilt worden sind. Während fast alle Informatikdidaktiker in der Vorbereitung und Teilnahme der ersten INFOS involviert waren, wurde fast zur gleichen Zeit vom 6. bis zum 8. September 1984 ziemlich genau 500 km entfernt an der Universität Dortmund eine Konferenz *Neue Medien und Lernen* durchgeführt, an der eine Vielzahl von Personen aus dem Bildungsbereich — allerdings bis auf Ausnahmen keine Informatiklehrer — teilnahmen.

Am letzten Tag dieser Veranstaltung wurde in einer Plenumsitzung nach einer Diskussionsrunde auf einem Podium mit ca. zehn Personen aus vielen Bereichen des Bildungswesens eine (vorbereitete) Resolution von den Teilnehmern fast einstimmig verabschiedet. Auf diesem Podium waren von der Gewerkschaft GEW bis zum Verband der Lateinlehrer unterschiedlichste Positionen vertreten, nur ein Vertreter aus dem Bereich der Informatik (Lehrer) fehlte. So konnte auch ein damals recht junger Informatiklehrer von einer Gesamtschule aus Hagen nicht verhindern, dass gegen seine (offensichtlich nicht gewünschten und als Störung der grossen sonst so einmütigen Gemeinsamkeit empfundenen) Einwände folgender Grundsatz im Rahmen dieser Erklärung festgelegt wurde:

„Eine ‚informationstechnologische Grundbildung‘ sollte am Ende der Sekundarstufe I in Form von Blockphasen eingerichtet werden. Diese Grundbildung soll jedoch nicht in Form eines eigenständigen Fachs, sondern projektförmig in Verbindung mit Naturwissenschaften, Sozialwissenschaften und Sprachen unterrichtet werden. [RZ85, S.198]“

Allerdings wurde auch festgestellt:

„Wenn es zutrifft, daß die Neuen Medien Universalmedien sind, die auf Informatik (engl. computer science) als Grundlagenwissenschaft beruhen, dann erhält die Notwendigkeit einer wissenschaftspropädeutischen Grundbildung für alle neue unabweisbare Begründung. [RZ85, S.199]“

Diese Erkenntnis wurde aber sofort wieder eingeschränkt:

„Diese wissenschaftspropädeutische Grundbildung dürfte sich allerdings nicht darauf beschränken, in die Grundlagen der Informatik einzuführen. [RZ85, S.199]“

Selten hat eine Konferenz die praktische Bildungspolitik beeinflusst wie diese. Ein durchgängiges Fach Informatik in der Sekundarstufe I gibt es in Deutschland landesweit bis heute nicht. Zwangsläufig kommt daher in mehr oder weniger unregelmässigen Abständen die Diskussion um ein Schulfach Informatik erneut auf. So wird auch derzeit im Rahmen der aktuellen Diskussion um die *Digitalisierung der Schule* erneut das Schulfach Informatik in der Sekundarstufe I von vielen gefordert. Diese Forderungen werden inzwischen durch die *GI Bildungsstandards Informatik* für den Primarbereich [Ar] und die Sekundarstufen I und II inhaltlich gestützt und beschrieben [Ar08, Ar16].

Es ist und tut gut, dass die Teilnehmer auf der 18. *GI-Fachtagung Informatik und Schule* wieder hier in Dortmund die Möglichkeit haben, durch ihre Diskussionen und Beratungen als Grundlage weiterer Forschungen in den Hochschulen sowie Unterrichten in den Schulen als auch Planungen im administrativen Bereich die Voraussetzungen dafür zu schaffen, den grandiosen Fehler der Tagung von vor 35 Jahren zu korrigieren.

Wünschen wir daher allen Schülerinnen und Schülern der Gegenwart und Zukunft, dass es keine weiteren 35 Jahre mehr dauert, bis

Informatik für alle

eine gelebte Selbstverständlichkeit ist.

INFOS 2019

In diesem Band zur Tagung *INFOS 2019 — Informatik für alle* befinden sich die angenommenen wissenschaftlichen Beiträge und Praxisberichte sowie die Beschreibungen der auf der Tagung durchgeführten Workshops und präsentierten Poster.

Das Programmkomitee hat aus 38 eingereichten wissenschaftlichen Beiträgen 12 ausgewählt. Dies entspricht einer Annahmequote von 32%. Von 33 eingereichten Praxisberichten wurden 19 angenommen. Dies entspricht einer Annahmequote von 58%. Damit sind von insgesamt 71 Einreichungen 31 Beiträge — also 44% — angenommen worden. Ich danke allen Autorinnen und Autoren für ihre Mitarbeit an der Konferenz und allen beteiligten Gutachterinnen und Gutachtern für den sehr konstruktiven Review-Prozess.

Ich hoffe, dass die Vorträge und Workshops auf der Konferenz für alle eine Bereicherung darstellen. Es bleibt zu hoffen, dass diese Tagung für die Teilnehmer und darüber hinaus vor allem für alle Lehrerinnen und Lehrer einen positiven Einfluss auf den Unterricht hat und wir entsprechend dem Konferenzmotto bald *Informatik für alle* als Normalität feststellen können.

Dortmund, im September 2019
Arno Pasternak

Literaturverzeichnis

- [Ar] Arbeitskreis 'Bildungsstandards Informatik im Primarbereich' der GI: Kompetenzen für informatische Bildung im Primarbereich. LOGIN-Verlag, Berlin.
- [Ar08] Arbeitskreis 'Bildungsstandards' der GI: Grundsätze und Standards für die Informatik in der Schule. LOGIN-Verlag, Berlin, 2008.
- [Ar16] Arbeitskreis 'Bildungsstandards SII' der GI: Bildungsstandards Informatik für die Sekundarstufe II. LOGIN-Verlag, Berlin, 2016.
- [Bi90] Bikos, Konstantinos: Informationstechnologische Grundbildung als Curriculum-Problem der Achtziger Jahre. Europäische Hochschulschriften: Pädagogik. Peter Lang, Frankfurt am Main, 1990.
- [Bo85] Bosler, Ulrich; Hampe, Wolfgang; Wanke, Ilona; van Weert, Tom J.: Grundbildung Informatik. J.B. Metzlersche Verlagsbuchhandlung, Stuttgart, 1985.
- [fPuTdS86] für Praxis und Theorie der Schule, Landesinstitut Schleswig-Holstein: Informationstechnische Grundbildung - Band I. Aulis Verlag Deubner & Co KG, Köln, Materialien für den Mathematikunterricht in allgemeinbildenden Schulen des Landes Schleswig-Holstein (Klassenstufe 8). Auflage, 1986.
- [He84] Heyderhoff, Peter: Didaktik der Schulinformatik. In (Arlt, Wolfgang, Hrsg.): Informatik als Herausforderung an Schule und Ausbildung. Jgg. 1, R. Oldenbourg Verlag, München, 1984.
- [KM72] KMK (Ständige Konferenz der Kultusminister der Länder der Bundesrepublik Deutschland): Vereinbarung zur Gestaltung der gymnasialen Oberstufe in der Sekundarstufe II. 1972.
- [Ku90] Kultusminister, NRW: Vorläufige Richtlinien zur Informations- und Kommunikationstechnologischen Grundbildung in der Sekundarstufe I, Jgg. 5051. Ritterbach, Frechen, 1990. Schriftenreihe des Kultusministers.
- [RZ85] Rolf, Hans-Günter; Zimmermann, Peter: Neue Medien und Lernen. Beltz Verlag, Weinheim, 1985.
- [vP86] von Puttkamer, Ewald, Hrsg. Informatik-Grundbildung in Schule und Beruf, GI-Fachtagung, Kaiserslautern, 29. September - 1. Oktober 1986, Proceedings, Jgg. 129 in Informatik-Fachberichte. Springer, 1986.

Tagungsleitung

Gesamtleitung: Arno Pasternak, TU Dortmund
Doktorandensymposium: Martin Weinert, TU Dortmund

Programmkomitee

Leitung des Programmkomitees:

Arno Pasternak (Vorsitz)
Peer Stechert (stellvertretender Vorsitz)

Christian Borowski	Universität Oldenburg
Michael Brinkmeier	Universität Osnabrück
Katrin Büttner	Goethe-Mittelschule Heidenau
Dino Capovilla	Humboldt-Universität zu Berlin
Beat Döbeli Honegger	Pädagogische Hochschule Schwyz
Ira Diethelm	Carl von Ossietzky Universität Oldenburg
Rebecca Doherty	Landrat-Lucas-Gymnasium Leverkusen
Dieter Engbring	Universität Bonn
Johannes Fischer	TU Dortmund
Jens Gallenbacher	TU Darmstadt
Kathrin Haselmeier	Universität Wuppertal
Werner Hartmann	Pädagogische Hochschule Bern
Lutz Hellmig	Universität Rostock
Tino Hempel	Richard-Wossidlo-Gymnasium
Henry Herper	Universität Magdeburg
Sven Hofmann	Universität Leipzig
Alexander Hug	Universität Koblenz-Landau
Ludger Humbert	Willy-Brandt-Gesamtschule Bergkamen
Lennard Kerber	Otto-Nagel-Gymnasium Berlin
Maria Knobelsdorf	Universität Wien
Martin Lehmann	Pädagogische Hochschule Bern
Peter Micheuz	Alpen-Adria-Universität Klagenfurt
Andreas Muehling	Universität Kiel
Dorothee Müller	Bergische Universität
Simone Opel	Universität Paderborn
Arno Pasternak	TU Dortmund
Johannes Pieper	Joseph-König-Gymnasium Haltern
Wolfgang Pohl	Bundesweite Informatikwettbewerbe
Mareen Przybylla	Universität Potsdam
Hermann Puhlmann	Leibniz-Gymnasium Altdorf
Gerhard Röhrer	Studienseminar für Gymnasien Darmstadt
Ralf Romeike	Freie Universität Berlin
Carsten Schulte	Universität Paderborn
Andreas Schwill	Universität Potsdam
Peer Stechert	RBZ Technik Kiel
Kerstin Strecker	Universität Göttingen
Renate Thies	Cusanus Gymnasium Erkelenz
Marco Thomas	Westfälische Wilhelms-Universität Münster
Michael Weigend	Westfälische Wilhelms-Universität Münster

Organisationsteam

Kensuke Akao	TU Dortmund
Frank Thorsten Breuer	TU Dortmund
Stefan Dissmann	TU Dortmund
Johannes Fischer	TU Dortmund
Gundel Jankord	TU Dortmund
Florian Kurpicz	TU Dortmund
Alfred Schenk	TU Dortmund
Martin Weinert	TU Dortmund

Joachim Deckers (Registrierungssystem)	GI FG NRW
Tobias Brinkjost (Webseite)	TU Dortmund

Lektorat

Eberhard Kranert	Fritz-Steinhoff-Gesamtschule Hagen
Jörg Kuhlmann	Essen
Torben Leopold	Fritz-Steinhoff-Gesamtschule Hagen
Uwe Marnowsky	Fritz-Steinhoff-Gesamtschule Hagen
Irene Pasternak	Essen

Proceedings-Support

Anatolij Fandrich	Carl von Ossietzky Universität Oldenburg
Nils Pancratz	Carl von Ossietzky Universität Oldenburg

Sponsor

Wir danken der Firma **Google** für die Finanzierung dieses Bandes und die weitere Unterstützung der Konferenz.

Inhaltsverzeichnis

Eingeladene Vorträge

Oliver Kaczmarek <i>Lernen in der digitalen Welt</i>	21
Hans Werner Heymann <i>Allgemeinbildung und Informatik</i>	
Ira Diethelm <i>Von Dagstuhl nach Frankfurt – auf dem Weg zu einem interdisziplinären Modell digitaler Bildung</i>	23
Torsten Brinda, Niels Brügger, Ira Diethelm, Thomas Knaus, Sven Kommer, Christine Kopf, Petra Missomelius, Rainer Leschke, Friederike Tilemann, Andreas Weich <i>Frankfurt-Dreieck zur Bildung in der digital vernetzten Welt - Ein interdisziplinäres Modell</i>	25
Dino Capovilla <i>Informatische Bildung und inklusive Pädagogik</i>	35

Wissenschaftliche Beiträge

Kevin Baum, Nadine Kirsch, Kerstin Reese, Pascal Schmidt, Lukas Wachter, Verena Wolf <i>Informatikunterricht in der Grundschule? - Erprobung und Auswertung eines Unterrichtsmoduls mit Calliope mini</i>	49
Alexander Best <i>Bild der Informatik von Grundschullehrpersonen: Ergebnisse eines mehrjährigen Projekts zu informatikbezogenen Vorstellungen</i>	59

Ira Diethelm, Sebastian Glücks <i>Analyse von Curricula auf Abdeckung der Kompetenzen zur Bildung in der digitalen Welt</i>	69
Alexander Hacke, Mareen Przybylla, Andreas Schwill <i>Beobachtungen zum inform. Problemlösen im Escape-Adventure-Spiel „Room-X“</i>	79
Kathrin Haselmeier <i>Informatik an Grundschulen – Stellschraube Lehrerbildung</i>	89
Kathrin Haselmeier, Ludger Humbert, Klaus Killich, Dorothee Müller <i>Interesse an Informatik und Informatikselbstkonzept bei Schülerinnen und Schülern zu Beginn der fünften Jahrgangsstufe des Gymnasiums - Eine empirische Untersuchung</i>	99
Niko Hausner, Matthias Wendlandt, Katharina Wendlandt <i>Informatikunterricht – Ein Muss zur politischen Mündigkeit</i>	109
Daniel Losch, Ludger Humbert <i>Informatische Bildung für alle Lehramtsstudierenden – Reformprozess einer allgemeinbildenden Informatikveranstaltung in der universitären Lehrerbildung</i>	119
Tilman Michaeli, Ralf Romeike <i>Debuggen im Unterricht - Ein systematisches Vorgehen macht den Unterschied</i>	129
Kathrin Müller, Carsten Schulte, Johannes Magenheim <i>Zur Relevanz eines Prozessbereiches Interaktion und Exploration im Kontext informatischer Bildung im Primarbereich</i>	139
Ilona Petrenko, Marco Thomas <i>Entwicklung ökologischen Denkens im Informatikunterricht</i>	149

Praxisberichte

Marc Berges, Matthias Ehmman, Martin Hennecke, Ute Heuer, Annabel Lindner, Rainer Gall, André Greubel, Nicole Günzel-Weinkamm, Verena Haller, Julia Kronawitter, Nicolai Pöhner <i>Erfahrungsbericht zur Qualifizierungsmaßnahme Informatik als Erweiterungsfach (Lehramt Realschule) in Bayern</i>	161
Dieter Engbring <i>„Bildung in der digital vernetzten Welt“ Ohne Informatik nicht denkbar!?</i>	171
Katharina Geldreich, Mike Talbot, Peter Hubwieser <i>Aufgabe ist nicht gleich Aufgabe – Vielfältige Aufgabentypen bewusst in Scratch einsetzen</i>	181
Lisa Göbel, Lutz Hellmig <i>Informatik spannend präsentieren – Ein Einstieg in den Studiengang Lehramt Informatik</i>	191
Andrea Gumpert, Pascal Zaugg <i>Fit für den Lehrplan 21 – Wie Klassenlehrpersonen auf den Informatikunterricht vorbereitet werden (können)</i>	201
Michael Hielscher, Christian Wagenknecht <i>FLACI – Eine Lernumgebung für theoretische Informatik</i>	211
Tamara Hochthurn, Jens Gallenbacher <i>Innovative Gestaltung und Einsatz von Erklärvideos im Informatikunterricht</i>	221
Beat Döbeli Honegger, Martin Hermida, Regina Schmid <i>Zur Entwicklung des Masterstudiengangs "Medien und Informatik"</i>	231
Ludger Humbert, Henry Herper, Alexander Best, Christian Borowski, Rita Freudenberg, Martin Fricke, Kathrin Haselmeier, Henry Herper, Volkmar Hinz, Dorothee Müller, Andreas Schwill, Marco Thomas <i>Empfehlungen der GI – Kompetenzen für informatische Bildung im Primarbereich</i>	237
Florian Kapp, Carsten Schulte <i>Einsatz von Jupyter Notebooks am Beispiel eines fiktiven Kriminalfalls . .</i>	247

Andreas Koch, Alexander Mittag <i>Informatikunterricht in Baden-Württemberg</i>	257
Urs Meier <i>TPA-Modell</i>	265
Peter Micheuz <i>Mischen possible. Didaktische und unterrichtspraktische Überlegungen zum Mischen im Informatikunterricht</i>	275
Simone Opel, Michael Schlichtig, Carsten Schulte, Rolf Biehler, Daniel Frischemeier, Susanne Podworny, Thomas Wassong <i>Entwicklung und Reflexion einer Unterrichtssequenz zum Maschinellen Lernen als Aspekt von Data Science in der Sekundarstufe II</i>	285
Nils Pancratz, Anatolij Fandrich, Christos Chytas, Mareike Daeglau, Ira Diethelm <i>Blöcke, Blumen, Mikrocontroller und das Internet of Things – Ein Konzept zum kontextorientierten Einsatz von Parametric Design und Physical Computing in Mädchen-Workshops</i>	295
Thomas Schmalfeldt <i>Einsatz von Skill Cards und Story Cards für einen kreativitätsfördernden Informatikunterricht auf der Sekundarstufe I</i>	305
Ann-Katrin Schmidt, Carsten Schulte <i>Das RetiBNE Café - Ein Konzept für Repair Cafés in der informatischen Bildung</i>	315
Stefan Seegerer, Annabel Lindner, Ralf Romeike <i>AI Unplugged – Wir ziehen Künstlicher Intelligenz den Stecker</i>	325
Kerstin Strecker, Eckart Modrow <i>Eine Unterrichtssequenz zum Einstieg in Konzepte des maschinellen Lernens</i>	335

Kurzbeiträge

Nils Pancratz, Ira Diethelm

Was haben Staubsaugerroboter, Spielekonsolen und Smartphones gemeinsam? Schülervorstellungen von Teil-Ganzes-Beziehungen in Informatiksystemen 347

Marco Thomas, Niklas Klinge

Zur Beliebtheit des Schulfach Informatik in der Oberstufe nordrhein-westfälischer Schulen - Studie zu Einflüssen auf die Wahl von Informatik in der Oberstufe 351

Bettina Waldvogel

Informatikwissen im Schulalltag sichtbar machen 355

Poster

Christoph Gräbl

Algorithmen im Anfangsunterricht mit RunCode 361

Claudia Hildebrandt, Matthias Matzner

Identifizierung leistungsstarker Schülerinnen und Schüler in der Informatik 363

Peter Micheuz

Gratulation! Sie haben die Prüfung bestanden 365

Philipp Straube, Martin Brämer, Hilde Köster

Freies Explorieren und Programmieren im Sachunterricht 367

Juliane Wegner, Mona Arndt

Aufhören oder weitermachen? Ursachen des Studienabbruchs im Fach Informatik 369

Workshops

Mike Barkmin, Torsten Brinda

Informatiksysteme für den Unterricht aufbereiten 373

Peter Brichzin, Petra Kastl, Ralf Romeike <i>Agile Schule – Weiterentwicklung des Projektunterrichts in der Informatik und darüber hinaus</i>	374
Julian Dorn <i>InstaHub Datenbanken und Datenschutz mit einem extra für den Unterricht entwickelten sozialen Netzwerk unterrichten</i>	375
Julian Dorn <i>JavaScript als ideale textbasierte Programmiersprache im Unterricht anhand von ada7 lebensnah und effizient für Mädchen und Jungen</i>	376
Anja Gärtig-Daug, Alexander Werner, Ute Schmid <i>Informatische Konzepte spielerisch begreifen und anwenden</i>	377
Laura Hembrock, Patrick Tschorn, Michael Brinkmeier <i>IoT zum Nachbauen</i>	378
Tobias Jördens, Jens Gallenbacher <i>Quadrologik – Workshop</i>	379
Andrea Langer, Nadine Bergner, Ulrik Schroeder <i>Mehr als Programmieren lernen - einen Mikrocontroller-Roboter zum Leben erwecken / Zweitägiges Workshopkonzept zur Mädchenförderung mit dem B-O-B-3</i>	380
Kris Markussen, Kirsten Alich, Jana Gerdes, Mareike Daeglau, Nils Pancratz, Ira Diethelm <i>Über Lottofeen zu (mehr) Computernerds: Undoing Gender im Informatikunterricht durch eine selbstreflektierte Konfrontation mit geschlechtsspezifischen Stereotypisierungen</i>	381
Maximilian Marowsky, Paul Ohm, Julian Dierker <i>Pair Programming im Klassenraum – Eine Digital Game-based Learning Methode für die Sekundarstufe I</i>	382
Elisaweta Ossovski, Michael Brinkmeier <i>Maschinelles Lernen Unplugged – Ein Workshop zum Maschinellen Lernen in der Oberstufe</i>	383

Tobias M. Schifferle, Eike Rösch, Peter Suter <i>Wie gelingt eine obligatorische Informatikweiterbildung für Tausende von Lehrpersonen der Volksschule? - ein Erfahrungsbericht mit erlebbaren Praxisteilen</i>	384
Michael Schlichtig, Simone Opel, Carsten Schulte, Rolf Biehler, Daniel Frischemeier, Susanne Podworny, Thomas Wassong <i>Maschinelles Lernen im Unterricht mit Jupyter Notebook</i>	385
Pascal Schmidt, Stefan Strobel <i>Künstliche Intelligenz (KI) im Informatikunterricht der Sekundarstufe II</i> .	386

Eingeladene Vorträge

Lernen in der digitalen Welt

Oliver Kaczmarek,
Sprecher für Bildung und Forschung der SPD-Bundestagsfraktion

Abstract: Den Wandel unserer Arbeits- und Lebenswelt gerecht zu gestalten, ist die politische Herausforderung vor der wir bildungspolitisch, aber auch weit darüber hinaus stehen. Digitalisierung, Ökologisierung und Globalisierung greifen ineinander und verändern unsere ganz persönliche Lebens- und Arbeitswelt ebenso wie die Gesellschaft als Ganzes. Gerade die Bildungspolitik ist deswegen gefordert zu handeln, da die Vermittlung von adäquaten Kompetenzen und Wissen entscheidend dafür sein werden, dass der Wandel erfolgreich im Sinne der Menschen gestaltet werden kann. Damit es gerecht dabei zugeht, sind drei zentrale Themen wichtig. Erstens gilt es die Selbstbestimmung der Menschen im digitalen Zeitalter zu garantieren. Zweitens sollen die Aussichten auf eine gute Beschäftigung gesichert werden und drittens soll von Anfang an eine Kultur des Lernens etabliert werden, die lebensbegleitendes Lernen fördert.

Selbstbestimmung im digitalen Zeitalter bedeutet, dass nicht die Maschinen über das Leben der Menschen bestimmen, sondern die Menschen über die Maschinen bestimmen. Dafür sind neue Kompetenzen im Umgang mit digitalen Geräten nötig. Wie gehen Menschen kritisch mit der ständigen Verfügbarkeit von Informationen um und wie können sie den Wert und die Qualität einer Information unterscheiden? Wie kann es gelingen, dass Menschen sich auch von der digitalen Welt abgrenzen können, damit sie nicht durch permanente Benachrichtigungen unter Stress gesetzt werden? Wie geht man am besten souverän mit den eigenen persönlichen Daten um?

Die Arbeitswelt der Zukunft wird sich stark von der heutigen unterscheiden. Viele Arbeitsabläufe werden automatisch ablaufen und die Tätigkeiten der Menschen werden sich wandeln. Sind heute noch viele Berufe von den Tätigkeitsfeldern Herstellen und Produzieren geprägt, wird in Zukunft der Bereich Überwachen und Kontrollieren im Vordergrund stehen. Gleichzeitig wird standortunabhängiges Arbeiten immer leichter möglich werden. Menschen aus unterschiedlichen Teilen der Welt können unabhängig von Zeit und Ort miteinander kooperieren. Auf diese gewandelte Arbeitswelt sollen unsere Bildungseinrichtungen optimal vorbereiten. Dabei ist ebenso zu beachten, dass Menschen auch unter den neuen Voraussetzungen der digitalen Welt, selbstbestimmt ihr Leben gestalten können, um ihre Persönlichkeit zu entfalten.

Eine Kultur des lebensbegleitenden Lernens wird sich nur dann etablieren, wenn sie von Anfang an gelebte Praxis ist. Neugier und die Freude am Erwerb von neuen Kompetenzen und neuem Wissen sind schon im frühkindlichen Bereich zu fördern. Der selbstverständliche Umgang mit neuen Informationen, neuer Technik und einem stetig wachsenden Wissen ist eine Kompetenz, die in den Schulen vermittelt werden soll. Damit wird klar, dass es nicht mehr um den abschließenden Erwerb von Wissen geht, sondern dass fortwährendes Lernen ein integraler Bestandteil des gesamten Lebens ist.

Von Dagstuhl nach Frankfurt

Ira Diethelm¹

Abstract: Dieser Hauptvortrag beschreibt die Entwicklung vom *Dagstuhl-Dreieck* und der *Dagstuhl-Erklärung zur Bildung in der digital vernetzten Welt* [Br16] hin zum *Frankfurt-Dreieck* [Br19], das auf den folgenden Seiten mit dem Originaltext der interdisziplinären Autor*innengemeinschaft beschrieben ist. Der Hauptvortrag skizziert den Entstehungsprozess und erläutert wesentliche Gründe für die Weiterentwicklung des *Dagstuhl-Dreiecks* und wesentliche Punkte des neuen Modells.

Keywords: Bildung in der digital vernetzten Welt, Dagstuhl-Dreieck, Frankfurt-Dreieck

Literaturverzeichnis

- [Br16] Brinda, Torsten; Diethelm, Ira; Gemulla, Rainer; Romeike, Ralf; Schöning, Johannes; Schulte, Carsten: *Dagstuhl-Erklärung: Bildung in der digital vernetzten Welt*. 2016. <http://www.dagstuhl-dreieck.de>.
- [Br19] Brinda, Torsten; Brüggem, Niels; Diethelm, Ira; Knaus, Thomas; Kommer, Sven; Kopf, Christine; Missomelius, Petra; Leschke, Rainer; Tilemann, Friederike; Weich, Andreas: *Frankfurt-Dreieck zur Bildung in der digital vernetzten Welt*. In (Pasternak, Arno, Hrsg.): *Informatik und Schule 2019: Informatik für alle*. Köllen Verlag, Bonn, S. 25–33, 2019.

¹ Carl von Ossietzky Universität, Didaktik der Informatik, 26111 Oldenburg ira.diethelm@uol.de

Frankfurt-Dreieck zur Bildung in der digital vernetzten Welt

Ein interdisziplinäres Modell

Torsten Brinda,¹ Niels Brügger,² Ira Diethelm,³ Thomas Knaus,⁴ Sven Kommer,⁵ Christine Kopf,⁶ Petra Missomelius,⁷ Rainer Leschke,⁸ Friederike Tilemann,⁹ Andreas Weich¹⁰

Abstract: Das *Frankfurt-Dreieck* ist eine Erweiterung und Fortschreibung des in der *Dagstuhl-Erklärung* enthaltenen *Dagstuhl-Dreiecks* und richtet sich in Ergänzung dazu nun in erster Linie an Forscher*innen und andere Personen, die sich - primär reflexiv und theoretisch – mit Bildung im Kontext des digitalen Wandels beschäftigen. Das Papier will die aus verschiedenen Disziplinen an die Gruppe der Autorinnen und Autoren herangetragenen konzeptionellen Lücken beispielsweise zur Gestaltung von Informatiksystemen oder zur Einordnung und Rolle des Individuums als handelndes und medial adressiertes Subjekt schließen. Entsprechend gelten die politischen Forderungen der *Dagstuhl-Erklärung* [Br16] weiterhin, werden konzeptionell ergänzt und auf außerschulische Bildungskontexte erweitert.

Keywords: Bildung in der digital vernetzten Welt, Dagstuhl-Dreieck, Informatik, Medienwissenschaften, Medienpädagogik

Die Funktion dieses Modells besteht darin, einen überfachlichen Orientierungs- und Reflexionsrahmen für Bildungsprozesse im *digitalen Wandel* bereitzustellen und möglichst alle relevanten Perspektiven daran beteiligter Disziplinen einzubeziehen¹¹. Das gemeinsam entwickelte Modell – im Weiteren bezeichnet als *Frankfurt-Dreieck*, benannt nach dem Ort seiner Entstehung in zwei Expert*innen-Workshops 2017 und 2018 in Frankfurt am Main – basiert auf dem in der sogenannten *Dagstuhl-Erklärung* enthaltenen *Dagstuhl-Dreieck*, das 2016 unter Beteiligung von Akteur*innen aus Informatik, Informatikdidaktik, Medienpädagogik, Schulpraxis, Wirtschaft und Bildungspolitik in einem mehrtägigen Workshop

¹ Universität Duisburg-Essen, Didaktik der Informatik, torsten.brinda@uni-due.de

² JFF-Institut für Medienpädagogik in Forschung und Praxis, München niels.brueggen@jff.de

³ Carl von Ossietzky Universität Oldenburg, Didaktik der Informatik, ira.diethelm@uol.de

⁴ PH Ludwigsburg | Frankfurt University of Applied Sciences, FTzM, knaus@ftz.de

⁵ RWTH Aachen, Institut für Erziehungswissenschaft, sven.kommer@rwth-aachen.de

⁶ DFF - Deutsches Filminstitut & Filmmuseum, Frankfurt, kopf@dff.film

⁷ Universität Innsbruck, Institut für Medien, Gesellschaft und Kommunikation, petra.missomelius@uibk.ac.at

⁸ Universität Siegen, Medienwissenschaft, leschke@medien-peb.uni-siegen.de

⁹ PH Zürich, Bildung und Erziehung, Fachteam Medienpädagogik, friederike.tilemann@phzh.ch

¹⁰ HBK Braunschweig, Institut für Medienwissenschaft, an.weich@hbk-bs.de | TU Braunschweig, Projektgruppe Lehre und Medienbildung, a.weich@tu-braunschweig.de

¹¹ Dieser Text erscheint daher nicht nur hier, sondern auch in anderen Publikationsorganen der beteiligten Fachgesellschaften: merz 5/19 und MedienPaedagogik.com, Medienimpulse 58/19 sowie den Webseiten der Gesellschaft für Informatik e.V.

auf Schloss Dagstuhl erarbeitet und von einer breiten Öffentlichkeit – entsprechend seiner Intention – insbesondere von Praktiker*innen und Politiker*innen wahrgenommen wurde. Ziel ist es vielmehr, aus den disziplinären Perspektiven von Informatik, Informatikdidaktik, Medienpädagogik und Medienwissenschaft die Phänomene einer digitalen Welt und die daraus resultierenden Erfordernisse für Bildungsprozesse zu beschreiben und dadurch eine gemeinsame Reflexionsbasis zu entwickeln sowie darauf aufbauend – in künftigen Schritten – die notwendigen Kompetenzen für Partizipation in einer digital geprägten Welt zu definieren.

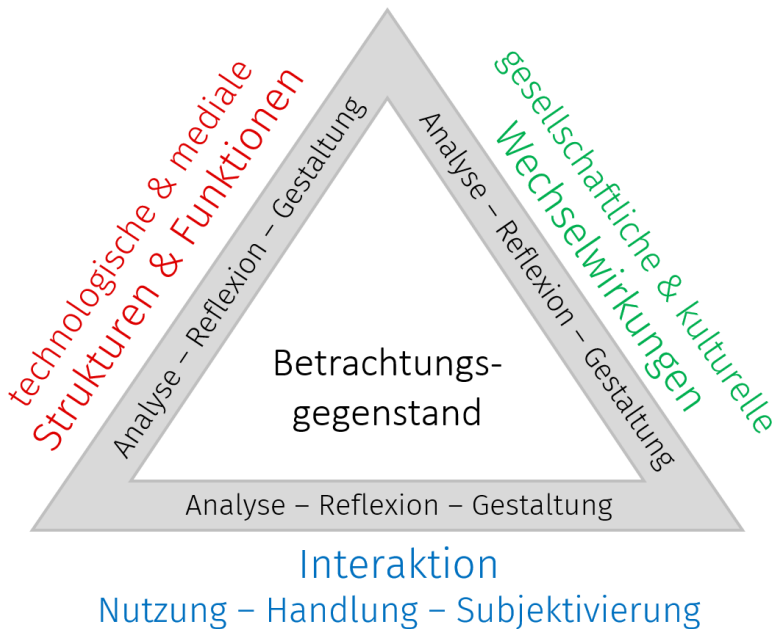
Eine Herausforderung im Diskussionsprozess der Autor*innengruppe war, dass es zu wesentlichen Kernbegriffen bislang kein etabliertes Begriffsverständnis gab – schon gar kein zwischen den beteiligten Disziplinen abgestimmtes. So wird beispielsweise „digitale Bildung“ häufig als Schlagwort verwendet (mal mit einem auf das Lehren und Lernen *mit* digitalen Mitteln eingeschränkten Bildungsverständnis, mal *einschließlich* informatischer Grundlagen gedacht usw.). Das Adjektiv „digital“ wird in der öffentlichen Diskussion und Berichterstattung oft als ein Synonym für „neuartig“ oder „modern“ verwendet. Dabei beschreibt es ursprünglich die Repräsentation von Daten und indirekt auch Information in einer Weise, die die automatische Verarbeitung mittels Computern ermöglicht, und „Digitalisierung“ damit die Umwandlung analoger in diskrete Werte, was heute im Wesentlichen durch binäre Signale realisiert wird. So ist Digitalisierung eines der drei Grundprinzipien der Informatik neben Automatisierung und Vernetzung, wird aber oft stellvertretend für diese genannt. Mit der Digitalisierung wurde die Voraussetzung für eine universelle Kompatibilität von Daten und Informationen geschaffen und zugleich die Bedingungen für die Integration bislang getrennter Praktiken, sozialer Strukturen und Technologien, was einen nachhaltigen Einfluss auf die tradierten räumlichen und temporären Unterscheidungen sowie soziale Ein- und Ausschließungen hat. Heute wird der Begriff der Digitalisierung in politischen und sozialen Kontexten vor allem zur Beschreibung von aktuellen informatisch und technisch induzierten gesellschaftlichen Transformationsprozessen genutzt.

Wir gehen davon aus, dass die Digitalisierung in den heutigen Gesellschaften die Kultur, die Infrastruktur und entsprechend die weitere Technologieentwicklung wesentlich mitprägt und sprechen daher vom *digitalen Wandel*. Die Teilhabe an politischen, kulturellen und ökonomischen Prozessen innerhalb der Gesellschaft setzt Fähigkeiten im Umgang *mit* und zur *Analyse, Reflexion* und *Gestaltung* von *digitalen Artefakten* voraus. Erforderlich hierfür ist die Kenntnis der informatischen Grundlagen sowie der medienwissenschaftlichen und erziehungswissenschaftlichen Zugänge und Diskurse.

Analog zum *Dagstuhl-Dreieck* werden im Modell drei Perspektiven ausdifferenziert, die Bildung für und über den digitalen Wandel aufgreifen muss. Diese werden im weiterentwickelten Modell bezeichnet als *technologisch-mediale Perspektive*, *gesellschaftlich-kulturelle Perspektive* und *Interaktionsperspektive*. Diesen Perspektiven sind jeweils die Prozesse *Analyse, Reflexion* und *Gestaltung* zugeordnet, die Lernende mit dem Ziel der Befähigung zur Partizipation an der durch Digitalisierung geprägten Welt und am digitalen Wandel jeweils durchlaufen sollen. Zugleich kann eine umfassende Analyse, Reflexion und Gestal-

tung des digitalen Wandels nur gelingen, wenn alle drei Perspektiven systematisch und sich wiederholend eingenommen werden.

Die *Mitte* des Modells bietet Raum für den jeweiligen *Betrachtungsgegenstand* der durch Digitalisierung geprägten Welt, also digitale Artefakte wie beispielsweise autonome Fahrzeuge, soziale Netzwerke, Hate Speech und Multitasking und damit in Zusammenhang stehende Phänomene, der dann aus Sicht der drei zuvor benannten Perspektiven und den damit verbundenen Prozessen aufgearbeitet werden soll.



Auf Basis dieses Modells sollen künftig Konkretisierungen im Hinblick auf Handlungsfelder wie Schule, außerschulische Bildungskontexte wie Kinder- und Jugendbildung, Kulturelle Bildung und Erwachsenenbildung, Berufsbildung und Hochschule, Lehrerinnen- und Lehrerbildung sowie Aus- und Fortbildung von pädagogischen Fachkräften entwickelt werden. Diese können dann in weiteren Schritten im Hinblick auf Kompetenzmodelle und fachdidaktischen- sowie mediendidaktische Fragen und insbesondere die Weiterentwicklung von vorhandenen (Unterrichts-) Konzepten und Empfehlungen der Fachgesellschaften [GI08, LK08, Gf13, LK15, GI16, Gf16, DG17, KMT17, Gf16, GI19] usw. ausgearbeitet werden.

Technologisch-mediale Perspektive

Ziele der Betrachtung aus einer *technologisch-medialen Perspektive* sind das Hinterfragen und Reflektieren der den Phänomenen und Artefakten der durch Digitalisierung geprägten Welt zugrundeliegenden Strukturen und deren Funktionsweisen sowie eine Befähigung zur (Mit-) Gestaltung solcher Artefakte und Phänomene. Dazu erfolgt eine Auseinandersetzung mit konzeptionellen Fragen, insbesondere mit informatischen und medialen Funktionsprinzipien digitaler Systeme, mit den zu deren Erstellung verwendeten informatischen und medialen Strukturierungs- und Gestaltungsmitteln und -formen, den sich durch sie ergebenden technischen Analyse- und Verarbeitungsmöglichkeiten sowie den an der „Oberfläche“ meist nicht sichtbaren kulturellen, politischen oder persönlichen Einschreibungen.

In dieser Perspektive werden damit zwei Aspekte verknüpft, die untrennbar miteinander verbunden sind:

1. Unter Anwendung langlebiger Informatik-Konzepte werden aus *informatischer Sicht* die Funktionsweise von digitalen Artefakten, die die digitale vernetzte Welt ausmachen, sowie damit in Zusammenhang stehenden Phänomenen hinterfragt und bewertet. Zugrundeliegende *Funktionsprinzipien* und *Strukturen der digitalen Artefakte* werden analysiert und aufgedeckt und damit Möglichkeiten zur *Gestaltung und Erweiterung der Funktion digitaler Systeme* unter Berücksichtigung von informatischen Problemlösestrategien und -methoden einerseits, aber auch zu einem reflektierten Umgang mit digitalen Systemen andererseits angelegt. Die Basis hierfür bilden theoretische und praktische Grundlagen der Informatik insbesondere in den Bereichen Digitalisierung, Automatisierung und Vernetzung und deren Anwendung auf aktuelle und gesellschaftlich relevante Themen (wie z. B. Big Data oder Künstliche Intelligenz) sowie Aussagen zu den praktischen und theoretischen Grenzen von Berechenbarkeit bzw. Automatisierung. Hinzu kommen Konzepte zur Kommunikation informatischer Systeme untereinander (z. B. Netzwerke, Protokolle, Verschlüsselung), Priorisierungen darin (insbes. Netzneutralität) sowie systematische Vorgehensmodelle zur Erstellung von digitalen Artefakten und Systemen.
2. Durch informatische Modellierung von Ausschnitten der Welt mit entsprechenden Mitteln und Werkzeugen sowie geprägt durch kulturelle Einschreibungen und die persönliche Perspektive von Entwickler*innen (z. B. Auswahl von Trainingsdaten für KI, normative Algorithmen ohne Legitimierung von Entwicklern), entstehen digitale Artefakte. Diese beeinflussen als soziotechnische Informatiksysteme mit charakteristischen Eigenschaften, Ästhetiken, Formen und Grenzen die menschliche Wahrnehmung und bedürfen daher auch einer Auseinandersetzung aus *medialer Sicht*. Von den Entwickler*innen und/oder den Auftraggeber*innen wird explizit und mitunter auch interessengeleitet, unreflektiert oder aufgrund kultureller Konventionen festgelegt, was sichtbar oder wahrnehmbar ist, wie auch, was in den Hintergrund tritt. Hierdurch wird die mit solchen Systemen mögliche Interaktion und insbesondere das Repertoire kultureller Ausdrucks- und Kommunikationsmöglichkeiten bestimmt.

Mit diesen charakteristischen Prägungen schreibt sich die Technologie mittels ihrer Artefakte, aber auch deren Geschichte und Genese, in die durch sie ermöglichten *kulturellen und sozialen Formen* ein: In ihnen sind *Sozialstrukturen* angelegt, in ihnen ist festgeschrieben, was in welcher Weise archiviert, was vergessen und ignoriert wird sowie was historisches Gewicht verliehen bekommt.

Darüber hinaus legen verwendete Technologien erforderliche Kompetenzen für ihre Nutzung fest. Umgekehrt kann die Reflexion und Kenntnis von solchen Determinationsverhältnissen in die Konstruktion von digitalen Artefakten einfließen, was zu einem dynamischen souveränen Umgang mit Technologien befähigen würde. Es ist daher unerlässlich, die Strukturen, Funktionen und Funktionsweisen von digitalen (Medien-) Systemen aus informatischer und medialer Sicht *analysieren, reflektieren und (mit-)gestalten* und diese Sichten aufeinander beziehen zu können. Solcherart fundiertes und verknüpftes Informatik- und Medienwissen erklärt technologische und mediale Phänomene mit langlebigen Konzepten und schafft zusammen mit der Entwicklung grundlegender Problemlösestrategien die Basis für die reflektierte Teilhabe an einer digital geprägten Welt.

Gesellschaftlich-kulturelle Perspektive

Der digitale Wandel prägt die sozialen Kommunikations- und Interaktionsbedingungen sowie die politische Organisation von Gesellschaften. Er bildet dabei nicht zuletzt auch einen kulturellen Möglichkeitsraum, der von Gesellschaften genutzt und gestaltet werden kann. Dasselbe gilt auch für die ökonomische Reproduktion von Gesellschaften. Es werden beispielsweise neue Arbeitsbedingungen, Produktionsmethoden und Austauschbedingungen entwickelt. Dadurch verändern sich die gesellschaftlichen Rollen von Akteur*innen sowie die Dynamiken gesellschaftlicher Entwicklung. Gesellschaften entwerfen Normen und Regeln für die Verwendung und den Einsatz von Technologien und Techniken, die die konkrete Bedeutung und den Einfluss von digitaler Technik strukturieren. So werden in gesellschaftlichen Aushandlungsprozessen die Bedingungen von Privatheit und Öffentlichkeit festgelegt, es werden Interaktionsmöglichkeiten geschaffen oder aber begrenzt und es wird der Zugang zu technischen Systemen geregelt. Gesellschaften prägen Bildungsinstitutionen und regulieren durch Bildungsangebote für den Erwerb entsprechender Kompetenzen das Verständnis für und die gesellschaftliche Nutzung von digitalen Technologien und Techniken. Der Grad der gesellschaftlichen Durchdringung mit Technologien und auch deren soziale Rolle wird daher wesentlich vom Bildungssystem bestimmt. Dabei bestimmt der Grad der Enkulturation digitaler Technologien und Techniken auch die Rolle einer digitalen Kultur in der Gesellschaft und deren Verhältnis zu analogen kulturellen Artefakten.

Aus *gesellschaftlich-kultureller Perspektive* werden deshalb Wechselwirkungen zwischen Individuen, Gesellschaft und digitalen Systemen vor dem Hintergrund der Medialisierung und des digitalen Wandels analysiert und reflektiert. Im Vordergrund stehen die Veränderungen, denen Individuen und Gesellschaft unterworfen werden, sowie eine Analyse und

Bewertung von Chancen und Problemen, die sich durch den digitalen Wandel ergeben. Das betrifft beispielsweise sich durch digitalisierungsbezogene Kompetenzen eröffnende Möglichkeiten für wirtschaftliches, ökologisches, nachhaltiges und politisches Handeln und die damit einhergehende Verantwortung einerseits sowie die sich durch Nutzung digitaler Systeme ergebenden Datenspuren der*des Einzelnen im Netz und die damit verbundenen Profilbildungen für kommerzielle oder ideologische Zwecke andererseits. Zudem werden unter den Bedingungen digitaler Infrastrukturen das Erkennen und die Bewertung medialer Einflüsse sowie die aktive Teilhabe an gesellschaftlichen und kulturellen Entwicklungen voraussetzungsreicher. Sie erfordern Hintergrundwissen und spezifische Kompetenzen, wie beispielsweise das Beurteilen von Information oder die Entwicklung eigener Standpunkte. Hierbei lassen sich widersprüchliche Tendenzen feststellen: die erhöhten Partizipationsmöglichkeiten steigern den potentiellen Einfluss von Individuen, wohingegen die wachsende Komplexität einer digital gewandelten/beeinflussten Kultur und die Geschlossenheit autonomer und/oder selbstlernender Systeme den individuellen und gesellschaftlichen Ein- und Zugriff wiederum erschweren. Dadurch stellt sich die Frage nach einer Mitgestaltung von „digitaler“ Kultur und ihrer Enkulturation grundlegend neu. Zugleich können in der digitalen Welt mittels digitaler Technologien (neue) soziale Ungleichheiten produziert beziehungsweise verfestigt werden, so dass auch Fragen sozialer Gerechtigkeit und sozialen Ausgleichs neu reflektiert werden müssen.

Ein weiterer Aspekt ist die historische und die damit einhergehende politische Entwicklung von Informations- und Kommunikationstechniken. Dazu muss analysiert werden, welche Normen und Regeln in mediengestützten sozialen Prozessen wirksam sind, wie und von wem sie ausgestaltet werden und welche Machtstrukturen hier eingeschrieben sind. Konkret sind beispielsweise Fragen der Netzneutralität in den Blick zu nehmen – auch im Hinblick der Entstehung des Internets und seiner Dynamiken vor dem Hintergrund historischer Prozesse. Dazu gehören auch ökonomische Implikationen digitaler Technologien und Techniken. So müssen Fragen wie die von Nutzung versus Besitz, die des Eigentums an Daten, die von Persönlichkeitsrechten, die der Mündigkeit der verschiedenen Akteur*innen, die der informationellen Selbstbestimmung sowie die eines zivilen Ungehorsams gegenüber immer autonomer werdenden technischen Systemen und die der gesellschaftlichen Teilhabe gerade auch aus einer ethischen Perspektive analysiert werden.

Interaktionsperspektive

Im Fokus der *Interaktionsperspektive* stehen die Menschen, zentral sind die Fragen, wie sie vor dem Hintergrund der technologisch-medialen und gesellschaftlich-kulturellen Voraussetzungen welche digitalen Medien und Systeme *warum* und *wozu* nutzen, *inwiefern* sie am digitalen Wandel teilhaben und ihn mitgestalten (können) sowie *wie* sie sich als handlungsfähige Subjekte konstituieren (vgl. auch Medienaneignung). Dabei sind die Aspekte Nutzung, Handlung und Subjektivierung zentral.

Unter *Nutzung* ist die funktionale Anwendung von digitalen Medien und Systemen beispielsweise für rezeptive, gestalterische, kommunikative, problemlösende und organisatorische Zwecke gefasst. Diese Nutzungsoptionen, die von Einzelnen oder Gruppen von Personen wahrgenommen, selektiert und ggf. auch verändert werden, beziehen sich auf digitale Artefakte und die von ihnen eröffneten Möglichkeiten.

Im Rahmen von *Handlungen* werden diese Nutzungsoptionen in unterschiedliche soziale Praktiken integriert. Dabei werden kulturell tradierte Interaktions- und Kommunikationsformen sowohl aufgenommen als auch transformiert. Eine bewusste Aneignung dieser Nutzungsoptionen setzt stets bestimmte Handlungsmotive wie auch die Reflexion und Analyse der technologischen und medialen Funktionsprinzipien und Potentiale sowie rahmender soziokultureller Praktiken voraus – dies gilt gleichermaßen für den Einsatz von Bildungsmedien, Lehr- und Lerntechniken. Auf dieser Grundlage lassen sich die Gestaltungspotentiale digitaler Artefakte realisieren. Derartige Handlungsoptionen bilden auch den Horizont für die individuelle Kompetenzentwicklung.

Mit *Subjektivierung* ist schließlich darauf verwiesen, dass im Zusammenwirken von digitalen Medien und Systemen sowie menschlichem Handeln auch die Identitätsbildung und -entwicklung angelegt, ermöglicht oder auch behindert werden können. Dies betrifft mehrere Ebenen: Konkret sind damit erstens Formen der Selbstthematizierung gemeint, die in und über digitale Medien und Systeme ermöglicht und nahegelegt werden. So sind beispielsweise in Interfaces von sozialen Netzwerken bestimmte Handlungsaufforderungen eingeschrieben, wie man sich in und über diese Dienste zeigen und darin agieren soll und sich zugleich damit selbst konstituiert. Aus der Interaktionsperspektive betrachtet, interessiert, welches Menschenbild durch diese Formen möglicher Selbstthematizierung konstituiert wird. Zweitens wird abstrakter auch die Frage gestellt, wie und vor dem Hintergrund welcher kulturellen Einschreibungen Subjekte in den jeweiligen Medien repräsentiert und adressiert sind, beispielsweise in Form von Interessenprofilen in Empfehlungs- und Filtersystemen oder auf Ebene von Interfaces und Interaktionsmöglichkeiten. Drittens sind beispielsweise im Angesicht von Data Analytics und Künstlicher Intelligenz traditionell auf Subjekte bezogene Konzepte wie Autonomie und Authentizität auch auf technologisch-medialer Ebene in den Blick zu nehmen.

Reflektiert werden soll aus der Interaktionsperspektive, wie und warum digitale Medien und Systeme als Werkzeuge jeweils für konkrete Vorhaben ausgewählt und genutzt werden. Dies erfordert eine Orientierung hinsichtlich der vorhandenen Möglichkeiten und Funktionsumfänge gängiger Werkzeuge in der jeweiligen Anwendungsdomäne sowie deren sichere Handhabung, aber auch die Kenntnis ökonomischer, gesellschaftlicher und politischer Interessen, welche Anbietende von digitalen Werkzeugen vertreten. Mit dem eigenen Handeln stellt sich so auch immer die Frage, welche anderen Handlungsoptionen individuell und sozial wünschenswert und realisierbar wären. Gleichzeitig ist aus dieser Perspektive immer auch zu reflektieren, welche Subjektpositionen technologisch-medial und kulturell angelegt sind, wie Subjekte sich in diesem Rahmen konstituieren und inwiefern sich Subjektivität angesichts digitaler autonomer Systeme transformiert.

Anschlüsse

Die drei Seiten des *Frankfurt-Dreiecks* beschreiben jeweils unterschiedliche Perspektiven für die *Analyse*, *Reflexion* und *Gestaltung* von Artefakten und Phänomenen einer durch digitale Medien und Systeme geprägten Welt. Dies schließt jeweils unterschiedliche Zugänge zur Erklärung der digitalen Artefakte und damit verbundener Phänomene ein. Das (theoretisch-konzeptionelle) Modell bietet eine begriffliche und strukturelle Grundlage, um an die Diskurse der Disziplinen Informatik, Informatikdidaktik, Medienpädagogik und Medienwissenschaft anschließen zu können, in einen produktiven interdisziplinären Austausch einzutreten und eigene anschlussfähige Theoriebildung zur Ausdifferenzierung und Konkretisierung voranzutreiben.

Für Bildungskonzepte, die digitale Medien und Systeme einschließlich der damit verbundenen Phänomene und ihrer Grundlagen adressieren und zur Teilhabe an der durch sie geprägten Welt befähigen sollen, ergibt sich aus dem *Frankfurt-Dreieck* die Maßgabe, dass sowohl die technologischen und medialen Strukturen und Funktionen, als auch die gesellschaftlich-kulturellen Wechselwirkungen sowie die Nutzungs-, Handlung- und Subjektivierungsweisen in Interaktionen mit digitalen Medien und Systemen einzubeziehen sind. Das übergeordnete Ziel muss dabei sein, digitale Artefakte und mit ihnen verbundene Phänomene im Zusammenspiel dieser drei Perspektiven analysieren, reflektieren, gestalten und damit erklären und beurteilen zu können.

Vor dem Hintergrund derartiger Bildungskonzepte gilt es im Austausch mit Bildungspolitik und -praxis konkrete Kompetenzanforderungen weiterzuentwickeln und im Zusammenwirken informatischer, informatikdidaktischer, medienwissenschaftlicher und medienpädagogischer Expertise (fach-)didaktische Szenarien und Lernmaterialien zu entwickeln, die den Auf- und Ausbau dieser Kompetenzen in Bildungseinrichtungen ermöglichen. Dieses Rahmenmodell kann dabei für alle Praxis- und Handlungsfelder in Bildungskontexten und pädagogischer Arbeit adaptiert werden: für die allgemeinbildende Schule, für die Hochschule, die Lehrer*innenbildung wie auch für außerschulische Bildungskontexte, wie die Kinder- und Jugendarbeit und Erwachsenenbildung. Perspektivisch ergibt sich so ein umfassender, wissenschaftlich fundierter und interdisziplinär getragener Katalog von Zielstellungen und Maßnahmen für Bildungskonzepte in einer durch digitale Medien und Systeme geprägten Welt.

Literaturverzeichnis

- [Br16] Brinda, Torsten; Diethelm, Ira; Gemulla, Rainer; Romeike, Ralf; Schöning, Johannes; Schulte, Carsten: , Dagstuhl-Erklärung: Bildung in der digital vernetzten Welt. <http://www.dagstuhl-dreieck.de>, 2016.
- [DG17] DGfE Sektion Medienpädagogik: Orientierungsrahmen für die Entwicklung von Curricula für medienpädagogische Studiengänge und Studienanteile. *Medienpädagogik: Zeitschrift für Theorie und Praxis der Medienbildung*, Dezember 2017.

- [Gf13] GfM - Gesellschaft für Medienwissenschaften, Strategiekommision und AG Medienkultur & Bildung: , "Medienkultur und Bildung"Positionspapier der GfM, 2013.
- [Gf16] GfM Gesellschaft für Medienwissenschaften: , Stellungnahme der Arbeitsgemeinschaft 'Medienkultur und Bildung' der Gesellschaft für Medienwissenschaft (GfM) zum Entwurf der Strategie der Kultusministerkonferenz "Bildung in der digitalen Welt", 2016.
- [GI08] GI - Gesellschaft für Informatik e.V.: , Grundsätze und Standards für die Informatik in der Schule. <http://www.informatikstandards.de>, 2008.
- [GI16] GI – Gesellschaft für Informatik e.V.: , Bildungsstandards Informatik – Sekundarstufe II. <http://www.informatikstandards.de>, 2016.
- [GI19] GI – Gesellschaft für Informatik e.V.: , Kompetenzen für informatische Bildung im Primarbereich. <http://www.informatikstandards.de>, 2019.
- [KMT17] Knaus, Thomas; Meister, Dorothee M.; Tulodziecki, Gerhard: Futurelab Medienpädagogik: Qualitätsentwicklung – Professionalisierung – Standards. Thesenpapier zum Forum Kommunikationskultur 2017 der GMK. MedienPädagogik: Zeitschrift für Theorie und Praxis der Medienbildung, Oktober 2017.
- [LK08] LKM - Länderkonferenz MedienBildung: , Kompetenzorientiertes Konzept für die schulische Medienbildung -LKM-Positionspapier Stand 01.12.2008. <https://lkm.lernnetz.de>, 2008.
- [LK15] LKM - Länderkonferenz MedienBildung: , Kompetenzorientiertes Konzept für die schulische Medienbildung – LKM-Positionspapier Stand 29.01.2015. <https://lkm.lernnetz.de>, 2015.

Informatische Bildung und inklusive Pädagogik

Dino Capovilla¹

Abstract: Informatiksysteme haben unser Zusammenleben in vielerlei Hinsicht verändert und von diesen Veränderungen haben auch behinderte Menschen maßgeblich profitiert. In der informatischen Bildung fand diese zentrale Bedeutung von Informatiksystemen als Mittel zur Förderung von Teilhabe bisher jedoch wenig Beachtung, was sich mit Blick auf die Neufassung der ländergemeinsamen inhaltlichen Anforderungen für die Informatik-Lehramtsstudiengänge offenbar ändern soll. Mit der vorliegenden Arbeit sollen einige Ansätze aufgezeigt werden, wie die Didaktik der Informatik im Sinne einer inklusiven Pädagogik für behinderungsbedingte Bedürfnisse anschlussfähiger werden kann. Dabei liegt der Fokus auf der Herstellung einer Vielfalt unterschiedlicher Handlungs- und Ausdrucksmodi und der sensorischen und kognitiven Parallelisierung von Unterrichtsinhalten.

Keywords: Informatische Bildung, Inklusive Pädagogik, Sensorische Parallelisierung, Anforderungsniveaus, Leichte Sprache, Assistive Technologien

1 Einleitung

Digitalisierung verändert und reorganisiert unser Interaktionsverhalten, unsere gesellschaftliche Selbstbeschreibung und letztlich sämtliche Strukturen unserer schulischen, beruflichen und privaten Lebensbereiche [KM16]. Dieser Wandel wird wesentlich durch rasante technologische Innovationen angetrieben, die geprägt durch Vorstellungen von informatisch hochqualifizierten Menschen verfügbar gemacht werden [GI16a]. Auf der anderen Seite wächst der Kreis der Nutzenden weiter und schließt somit immer mehr Menschen ein, deren Stärken in technikfernen Bereichen liegen oder die gar mit ihrer Technikaversion kokettieren.

Bereits vor 30 Jahren hatten Wilfried und Ute Brauer [BB89] unser heutiges informationsgesellschaftliches Selbstverständnis bemerkenswert treffend prophezeit. Durch die omnipräsente, einfache und unbegrenzte Verfügbarkeit von Informationen werde die Überzeugung reifen, dass eine verständnisschaffende Ausbildung im Umgang mit Informatik-Systemen nicht mehr notwendig sei. In dieser reinen Anwendungsorientierung werde sich die technomorphe Vorstellung von Informatiksystemen hin zu einer anthropomorphen Vorstellung mythologisieren. Der Umgang mit solchen vermenschlichten Informatiksystemen werde in Zukunft weniger rational und viel mehr emotional sein. Diese Entwicklung nahm mit Karl Klammer ihren Anfang und ist mit Siri, Alexa und Cortana längst Wirklichkeit geworden.

¹ Institut für Rehabilitationswissenschaften/Humboldt-Universität zu Berlin, Abteilung Pädagogik bei Beeinträchtigungen des Sehens, Unter den Linden 6, 10099 Berlin, dino.capovilla@hu-berlin.de

Um Menschen auf die Anforderungen und Herausforderungen unserer Zeit im Rahmen ihrer schulischen Bildung vorzubereiten und ihnen ein möglichst hohes Maß an Selbstbestimmung und Autonomie in der Anwendung von digitalen Systemen zu geben, fordert die KMK [KM16] eine flächendeckende und systematische Vorbereitung. Dies impliziert, dass diese Vorbereitung kein durch Intuition oder Learning-by-doing gesteuerter Prozess der Selbstläufigkeit ist, sondern auf einer nicht zu ignorierenden und durchaus komplexen Fachlichkeit beruht [GI18]. Vor diesem Hintergrund einer informatischen Bildung, die sich an alle Menschen richtet, geht es im vorliegenden Artikel um die Zielgruppe von Menschen mit Behinderungen. Wie zu zeigen sein wird, nimmt die informatische Bildung eine erweiterte Bedeutung bei den Bemühungen um die Realisierung schulischer Inklusion und gesellschaftlicher Teilhabe ein. Dabei sei vorausgeschickt, dass es im inklusiven Kontext keine spezielle Didaktik der Informatik für Menschen mit Behinderungen geben kann. Das sonderpädagogische Anliegen diesbezüglich besteht darin, die allgemeine Didaktik der Informatik anschlussfähig für die besonderen Lernvoraussetzungen von Menschen mit Behinderungen zu gestalten, was im Wesentlichen mit einigen methodischen und wenigen inhaltlichen Erweiterungen verbunden ist. Es gilt also zu überlegen, welche Veränderungen im Regelunterricht sinnvoll und notwendig sind, damit mehr Lernende mit Behinderungen die Dinge lernen können, die auch alle unbehinderten Lernenden lernen.

Bevor nun diese besondere Rolle von Informatiksystemen näher beschrieben wird und anschließend der Versuch unternommen wird, einige inklusive pädagogische Ansätze mit der informatischen Bildung zu verbinden, sollen die Begriffe informatische Bildung und Inklusion in einer für diesen Artikel geeigneten Form bestimmt werden.

Hubwieser [Hu07, S. 43ff.] subsumiert die Unterstützung von Lernprozessen, die Bedienungsschulung und den Informatikunterricht unter das Dach der informatischen Bildung, wobei er hinsichtlich des Informatikunterrichts die Prinzipien, Konzepte und Strategien zur Planung, Konstruktion, Beschreibung und Bewertung abstrakter Informatiksysteme besonders hervorhebt. Ergänzt und konkretisiert werden sollte diese Definition durch medienpädagogische Aspekte, die bei Hubwieser implizit im Zusammenspiel aus Benutzungsschulung und Informatikunterricht verortet sein dürften. Medienbildung soll Kompetenzen in den Bereichen Wahrnehmungs-, Nutzungs- und Konstruktionsprozesse von und mit Medien, sowie der kritischen Verarbeitung und Bewertung von Medien und deren Einflüsse fördern [De18]. Die informatische Bildung soll damit zusammenfassend Lernende zu kompetenten und reflektierten Anwenderinnen und Anwendern von bestehenden Informatiksystemen machen und ihnen durch das Verständnis der Wirkprinzipien dieser Informatiksysteme Möglichkeiten zu deren aktiver Mitgestaltung eröffnen [CA12, S. 4].

Eine allgemein akzeptierte Begriffsbestimmung schulischer Inklusion scheint es weder im wissenschaftlichen noch im gesellschaftlichen Diskurs zu geben. Trotz dieser begrifflichen Unschärfe wurde schulische Inklusion 2012 zum Bildungsziel aller Bundesländer erhoben, was mit der Idee des gemeinsamen Lernens von Kindern und Jugendlichen mit und ohne Behinderungen verbunden wurde [KM12]. Da jedoch

gemeinsames Lernen nicht bereits mit der lediglich gemeinsamen Anwesenheit im selben Raum realisiert wird, wird in der Forschungsliteratur das gemeinsame Lernen mit den beiden Zielen der effektiven Förderung und der sozialen Anerkennung der Lernenden mit sonderpädagogischem Förderbedarf verbunden [Ge15].

2 Informatiksysteme und die Bildung behinderter Menschen

Die UN-Behindertenrechtskonvention beschreibt in diversen Artikeln die Bereitstellung und den Einsatz von Technologien und Medien als wesentlich für die Durchsetzung der Rechte von behinderten Menschen. Dabei geht es u.a. um die Verwirklichung des Rechts auf uneingeschränkte Teilhabe an der Informationsgesellschaft, der mediatisierten Welten und der Kommunikationskultur [Bo14]. Vor diesem Hintergrund sollte die von Diethelm und Brinda [GI18] vorgeschlagene Bestimmung der Rollen von Informatiksystemen im Bildungsprozess ergänzt werden. Informatiksysteme dienen im Bildungsprozess nicht nur als Organisationsmittel, Lehr-Lern-Mittel, Werkzeug und Unterrichtsgegenstand, sondern auch als Mittel zur Förderung von Teilhabe.

In dieser Rolle haben Informatiksysteme die Lebensbedingungen von behinderten Menschen in den letzten Dekaden entscheidend verbessert [Mi15] und in vielen Fällen überhaupt erst die Rahmenbedingungen geschaffen, die gemeinsames Lernen möglich machen [CH13]. Wesentlich dabei sind die Möglichkeiten, die sich durch die technologiegestützte Interaktion zwischen behinderten und unbehinderten Lernenden ergeben haben und die Möglichkeiten, die den Zugang zu Informationen immer weiter in den Bereich der Selbstbestimmung von behinderten Menschen verschieben. Da zahlreiche dieser Technologien und Konzepte inhaltlich zurecht der Informatik zugeordnet werden und vor allem die Bedienung immer enger mit der Anwendung und dem Verständnis von Standard-Soft- und Hardware verbunden ist, muss sich die informatische Bildung gründlich mit ihrer Rolle in der Umsetzung der schulischen und gesellschaftlichen Inklusion auseinandersetzen. Diese besondere Rolle wird in der Neufassung der ländergemeinsamen, inhaltlichen Anforderungen für die Fachwissenschaften und Fachdidaktiken in der Lehrerbildung für Informatik [KM17] spürbar, mit der zahlreiche Bezüge zu inklusionsrelevanten Inhalten hergestellt wurden.

In der Verbindung von Informatiksystemen und Inklusion gilt es zwischen dem Ziel und dem Weg zu unterscheiden. Das Ziel beschreibt Caspers [AM11] als die vollständige informationelle Partizipation, die darin besteht, alle Informations- und Kommunikationssysteme so zu gestalten, dass sie für alle Menschen unabhängig von Hardware, Software, Sprache, Kultur, Ort, physischen oder kognitiven Fähigkeiten nutzbar sind. Auf dem Weg zu diesem idealen Ziel müssen zwei grundsätzlich verschiedene Herangehensweisen zur technologiegestützten Förderung von Teilhabe kombiniert werden: Auf der einen Seite müssen die Technologien und das mediale Angebot selbst so verändert werden, dass es für möglichst viele Personen zugänglich ist. Auf der anderen Seite müssen die individuellen Voraussetzungen der einzelnen Person an

die Anforderungen der verfügbaren Technologien und des medialen Angebots angepasst werden, was durch spezielle Technologien, Medien und Bedienungskonzepte gelingt.

Insbesondere die erstgenannte Herangehensweise ist im Kontext der allgemeinen informatischen Bildung relevant. Mithilfe von Technologien und veränderten medialen Angeboten sollen Barrieren im Informationszugang und in der Interaktion abgebaut werden und Produkte sowie ganze Lebensräume für möglichst viele Menschen jeden Alters und unterschiedlicher Fähigkeiten verfügbar gemacht werden [SMM98]. Um ein entsprechendes gesellschaftliches Selbstverständnis zu etablieren und die Grundlagen für die Entwicklung entsprechender Kompetenzen zu legen, erscheint es sinnvoll, alle Lernenden möglichst früh in ihrer Bildungsbiographie mit dieser besonderen Bedeutung und Funktion von Technologien und Medien für unser Zusammenleben zu konfrontieren und zu sensibilisieren. Da das Ideal der universell designten Welt jedoch in absehbarer Zeit praktisch nicht vollständig erreicht werden kann, wird es auch weiterhin zusätzliche assistive Technologien und Medien geben müssen, welche die physischen, sensorischen oder kognitiven Voraussetzungen der einzelnen Person dahingehend stärken, dass sie in unterschiedlichen Umgebungen mehr Unabhängigkeit gewinnen und ihre behinderungsspezifischen Eigenarten in den Hintergrund rücken [SMM98]. Die konkrete Einweisung und Unterstützung im Umgang mit diesen ergänzenden Technologien und Medien stellt jedoch aufgrund des individuellen Charakters und der hohen Spezifität keinen Gegenstand der allgemeinen informatischen Bildung dar und ist in der additiven sonderpädagogischen oder rehabilitativen Förderung zu verorten [Ca15, S. 30].

Insbesondere in den letzten Jahren sind die Strukturen in unserer Welt beachtlich universeller geworden und immer mehr handelsübliche Endgeräte bieten Funktionen an, die individualisierte assistive Hard- und Softwarelösungen überflüssig machen. Konkret bedeutet dies, dass technologiegestützte Zugänglichkeit in Zukunft immer weniger eine Frage der Verfügbarkeit und der ökonomischen Möglichkeit sein wird, sondern viel mehr eine didaktische Herausforderung, die sich immer weiter in den Kernbereich der informatischen Bildung verschieben wird [Ca15, S. 28].

3 Ansätze für einen inklusiven Informatikunterricht

Aus der Perspektive der inklusiven Pädagogik muss die allgemeine Didaktik der Informatik den besonderen Lernvoraussetzungen von behinderten Menschen Rechnung tragen und entsprechend anschlussfähig sein. Diese Anschlussfähigkeit wird im Wesentlichen durch diverse methodische und wenige inhaltliche Erweiterungen realisiert. Die zentrale Fragestellung der inklusiven Didaktik der Informatik besteht also darin, welche Veränderungen im Regelunterricht sinnvoll und notwendig sind, damit mehr Lernende mit Behinderungen die Dinge lernen können, die auch alle unbehinderten Lernenden lernen. Inklusiver Informatikunterricht zeichnet sich daher entsprechend der Forderung der Kultusministerkonferenz [KM17] durch eine systematische Ausrichtung an

einer heterogenen Lerngruppe aus, die weit über die bereits heute übliche Binnendifferenzierung hinausgeht.

Bei der Unterrichtsplanung für homogene Lerngruppen orientiert sich die Lehrkraft in der Regel an ihrer fiktiven Vorstellung der oder des durchschnittlichen Lernenden und es obliegt den realen Lernenden, sich den hierdurch geschaffenen Bedingungen anzupassen. Genau dieses Vorgehen wird einer heterogenen Lerngruppe nicht gerecht. Im Unterricht für heterogene Lerngruppen wird versucht, anhand der individuellen Lernvoraussetzungen der einzelnen Lernenden zusätzliche, inhaltsgleiche Lernangebote zu schaffen, die zum einen den Unterricht für genau diese einzelnen Lernenden öffnen und zum anderen auch für weitere Lernende Vorteile mit sich bringen können. Der Unterricht orientiert sich damit an den Gemeinsamkeiten mehrerer „realer“ Lernender und nicht an den Eigenarten einer oder eines fiktiven vorweggenommenen Einzelnen. Eine solche systematische Ausrichtung lässt sich naturgemäß auf unterschiedlichen Wegen erreichen, die abhängig von der tatsächlichen Lerngruppe auch unterschiedlich gut geeignet sind.

Einige dieser Wege, die inzwischen auch in Europa an Bedeutung gewinnen, orientieren sich an den sieben Prinzipien des Universal Designs [SMM98], welche auf unterschiedliche Art und Weise auf den Bildungskontext übertragen werden [DV15]. Diese lassen sich zu drei Gruppen klassifizieren [Bu08]. Bei Ansätzen wie dem Universal Design in Education werden die sieben Prinzipien des Universal Designs direkt übertragen. Bei Ansätzen wie dem Universal Design for Instruction werden die sieben Prinzipien verändert und erweitert. Bei Ansätzen wie dem Universal Design for Learning werden schließlich gänzlich neue Prinzipien geschaffen.

Im Kontext der informatischen Bildung erscheint vor allem der Ansatz des Universal Designs for Learning [RM02] als geeignet, da die parallelen Lernangebote vor allem mithilfe von digitalen Werkzeugen und Artefakten bereitgestellt und realisiert werden. Informatiksysteme entfalten ihre Wirkung im Lernprozess damit nicht nur in der Rolle als Mittel zur Förderung von Teilhabe, sondern auch in den von Diethelm und Brinda [GI18] genannten Rollen als Organisationsmittel, Lehr-Lern-Mittel und Werkzeug. Das Universal Design for Learning fußt auf den vier Handlungsfeldern Lernziele, Methoden, Bewertung und Material, die mit den drei Strategien Vielfalt der Handlungs- und Ausdrucksmodi, der Motivations- und Intensionsmodi und der Repräsentationsmodi universeller gestaltet werden sollen [RM02]. Im Folgenden sollen die beiden Strategien Vielfalt der Handlungs- und Ausdrucksmodi und Vielfalt der Repräsentationsmodi beispielhaft mit der Unterrichtsplanung einer Einheit zur Datenstruktur „Baum“ verbunden werden, um die praktische Anwendung zu verdeutlichen.

In Bezug auf verschiedene Ausdrucks- und Handlungsmodi werden die inhaltlichen Lern- und Leistungsziele festgeschrieben, während die Wege und Mittel zu diesen Zielen bewusst freigestellt werden. Zum einen geht es also darum, dass sich die Lernenden zwischen unterschiedlichen Handlungsmodi entscheiden und zwischen diesen auch im Verlauf des Lernprozesses wechseln können. Zum anderen bleibt auch der Ausdrucksmodus des Produkts am Ende des Lernprozesses weitgehend unbestimmt. Die

Vergleichbarkeit und Bewertung der einzelnen Leistungen wird nicht anhand der Abweichungen der realen Produkte von einem fiktiven Normal- und Idealprodukt hergestellt, sondern über die mit dem Produkt abgebildete, zugrundeliegende Kompetenzvermutung.

In der Praxis muss diese Freiheit zwecks Realisierbarkeit eingegrenzt werden, da erfahrungsgemäß zahlreiche Lernende diese Freiheiten nicht ohne Führung ausschöpfen könnten oder wollen. Das Konzept „Baum“ könnte die Lehrkraft beispielsweise mit den beiden Ausdrucksmodi Quellcode und verbal-visuelle Darstellung planen. Innerhalb der Ausdrucksmodi könnte sie die Handlungsmodi über verschiedene Anforderungsniveaus realisieren. Einen sinnvollen Ansatz für die Realisierung von Anforderungsniveaus zeigt Roth [GU13] auf, der zwischen Reproduktion, Rekonstruktion und Konstruktion unterscheidet. Für den Ausdrucksmodus Quellcode könnten die drei Niveaus folgendermaßen verwirklicht werden: Abschreiben einer bestehenden Codevorlage und Vervollständigen einer bestehenden Lücken-Codevorlage (Reproduktion), Erstellung eines Codes durch Verkettung vermischter Programmteile oder auf der Grundlage eines funktionsähnlichen Codes (Rekonstruktion) und schließlich selbständiges Schreiben des Codes (Konstruktion). Für den Ausdrucksmodus der verbal-visuelle Darstellung könnten die Anforderungsniveaus wie folgt umgesetzt werden: Suchen und nachspielen von passenden Videos oder Vervollständigung eines thematischen Posters (Reproduktion), strukturelle Neugestaltung (Collage) eines thematischen Posters oder Nachbildung einer ikonischen Video-Visualisierung durch haptische Modelle (Rekonstruktion) und schließlich selbständige Gestaltung eines verbal-visuellen Mediums (Konstruktion). Damit würden sechs Zugänge entstehen, zwischen denen die Lernenden entsprechend ihrer Schwächen, Stärken und Neigungen wählen und natürlich auch im Lernverlauf wechseln können.

Bei der Vielfalt verschiedener Repräsentationsmodi werden die Unterrichtsinhalte mithilfe verschiedener möglichst gleichwertiger Repräsentationen angeboten. Die Bedeutung der unterschiedlichen Repräsentationsmodi unterstreicht auch die Kultusministerkonferenz [KM17] durch die ausdrückliche Nennung im fachspezifischen Kompetenzprofil. Demnach sollen Informatiklehrkräfte Möglichkeiten zur Illustration von informatischen Prinzipien, welche die visuelle, auditive und haptische Wahrnehmung ansprechen und die Regeln für leichte Sprache kennen. Für die Realisierung verschiedener Repräsentationsmodi gibt es mehrere Ansätze. Eine erste mögliche Form sind multikodale Aufbereitungen [We02]. Hierbei unterscheiden sich die Angebote hinsichtlich ihrer Repräsentation durch verschiedene Symbol- oder Kodierungssysteme. Ein Beispiel hierfür sind die äquivalenten textuellen Beschreibungen (Verbalisierung) von Tabellen oder Diagrammen in wissenschaftlichen Arbeiten. Eine zweite Möglichkeit besteht in der multimodalen Aufbereitung der Unterrichtsinhalte, bei der gleichzeitig unterschiedliche Sinnesmodalitäten angesprochen werden [We02]. Im Unterschied dazu sprechen monomodale Darbietungen in einem bestimmten Zeitpunkt genau einen Sinneskanal an. Das klassische Beispiel für eine multimodale Darbietung ist der Lehrfilm, bei dem das visuelle Angebot durch eine nicht deckungsgleiche akustisch wahrnehmbare Erklärung unterstützt wird. Eine dritte Methode ist die sensorische Parallelisierung [Ca15, S. 52].

Hierbei wird derselbe Inhalt in mindestens zwei unterschiedlichen sensorischen Formen angeboten. Wesentlich ist dabei, dass die sensorisch parallelen Angebote inhaltlich möglichst vollständig und weitestgehend substituierbar sein sollten. Dies bedeutet also, dass bei der sensorischen Parallelisierung mehrere eigenständige und inhaltsgleiche mono- oder multimodale Lernangebote zur Verfügung stehen, die sich insbesondere durch die Kombination der Kodierungsformen unterscheiden. Hierzu gehört beispielsweise die Digitalisierung von handgeschriebenem Material, die auditive Aufbereitung von gedruckten Texten, die ergänzende Bereitstellung von zugänglichen Präsentationsfolien, die textuelle Beschreibung (Verbalisierung) von Grafiken, Bildern und visuellen Modellen oder auch die Anfertigung von haptischen oder taktilen Modellen die zweidimensionalen Abbildungen entsprechen [CH13].

Neben dieser wahrnehmungsorientierten, eher technischen Herangehensweise, lässt sich die Vielfalt der Repräsentationen auch im Bereich der kognitiven Zugänglichkeit erhöhen. Ausgangspunkt sind in diesem Zusammenhang Leseeinschränkungen, welche das Verstehen und Behalten von Allgemein- oder Fachtexten erschweren oder unmöglich machen können [BM18]. Leseeinschränkungen können dabei ganz unterschiedliche Wurzeln haben, weshalb es auch unterschiedliche Ansätze gibt, diesen Barrieren zu begegnen. Einige Lernende können beispielsweise vom Einsatz inhaltsgleicher Versionen in unterschiedlichen Sprachen profitieren [Ca15, S. 125f.]. Ein solches mehrsprachiges Angebot lässt nicht nur positive Effekte für den Fremdspracherwerb erwarten, sondern erhöht auch die Zugänglichkeit für Lernende mit internationaler Geschichte. Zudem lassen sich in guten mehrsprachigen Darstellungen möglicherweise spannende kulturelle Facetten erkennen, die das Sprachgefühl fördern und motivieren. Die textuelle Parallelisierung muss jedoch nicht notwendigerweise mehrsprachig erfolgen. In Bildungsinstitutionen ist nach wie vor eine mittelschichtorientierte Sprache maßgebend [Bo14]. Diese kann insbesondere für Lernende mit Leseeinschränkungen ein Lernhindernis darstellen. Die informatische Bildung baut zudem in vielen Bereichen auf ein ausgeprägtes Fachvokabular auf, welches für Lernende eine faktische oder motivationale Barriere darstellen kann. In beiden Fällen können inhaltsgleiche Varianten auch mit hohem sprachlichem Anspruch [Bo14], sowie Varianten unterschiedlicher Sprachniveaus sinnvoll sein.

Ein systematischer Ansatz zur textuellen Parallelisierung auf unterschiedlichen Sprachniveaus ist die Leichte Sprache. Bei Leichter Sprache geht es darum, die Verständlichkeit standarddeutscher Texte für Personen mit Leseeinschränkungen zu erhöhen. Leichte Sprache zeichnet sich u.a. dadurch aus, dass einfache und präzise Worte in einfachen grammatikalischen Formen zu kurzen Sätzen zusammengefasst werden. Auf Worttrennungen, Abkürzungen, Passivformen und Negationen wird verzichtet. Sätze sind in größerer Schrift linksbündig ausgerichtet und die Satztrennung erfolgt durch Zeilenumbrüche. Schwierige Worte und Inhalte werden zudem durch weitere textuelle Erklärungen verdeutlicht [BM18].

Ein erster Eindruck lässt sich anhand der Definition des Studienfachs der Informatik im Wörterbuch für leichte Sprache (hurraki.de) gewinnen, wobei hier auf die üblichen

Zeilenumbrüche aus redaktionellen Gründen verzichtet wurde: „Informatik ist die Lehre von der Information. Die Menschen bekommen heute viele Informationen. Die Informationen kommen aus den Nachrichten. Die Informationen kommen von anderen Menschen. Die Informationen kommen von den Computern anderer Menschen. Die meisten Menschen lesen das nicht nur, sondern machen daraus neue Informationen. Sie rechnen Zahlen aus. Sie schreiben Texte. Sie drucken Listen. Nehmen Fotos auf oder Videos oder Musik. Kein Mensch weiß heute mehr, welche Informationen es gibt, und wo die alle herkommen. Darum brauchen wir Computer. [...]“. Zwischen der akademischen Standardsprache und der leichten Sprache lässt sich das Konzept der einfachen Sprache verorten, bei der lediglich sprachliche und keine inhaltlichen Vereinfachungen vorgenommen werden [BM18]. Konkret bedeutet dies, dass überschaubar lange Sätze genau einen Gedanken enthalten sollen und nicht zusammengesetzt werden, dass aktiv formuliert wird und sprachliche Bilder, Synonyme und Untertöne konsistent konkretisiert werden. Im vorliegenden Text würde also die parallele Verwendung der Begriffe „Informatiksysteme“, „Technologien und digitale Medien“ sowie „Digitale Systeme und Artefakte“ nicht mit den Grundsätzen der einfachen Sprache vereinbar sein.

Abschließend stellt sich die Frage, für welche Lernenden dieses Angebot geschaffen werden sollte und inwiefern der damit verbundene Aufwand überhaupt gerechtfertigt ist. Der günstigste Faktor für einen sensiblen Umgang mit den Bedürfnissen behinderter Lernender und eine damit verbundene zielführende Binnendifferenzierung bleibt, unabhängig von allen theoretischen und praktischen Konzepten und wissenschaftlichen Veröffentlichungen, eine engagierte und kluge Lehrkraft. Daran haben natürlich auch die postulierten Bildungsrechte in der Behindertenkonvention nichts geändert. Die entscheidende Veränderung durch die Behindertenrechtskonvention für die Unterrichtsplanung besteht darin, dass eben keine konkrete Person mehr notwendig ist, um Lernangebote zu schaffen, die ein breiteres Spektrum von Lernenden adressieren.

Für diesen inklusiven Denkansatz spricht außerdem, dass nicht nur Lernende mit Behinderungen von dieser systematischen Binnendifferenzierung profitieren können, sondern auch ein Teil der Lernenden ohne festgestellten Förderbedarf. Beispielsweise lässt die multimodale sensorische Aufbereitung eine tiefere Verarbeitung der Lerninhalte potentiell bei allen Lernenden erwarten [Br11, S. 95], während Lernende mit bestimmten sensorischen Präferenzen oder Beeinträchtigungen genau die Lernangebote wählen können, welche ihre Bedürfnisse treffen. Vergleichbar lässt sich auch im Bereich der sprachlichen Vielfalt und für die Vielfalt bei den Handlungs- und Ausdrucksmodi argumentieren.

4 Assistive Technologien

Wie bereits beschrieben, werden bei der Ausrichtung des Unterrichts an heterogenen Lerngruppen die Lernbedingungen so verändert, dass der Unterricht für eine bedeutend größere Zahl von Lernenden – unabhängig von deren individuellen Voraussetzungen –

offen steht. Davon abgesehen wird es jedoch vermutlich immer Lernvoraussetzungen geben, die so speziell sind, dass die allgemeine Unterrichtsausrichtung an Grenzen stößt. Ein Ansatz, um diese verbleibenden, fehlenden Fähigkeiten und Fertigkeiten im Unterricht zu kompensieren, besteht im Einsatz von assistiven Technologien. Klassische Beispiele für solche assistiven Technologien sind Elektrorollstühle, Prothesen, Sprachausgaben, Hörgeräte, Tafelkammersysteme usw.

Die Kultusministerkonferenz [KM17] ist diesbezüglich bei der Neufassung des fachspezifischen Kompetenzprofils überraschend weit gegangen und postuliert, dass Informatiklehrkräfte die Möglichkeiten, die sich durch den Einsatz von assistiven Technologien im Informatikunterricht eröffnen, einschätzen und bewerten können sollen. Auch wenn es hier nicht um den konkreten Einsatz von assistiven Technologien geht, ist diese geforderte Kompetenz durchaus anspruchsvoll. Beispielsweise sollte eine Informatiklehrkraft beurteilen können, ob und wie sehbehinderte oder blinde Lernende, ggf. durch den Einsatz von assistiven Technologien, funktionale Modelle erstellen können. Auf der Grundlage dieser Einschätzung sollte dann die Informatiklehrkraft die notwendige, sonderpädagogische Unterstützung einfordern, die ressourcenabhängig geleistet werden kann.

Wie bereits erwähnt, gehört die Einweisung und Unterstützung beim Einsatz assistiver Technologien zum additiven, sonderpädagogischen Angebot. Da nun aber assistive Technologien im schulischen Kontext häufig Informatiksysteme sind, stellt sich die Frage, ob sich diese aus der alleinigen, funktionalen Verknüpfung mit Hilfsmitteln lösen und für sinnvolle pädagogische Anwendungskontexte in der allgemeinen informatischen Bildung öffnen lassen [SB12, S. 21]. Auf der Grundlage des Dagstuhl-Dreiecks [GI16b] lassen sich solche Unterrichtsgegenstände vor allem aus der gesellschaftlich-kulturellen und technologischen Perspektive begründen. Aus gesellschaftlich-kultureller Perspektive verspricht die Auseinandersetzung mit assistiven Technologien diverse Einsichten und eine Sensibilisierung für die ansonsten häufig vorurteilsgeformten Lebensbedingungen behinderter Menschen. Ein kritisch hinterfragender Blick aus der technologischen Perspektive auf die Funktionsweise von assistiven Technologien könnte in Einzelfällen auch besondere methodische oder inhaltliche Ansätze rechtfertigen, wie das folgende Beispiel zeigen soll [Ca15, S. 30ff.; CG16].

Sehbeeinträchtigte Menschen verwenden grundsätzlich handelsübliche Hard- und Software, die durch assistive Technologien ergänzt wird. Im Unterschied zum herkömmlichen Arbeitsplatz wird der Rechner mit einer Screen Reader Software ausgestattet. Diese Software abstrahiert den Bildschirminhalt und extrahiert relevante Informationen. Diese Informationen können dann mittels einer Sprachausgabe oder einer Braillezeile auditiv bzw. haptisch wahrnehmbar ausgegeben werden.

Die Bedienungskonzepte dieses dadurch entstehenden assistiven Informatiksystems unterscheiden sich grundlegend von der herkömmlichen Arbeitsweise von unbehinderten Menschen. Entscheidend hierfür ist, dass die grafische Benutzeroberfläche bei sehbeeinträchtigten Menschen ihre zentralen Vorteile verliert: Das sind die intuitive

Führung und die Maussteuerung. Die sich aus dem Verzicht auf Intuition und Maus ergebenden Nachteile müssen durch das abstrakte Verständnis der Bildschirminhalte und ein tastaturbasiertes Steuerungskonzept ersetzt werden. Mit anderen Worten heißt das, dass sehbeeinträchtigte Menschen ein maßgeblich visuell orientiertes System soweit abstrahieren müssen, dass die enthaltenen Konzepte unabhängig von der konkreten Ausgestaltung am Bildschirm verstanden werden können. Davon ausgehend müssen alternative, in der Regel nicht intuitive Steuerungstechniken erlernt werden. Muss ein Computer ohne das Ausgabemedium Bildschirm bedient werden, besteht Lernprozess vor allem darin, die Klassen der grafischen Benutzeroberfläche kennenzulernen und anhand von Beispielen die zugehörigen, nicht visuell-orientierten Bedienungskonzepte zu erarbeiten. Das tatsächliche, nicht visuell-orientierte Arbeiten besteht dann darin, die Klasse des gefundenen Objekts zu erkennen, sich an das Bedienungskonzept für genau diese Klasse zu erinnern und dieses dann zielorientiert anzuwenden. Eine solche Herangehensweise lässt sich erfahrungsgemäß als alternativer Handlungsmodus durchaus in den Informatikunterricht integrieren oder bereichert die ansonsten rein intuitiv geleitete, rezepthafte Bedienungsschulung.

5 Schluss

Am Ende dieses Artikels stehen zusammenfassend drei Thesen: Erstens kann es in einer inklusiven Pädagogik keine spezifische Didaktik der Informatik geben, sondern die Didaktik der Informatik muss sich dahingehend verändern, dass sie hinsichtlich der besonderen Bedürfnisse behinderter Lernender anschlussfähig wird. Zweitens sind Informatiksysteme wesentlich für die Förderung von Teilhabe und sollten in dieser Rolle auch Gegenstand der informatischen Bildung sein. Drittens verspricht eine sensorische und kognitive Parallelisierung Vorteile für behinderte und unbehinderte Lernende und ist damit eine inklusive und keine sonderpädagogische Maßnahme.

Vor diesem Hintergrund erscheint es bemerkenswert, dass die inklusionsfördernde Wirkung der informatischen Bildung in der Argumentation zur flächendeckenden Einführung eines Pflichtfachs Informatik bisher so wenig Beachtung gefunden hat. Informatische Bildung ist offenkundig für viele behinderte Lernende ganz unmittelbar Inklusionsunterricht und wenn wir dann davon ausgehen, dass wir durch das gegenseitige Verständnis von Interaktionsformen, Arbeitsweisen, Möglichkeiten etc. Vorurteile abbauen und in manchen Bereichen voneinander lernen können, dann auch für alle anderen.

6 Literaturverzeichnisses

- [AM11] AM, Aktion Mensch e.V.: Einführung in die Barrierefreiheit. www.einfach-fuer-alle.de/artikel/einfuehrung-barrierefreiheit/, Stand: 19.01.2019.

- [BB89] Brauer, W.; Brauer, U.: Better Tools - Less Education? In (Ritter, G. Hrsg.): Information Processing IFIP'89. Elsevier Science Publishers B.V., Amsterdam, S.101–106, 1989.
- [BM18] Bredel, U; Maaß, C.: Leichte Sprache. In (Maaß, C.; Rink, I. Hrsg.): Handbuch Barrierefreie Kommunikation. Frank & Timme, Berlin, S. 251-271, 2018.
- [Bo14] Bosse, I.: Zur Rolle der Medienpädagogik im Inklusionsprozess. VHN 2/2014, S. 149-152, 2014.
- [Br11] Brand, M.: Gedächtnis. In (Schütz, A.; Brand, M.; Selg, H.; Lautenbacher, S. Hrsg.): Psychologie: Eine Einführung in ihre Grundlagen- und Anwendungsfächer. W. Kohlhammer, Stuttgart, S. 89-103, 2011.
- [Bu08] Burgstahler, S.: Universal Design of Instruction. From Principles to Practice. In: Universal Design in Higher Education. MA: Harvard Education Press, Cambridge, S. 23-44, 2008.
- [Ca15] Capovilla, D.: Inklusion in der Informatischen Bildung am Beispiel von Menschen mit Sehschädigung. PhD thesis, TU München, München, 2015.
- [CA12] CASWG: Computer Science: A curriculum for schools. www.computingschool.org.uk/data/uploads/ComputingCurric.pdf, 19.01.2019, 2012.
- [CG16] Capovilla, D.; Gebhardt, M.: Assistive Technologien für Menschen mit Sehschädigung im inklusiven Unterricht. Zeitschrift für Heilpädagogik 1/2016, S. 4-15, 2016.
- [CH13] Capovilla, D.; Hubwieser, P.: Soziale Inklusion als fachdidaktisches Problem der Informatik. blind – sehbehindert 4/2013, S. 226-235, 2013.
- [De18] Dengel, A.: Digitale Bildung: ein interdisziplinäres Verständnis zwischen Medienpädagogik und Informatik. MedienPädagogik 33, S. 11–29, 2018.
- [DV15] DVfR: Universal Design im Kontext von Hochschule und Hochschulbildung. www.reha-recht.de/fachbeitraege/beitrag/artikel/beitrag-d11-2015/, Stand: 19.01.2019.
- [Ge15] Gebhardt, M.: Gemeinsamer Unterricht von Schülerinnen und Schülern mit und ohne sonderpädagogischen Förderbedarf. Ein empirischer Überblick. In (Kiel, E. Hrsg.): Inklusion im Sekundarbereich. W. Kohlhammer, Stuttgart, S. 39-52, 2015.
- [GI16a] GI: Stellungnahme zum KMK-Strategiepapier „Bildung in der digitalen Welt“, fb-iad.gi.de/fileadmin/FB/IAD/Dokumente/gi-fbiad-stellungnahme-kmk-strategie-digitale-bildung.pdf, 19.01.2019, 2016.
- [GI16b] GI: Dagstuhl-Erklärung: Bildung in der digitalen vernetzten Welt, gi.de/fileadmin/GI/Hauptseite/Themen/Dagstuhl-Erklärung_2016-03-23.pdf, 19.01.2019, 2016.
- [GI18] GI: Stellungnahme zu den Veränderungsvorschlägen zu den „Ländergemeinsamen inhaltlichen Anforderungen für die Fachwissenschaften und Fachdidaktiken in der Lehrerbildung“ (Fachstandards) der Kultusministerkonferenz auf Basis der KMK-Strategie zur „Bildung in der digitalen Welt“, fb-iad.gi.de/fileadmin/FB/IAD/Dokumente/gi-fbiad-stellungnahme-kmk-anforderungen-lehrerbildung-digitalisierung.pdf, Stand: 19.01.2019, 2018.

- [GU13] GU, Thüringer Forschungs- und Arbeitsstelle für den Gemeinsamen Unterricht/Inklusion: Hinweise und Anregungen zur Differenzierung, www.gu-thue.de/material/RROTH_Differenzierung.pdf, 19.01.2019, 2013.
- [Hu07] Hubwieser, P.: Didaktik der Informatik. 3. Auflage, Springer VS, Berlin, 2007.
- [KM12] KMK: Sonderpädagogische Förderung in Schulen 2001 bis 2010, www.kmk.org/fileadmin/pdf/Statistik/Dokumentationen/Dokumentation_SoPaeFoe_2010.pdf, 19.01.2019, 2012.
- [KM16] KMK: Bildung in der digitalen Welt, www.kmk.org/fileadmin/Dateien/pdf/PresseUndAktuelles/2017/Strategie_neu_2017_datum_1.pdf, 19.01.2019, 2016.
- [KM17] KMK: Ländergemeinsame inhaltliche Anforderungen für die Fachwissenschaften und Fachdidaktiken in der Lehrerbildung, www.kmk.org/fileadmin/Dateien/veroeffentlichungen_beschluesse/2008/2008_10_16-Fachprofile-Lehrerbildung_01.pdf, 19.01.2019, 2017.
- [Mi15] Miesenberger, K.: Advanced and Emerging Solutions: ICT and AT in Education of Low Vision and Blind Students. In (Kouroupetroglou, G. Hrsg.): Proceedings of ICEAPVI'15. Athen, S. 17-26, 2015.
- [RM02] Rose, D.; Meyer, A.: Teaching Every Student in the Digital Age. Universal Design for Learning. Association for Supervision and Curriculum Development, Alexandria, 2002.
- [SB12] Schnaak, T.; Böhmig, S.: Inklusive Medienpädagogik – was ist das? In (Landesarbeitsgemeinschaft Lokale Medienarbeit NRW in Kooperation mit der Technischen Jugendfreizeit- und Bildungsgesellschaft Hrsg.): In Materialien für eine inklusive Medienpädagogik, S. 17-21, 2012.
- [SMM98] Story, M. F.; Mueller, J. L.; Mace, R. L.: The Universal Design File. Designing for People of All Ages and Abilities. Revised Edition, files.eric.ed.gov/fulltext/ED460554.pdf 27.04.2019, 1998.
- [St14] Sturm, T.: Rekonstruktion der Herstellung und Bearbeitung von Differenz im inklusiven Unterricht mithilfe der Dokumentarischen Videointerpretation. In: (Bohnsack, R.; Fritzsche, B.; Wagner-Willi, M. Hrsg.): Dokumentarische Video- und Filminterpretation. Barbara Budrich, Opladen, S. 153-178, 2014.
- [We02] Weidenmann, B.: Multicodierung und Multimodalität im Lernprozess. Information und Lernen mit Multimedia und Internet 3, S. 45-62, 2002.

Wissenschaftliche Beiträge

Informatikunterricht in der Grundschule? – Erprobung und Auswertung eines Unterrichtsmoduls mit Calliope mini

Kevin Baum¹, Nadine Kirsch¹, Kerstin Reese¹, Pascal Schmidt¹, Lukas Wachter¹, Verena Wolf¹

Abstract: Im Zuge der Digitalisierung wird die Lebenswelt von Grundschulkindern mehr und mehr durch Informatiksysteme geprägt. Informatische Bildung in der Grundschule kann zur Entmystifizierung solcher Systeme maßgeblich beitragen und so die Grundlage für tiefgreifendes Verständnis legen. Bisher ist allerdings nur wenig bekannt über das Vorwissen der SchülerInnen in Bezug auf Informatiksysteme und über die Wirksamkeit der eingesetzten Unterrichtsmaterialien im Hinblick auf den Erwerb von informatischen Kompetenzen. In der hier beschriebenen Studie mit insgesamt 137 Kindern wurde das Vorwissen von Viertklässlern mit einem Leitfadeninterview ermittelt. Danach wurde eine fünfständige Unterrichtseinheit zu algorithmischen Grundbausteinen und dem Mikrocontroller Calliope mini durchgeführt. Die Unterrichtseinheit endete mit einer Aufgabenstellung, für die die Kinder das Eingabe-Verarbeitung-Ausgabe-Prinzip (EVA-Prinzip) nachvollziehen und mit dem Mikrocontroller umsetzen mussten. Die Bearbeitung dieser Aufgabenstellung wurde per Screencast festgehalten. Unsere Ergebnisse zeigen, dass ein Großteil der Kinder bestenfalls vage Vorstellungen von Informatiksystemen hat, aber sehr gut in der Lage ist, ein altersangemessenes System wie den Calliope mini zielgerichtet zu programmieren.

Keywords: Grundschule; Informatik; Calliope mini

1 Motivation und Ziele der Studie

Informatiksysteme halten mehr und mehr Einzug in den Lebensalltag aller Menschen, darunter auch SchülerInnen der Primarstufe. Doch im Gegensatz zu Phänomenen der natürlichen Umwelt, über die der Sachkundeunterricht größtenteils aufklärt, sind Systeme der digitalen Welt noch nicht Gegenstand des Unterrichts in der Primarstufe. Im Gegensatz zu anderen Ländern gibt es in Deutschland zur Zeit weder ein eigenes Unterrichtsfach noch eine feste Verankerung dieser Themen in den Lehrplänen der bestehenden Grundschulfächer (vgl. [ST11]). Das Calliope-Mini-Projekt des Ministeriums für Bildung und Kultur (MBK) Saarland ist ein erster Schritt, um dies zu ändern: Seit Anfang 2017 werden Lehrerfortbildungen vom Landesinstitut für Pädagogik und Medien im Saarland angeboten, die den Mikrocontroller Calliope mini als didaktisches Informatiksystem zum Einsatz in der Schule vorstellen und Grundkonzepte zur informatischen Bildung in der Primarstufe vermitteln. Das Gerät soll nach Vorstellung des MBK ab Klassenstufe 3 saarlandweit im

¹ Universität des Saarlandes, Campus, DE-66123 Saarbrücken, [kevin.baum|kerstin.reese|pascal.schmidt|verena.wolf]@uni-saarland.de, [s8nakirs|s8luwach]@stud.uni-saarland.de

Regelunterricht eingesetzt werden [LP17].

Ziel der hier beschriebenen Studie ist es, die Vorkenntnisse von GrundschülerInnen zu beleuchten und Calliope mini auf seine Tauglichkeit bei der Einführung von Algorithmen und der Funktionsweise (reaktiver) Informatiksysteme im Regelunterricht zu untersuchen. Das sind wichtige Kompetenzen, die auch von der GI für den Primarbereich gefordert werden (vgl. [Be18]). Die Forschungsfragen, die hierzu geklärt werden sollen, lassen sich auf zwei wesentliche Hauptfragen A und B einschränken.

- A Gelingt es Kindern der 4. Klasse nach einer dreistündigen Einführung in Algorithmen und die Programmierung des Calliope mini, eigenständig Programme zur Lösung vorgegebener Eingabe-Ausgabe-Zusammenhänge zu entwerfen?

Daraus ergeben sich folgende Teilfragen:

- A1 Nutzen die Kinder zielgerichtet algorithmische Grundbausteine (Eingaben, Ausgaben, Sequenzen) sowie Sensoren und Aktoren?
A2 Lassen sich typische Missverständnisse oder Verständnisschwierigkeiten identifizieren?

Fragestellung B bezieht sich primär auf das Vorwissen der Lernenden:

- B Erkennen Kinder ein ihnen unbekanntes Informatiksystem als solches (und nicht nur als Spielzeug oder allgemeines technisches Gerät) und wie beschreiben sie ein solches System?

Hier werden drei Teilfragen unterschieden:

- B1 Welche Bestandteile und Funktionsweisen eines beobachteten Informatiksystems nennen die Kinder?
B2 Beschreiben die Kinder einen zutreffenden kausalen Zusammenhang zwischen Eingabe und Ausgabe?
B3 Falls die Kinder einen solchen Zusammenhang hergestellt haben: Welche Ursache schreiben die Kinder dem genannten Zusammenhang zu?

In der Studie wurde zunächst das Vorwissen von Viertklässlern ermittelt. Danach wurde eine fünfständige Unterrichtseinheit zu algorithmischen Grundbausteinen und dem Mikrocontroller Calliope mini durchgeführt. Die Unterrichtseinheit endete mit einer Aufgabenstellung, für die die Kinder das Eingabe-Verarbeitung-Ausgabe-Prinzip (EVA-Prinzip) nachvollziehen und mit dem Mikrocontroller umsetzen mussten.

Unsere Ergebnisse zeigen, dass ein Großteil der Kinder bestenfalls vage Vorstellungen von Informatiksystemen hat, aber sehr gut in der Lage ist, ein altersangemessenes System wie den Calliope mini zielgerichtet zu programmieren. Er ermöglicht den Kindern einen ersten Zugang zum EVA-Prinzip und wird erfolgreich von ihnen zur Problemlösung eingesetzt.

2 Organisation der Studie

Die Studie wurde von September bis Dezember 2018 in acht Klassen an fünf saarländischen Grundschulen mit insgesamt 137 Kindern durchgeführt (Tab. 1). Die nachfolgend beschrie-

Nr.	1	2	3	4	5	6	7	8	Σ
Schulcode	1	1	1	2	3	4	4	5	
Anzahl SchülerInnen	20	17	21	18	17	15	18	11	137

Tab. 1: Anzahl der an der Studie teilnehmenden Schulen, Klassen und SchülerInnen

bene Unterrichtseinheit hat einen Umfang von drei Doppelstunden. Unterrichtet wurde im Klassenraum mit einem Team von zwei Personen, wobei die Lehrperson der Klasse zusätzlich anwesend war. Das erforderliche technische Equipment – Calliope mini, Beamer, Laptops – wurde vom Projektteam vorbereitet und mitgebracht. Auf den Laptops wurde mithilfe eines lokalen Web-Servers eine Offline-Version des Microsoft MakeCode-Editors [Mi] betrieben, um von einem Internetzugang unabhängig zu sein. Der Calliope mini war als Download-Ordner des Web-Browsers konfiguriert, so dass beim Klick auf den Knopf „Herunterladen“ sofort der angeschlossene Calliope mini bespielt wurde.

Um die Vorstellungen der Kinder von Informatiksystemen zu untersuchen, wurden Cubelets benutzt [Mo]. Die über Magnete zusammensteckbaren Cubeletmodule hatten sich in Vortests bewährt, weil sie nicht direkt als „Computer“ erkennbar sind. Cubeletmodule sind Blöcke mit unterschiedlichen Funktionen, wie z.B. Motoren oder Sensoren. Damit stellen sie ein abstraktes, programmierbares Informatiksystem dar, das den Kindern recht wahrscheinlich nicht bereits bekannt war.

3 Verlauf der Unterrichtseinheit

Erste Doppelstunde: Interviews, Wortfeldarbeit und AIGo. Zur Exploration des Vorwissens wurden alle SchülerInnen zunächst einzeln mithilfe eines standardisierten Leitfadens interviewt: Nach einer kurzen Demonstration der Funktionsweise eines Cubelet-Aufbaus mit Motor- und Sensorblock – das Erkennen eines Gegenstandes im Nahbereichs des Sensors setzt den Motor für wenige Sekunden in Bewegung – wurden die SchülerInnen aufgefordert, das Gesehene zu beschreiben bzw. zu erklären.

Die im Verlauf der Interviews gesammelten Begriffe wurden auf Kärtchen notiert und im Plenum nacheinander an die Tafel geheftet, ggf. geklärt und anhand verschiedener Kriterien (z.B. Verhalten vs. Bestandteile des Roboters) sortiert. Neben einer gemeinsamen (sprachlichen) Ausgangsbasis wurde dadurch eine erste Sensibilisierung für die im Hinblick auf das EVA-Prinzip notwendige Unterscheidung zwischen Ein- und Ausgaben geschaffen. Die Einführung algorithmischer Grundkonzepte erfolgte zunächst unplugged, um das gleichzeitige Auftreten konzeptioneller und technischer Herausforderungen zu verhindern und um eine Ablenkung durch rein technische Probleme zu verhindern. Dafür wurde das Algorithmenspiel Algo! verwendet, welches lediglich LEGO-Zubehör und Befehlskarten mit Piktogrammen nutzt. Die einfachste Aufgabenvariante besteht darin, eine Spielfigur entlang eines Weges zu steuern, indem der passende Algorithmus mithilfe von elementaren Anweisungen eines begrenzten Befehlssatzes beschrieben wird. Nach einer kurzen Kennenlernphase bearbeiteten die SchülerInnen in Zweiergruppen mit jeweils eigenem

Spielset unterschiedlich komplexe Szenarien (z.B. auch die Umkehraufgabe: Wegebau zu vorgegebenem Algorithmus).

Zweite Doppelstunde: Calliope ohne und mit Strom. In der ersten Hälfte der Doppelstunde lernten die Kinder die Bestandteile der (zunächst noch stromlosen) Minis kennen und ordneten diese den Kategorien „Eingabe“, „Ausgabe“ und „Sonstiges“ zu. Anschließend erkundeten sie die Funktionsweise der Geräte: Hierzu wurden vier unterschiedlich programmierte Sorten von Minis (Kombinationen aus Eingabe (Knopf A gedrückt, geschüttelt) und Ausgabe (Smiley-Folge, Ampel-Farben)) an die SchülerInnen verteilt und jeweils ein prototypischer Vertreter in eine Raumecke platziert. Die SchülerInnen mussten sich nun so im Klassenraum positionieren, dass in den Ecken nur gleich funktionierende Minis (d.h. gleiche Kombination von Ein- und Ausgabe) zusammenkamen. In der anschließenden Besprechung wurde herausgearbeitet, dass sich das unterschiedliche Verhalten der Minis durch unterschiedliche, von Menschen gemachte Programmierung erklärt – damit war eine Motivation zum Kennenlernen der MakeCode-Entwicklungsumgebung gegeben.

Die Präsentation der Entwicklungsumgebung und die Programmierung eines Mini wurden zunächst am Beamer erläutert und anschließend von den SchülerInnen in Zweiergruppen am eigenen Rechner nachvollzogen. Nach erfolgreicher Reproduktion eines zweiten Beispiels durften die Kinder die vorgegebenen Programme verändern und weitere Befehle aus den Bereichen „Grundlagen“, „Eingabe“ und „Musik“ ausprobieren.

Dritte Doppelstunde: Abschlussaufgaben und Projekte. In der dritten Doppelstunde bearbeiteten die Kinder in Zweiergruppen vier Programmieraufgaben mit ansteigendem Schwierigkeitsgrad. Alle Programme sind mit sequenziellen Befehlen lösbar, wie auf der ersten Stufe des Stufenmodells von [HWW17]. Die SchülerInnen wurden nicht darüber informiert, dass es sich um „Testaufgaben“ handelt, um eine möglichst natürliche und ungezwungene Atmosphäre zu schaffen. Im Einzelnen sollten folgende Aufgaben bearbeitet werden:

Aufgabe 1: Reproduktion einer bereits bekannten Programmierung („Auf Knopfdruck A: Smileys anzeigen“).

Aufgabe 2: Transfer durch leichte Variation („Auf Knopfdruck B: Text „CODE“ anzeigen“).

Aufgabe 3: Vorlage eines Calliope mini, dessen Verhalten die Kinder zunächst erkunden und dann in Form eines Programms rekonstruieren sollten („Auf Knopfdruck A: längere Sequenz von unterschiedlichen Ausgaben“).

Aufgabe 4: Aufgabenstellung wie Aufgabe 3, bis auf komplexeres Verhalten des Calliope mini (drei unterschiedliche Inputs, die jeweils unterschiedliche Outputfolgen auslösen).

Die Bearbeitung der Abschlussaufgaben beanspruchte durchschnittlich 50 Minuten. In der restlichen Zeit durften die SchülerInnen selbstständig eigene Projekte realisieren und diese am Ende der Doppelstunde vor der gesamten Klasse präsentieren. Die Auswertung des freien Programmierens ist nicht Teil dieser Veröffentlichung.

ID	Kategoriebezeichnung	Definition	Ankerbeispiel
OK 1	Verhalten		
UK 1.1	kann Verhalten reproduzieren	SchülerIn wird selber aktiv, handelt und reproduziert das gezeigte Verhalten	
UK 1.2	kann Verhalten beschreiben	SchülerIn beschreibt das Beobachtete	„Wenn man die Hand dahin hält, fährt er nach links“, Interview-Nr. 59
OK 2	Begriffe		
UK 2.1	benutzt mystische Begriffe	Aussagen, in denen Begriffe auftreten, die vermenschlichen oder eine höhere Gewalt hinzuziehen	„Es hat Angst, dass es kaputt geht und weiß, dass es weglaufen soll“, Interview-Nr. 61
UK 2.2	benutzt nicht-mystische, aber nicht-technische Begriffe	Aussagen, in denen Begriffe aus der Biologie oder Analogien mit dem menschlichen Körperbau, z.B. Augen, beinhaltet sind	„Die Augen reagieren auf die Hand“, Interview-Nr. 64
UK 2.3	benutzt technische Begriffe	Aussagen, in denen Begriffe, die für ein technisches System angemessen sind, z.B. Räder, Sensoren, beinhaltet sind	„Da ist ein Sensor drin, der Bewegung wahrnimmt“, Interview-Nr. 82
OK 3	Weiterführend/Spezieller		
UK 3.1	gibt Antwort auf „Warum“-Frage	Antwort auf die „Warum“-Frage, die für den Schüler/die Schülerin sinnvoll ist	„Da ist was drin, das dafür sorgt, dass es sich bewegt“, Interview-Nr. 96
UK 3.2	erwähnt, dass ein Mensch das so gebaut/programmiert hat	Aussagen, in denen erwähnt wird, dass ein Mensch das Informatiksystem gebaut/programmiert hat	„Jemand hat sich was ausgedacht und wollte es umsetzen“, Interview-Nr. 93
UK 3.3	verwendet den Begriff „Programmierung“	SchülerIn kennt das Wort und verwendet es im Interview	„Das ist so programmiert“, Interview-Nr. 116
UK 3.4	hat eine Erklärung dafür, was „Programmierung“ ist	Auf Nachfrage: Erklärung des Begriffs „Programmierung“, die für den Schüler/die Schülerin sinnvoll ist	„Man kann ihm sagen, was es machen soll“, Interview-Nr. 103

Tab. 2: Kategoriensystem zur Einordnung der Aussagen in den Interviews

4 Erhebungsinstrumente und Auswertungsmethoden

Während der Interviews zur Vorwissensabfrage wurden Reaktionen und Antworten der Kinder von einem Protokollanten in Form von schriftlichen Notizen festgehalten.

Bei den Programmieraufgaben wurden die Aktionen am Computer per Screencast erfasst. Die Screencasts wurden mit der Software OBS [Op] angefertigt, die automatisch mit dem Editor startete. Während der Bearbeitung der Programmieraufgaben wurden Rückfragen der Kinder beantwortet, ggf. auch Hilfestellungen und Tipps geben – beides wurde mit Zuordnung zu dem/der betreffenden Schüler(in) und Zeitstempel notiert. Ton- oder Videoaufzeichnungen wurden nicht angefertigt.

Die Interviews wurden in Anlehnung an Mayring (vgl. [Ma02]) ausgewertet, da lediglich Notizen zu den Schülerantworten und -reaktionen vorlagen und keine vollständigen Transkriptionen. Tab. 2 zeigt das Kategoriensystem, welches induktiv entwickelt und in Rückkopplung mit seiner Anwendung mehrfach angepasst wurde.

Da die Notizen kurz und die Definitionen und Ankerbeispiele bereits sehr konkret sind, wurde auf Codierregeln verzichtet. Das Kategoriensystem wurde von mehreren Personen zur unabhängigen Auswertung genutzt. Im nächsten Schritt ließen sich die Unterkategorien „mystische“ (UK 2.1), „nicht-technische“ (UK 2.2), „technische“ (UK 2.3) Begriffe, die Antwort auf die „Warum-Frage“ (UK 3.1) und die Erklärung des Begriffs „Programmierung“ (UK 3.4) in Anlehnung an Mayring reduzieren.

Bei der quantitativen Auswertung der Screencasts ist für jedes Team und jede Aufgabe bestimmt worden, ob die Aufgabe vollständig, bis auf kleine Mängel oder gar nicht gelöst wurde. Ebenso wurde die Bearbeitungsdauer erfasst. Aufgabe 4 wurde aufgrund ihrer Komplexität genauer ausgewertet. Um diese Aufgabe erfolgreich zu lösen, musste sie in Teilaufgaben zerlegt werden. Der Erfolg beim Erkennen und Bearbeiten der Teilaufgaben und das Vorgehen der Schülerinnen und Schüler wurden festgehalten.

5 Ergebnisse der Studie

5.1 Ergebnisse der Interviewauswertung

Es lagen 133 Notizen zu Interviews vor, die ausgewertet werden konnten (Tab. 3). Als mystische Begriffe (UK 2.1) wurde z.B. „Jedi-Fahrzeug“ genannt. Es wurde unterstellt, dass die Cubelets „denken“ oder „sich erschrecken“. Zusammenfassend kann gesagt werden, dass diese Gruppe von Kindern die Cubelets als handelnd beschreibt. Neun Kinder haben im Interview mystische Begriffe verwendet.

Die Reduktion der nicht technischen Begriffe (UK 2.2) ergab, dass diese Gruppe von Kindern das eingesetzte Informatiksystem häufig mit Analogien aus der Biologie beschreibt. Die Cubelets haben „Augen“ oder die „Sonne löst etwas aus“. Meist werden Begriffe für den sichtbaren Teil der Sensoren gesucht: „Löcher“, „das Schwarze“ oder „Punkte“. 41 Kinder haben wenigstens einen nicht-technischen Begriff benutzt.

An technischen Begriffen (UK 2.3) wurden meist „Sensor“ oder Analogien („Scanner“, „Bewegungsmelder“) genannt. Die sichtbaren, mechanischen Bestandteile der Cubelets wurden öfter mit Begriffen wie „Räder“ oder „Rollen“ beschrieben. Innerhalb der Cubelets wurden „Kabel“, „Motoren“, „Elektronik“ und „Batterien“ vermutet. Manche Kinder dieser Gruppe erklärten die berührungsfreie Interaktion mit den Cubelets mit „Magneteten“ oder „Wärme“. Einige sagten, dass „Signale“ oder „Daten“ vom Sensor an den Motor geschickt werden. Insgesamt kann der Gebrauch der technischen Begriffe als überwiegend vage bezeichnet werden. 115 Kinder benutzten technische Begriffe.

Als Antwort auf die „Warum“-Frage (UK 3.1) wurde meist genannt, dass das System „so programmiert“ oder etwas „so eingebaut“ sei. Als Erklärung wurde oft nur gesagt, dass es einen „Sensor“ gibt. Die Antworten blieben kurz und kamen über Schlagworte – „Technik“, „Elektronik“, „Signale“ – kaum hinaus. Oft wurde erst auf Nachfrage eine Antwort gegeben. 66 Kinder gaben eine Antwort auf die „Warum“-Frage.

Wenn die Kinder den Begriff „Programmieren“ benutzt haben, wurde nachgefragt, was das sei (UK 3.4). Die Kinder erläuterten, dass „ein Mensch etwas sagt“ oder „etwas eingestellt wird“. Das macht man „am Computer“ und „überträgt“ es auf das Informatiksystem. 24

Schulcode	1	1	1	2	3	4	4	5	Σ
Anzahl Interviews	17	17	21	18	17	15	17	11	133

Tab. 3: Anzahl der verwertbaren Interviews

	bearbeitet	vollst. gelöst	kl. Fehler	nicht gelöst	min	∅	max
Aufg. 1	63	58	4	1	01:00	03:19	19:31
Aufg. 2	59	57	1	1	00:38	04:48	17:59
Aufg. 3	62	47	10	5	03:23	13:38	37:12
Aufg. 4	54	9	0	45	13:18	23:09	36:29

Tab. 4: Anzahl der bearbeiteten Aufgaben, Erfolg und Bearbeitungszeiten

Kinder hatten eine eher vage Vorstellung davon, was Programmieren ist. Kein Kind schien von eigenen Erfahrungen berichten zu können.

5.2 Ergebnisse aus der Screencast-Auswertung

Nach der Erhebungsphase lagen 63 auswertbare Screencasts mit insgesamt mehr als 50 Stunden Videomaterial vor. Tab. 4 zeigt den nach Aufgaben aufgeschlüsselten Erfolg der Bearbeitung sowie die Bearbeitungsdauer.

Festzustellen ist, dass bei den ersten drei Aufgaben jeweils über 90% der Gruppen eine korrekte Lösung oder eine Lösung mit nur kleinen Fehlern gefunden haben. Zur erfolgreichen Bearbeitung der 3. Aufgabe mussten die Teams ein komplexeres Verhalten des Calliope mini nachbilden. Dazu musste das Verhalten der Vorlage genau beobachtet und Schritt für Schritt programmiert werden. Nahezu 75% der Teams konnten diese Aufgabe vollständig lösen, während nur ca. 10% keine Lösung gelang.

Bei der 4. Aufgabe mussten mehrere Eingabe-Ausgabe-Zusammenhänge erkannt und implementiert werden. Besonders gut gelang es den Kindern, eine rein visuelle Sequenz zu erkennen und zu reproduzieren: Zwei Drittel konnten die korrekte Reihenfolge für das Lauflicht auf dem LED-Feld programmieren.

Zur Bestimmung der Bearbeitungsdauer der einzelnen Aufgaben (vgl. Abb. 1) wurde die Zeit gemessen, die entsprechend der Screencasts an der Lösung der Aufgabe verbracht wurde. Da die Aufgaben ohne Vorplanung lösbar waren und im Unterricht keine Planungszeit ohne Aktivität am Computer beobachtet werden konnte, gehen wir davon aus, dass die ermittelten Bearbeitungszeiten in Tab. 4 ausreichend genau sind. Zu erkennen ist, dass die Bearbeitungsdauer mit ansteigendem Schwierigkeitsgrad zunehmend stärker variiert. Das lässt vermuten, dass die erste Aufgabe als Einstieg den meisten Kindern gerecht wurde.

Die in einzelnen Fällen sehr geringe Bearbeitungszeit bei Aufgabe 2 ergibt sich daraus, dass nur wenige Teams den Baustein „Zeige Zeichenfolge“ benutzt haben. Die meisten Teams haben jeden Buchstaben einzeln auf dem LED-Feld gezeichnet, womit diese Umsetzung länger dauerte.

Gut 87% der Teams waren während (mehr oder minder) der gesamten Bearbeitungszeit mit den Aufgaben beschäftigt. Die Teams, die nicht an der Lösung gearbeitet haben, schauten sich an anderer Stelle des Editors um. Kein Team hat den Editor verlassen, andere Programme aufgerufen oder die Arbeit am Computer komplett eingestellt.

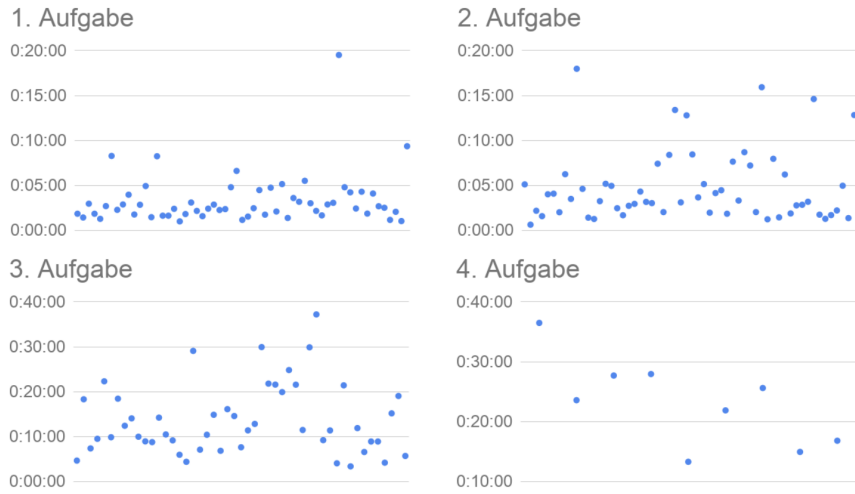


Abb. 1: Dauer der erfolgreichen Bearbeitung für jede Aufgabe

Die 4. Aufgabe bestand aus drei Teilaufgaben, die grundsätzlich in beliebiger Reihenfolge abgearbeitet werden konnten:

Teilaufgabe 1: Dauerhaft blinkendes Muster auf dem LED-Feld

Teilaufgabe 2: Wenn Knopf A gedrückt wird, werden 3 verschiedene Töne gespielt

Teilaufgabe 3: Wenn Knopf B gedrückt wird, werden 3 verschiedene Töne gespielt und der Ampelfarbverlauf (Rot, Gelb, Grün) dargestellt

Fast alle Teams bearbeiteten als erstes Teilaufgabe 1, die auch am häufigsten korrekt gelöst worden ist (gut 60% aller Teams). Bei den Teilaufgaben 2 und 3 sank der Anteil der erfolgreichen Teams auf ca. ein Viertel.

5.3 Beobachtungen während der Arbeit an den Abschlussaufgaben

Gelegentlich fehlten Teile des zu entwickelnden Algorithmus, wie z.B. das Ausschalten des LED-Felds am Ende des ersten Programms. Insbesondere bei Aufgabe 3 fehlte oft der finale Ton, da der Calliope mini nicht sehr laut piept. Es fiel einigen Kindern schwer, selbstständig neue Bausteine, wie z.B. den „dauerhaft“-Baustein für Aufgabe 4 oder den „Wenn Knopf B gedrückt“-Baustein für Aufgabe 2, zu integrieren. Wir vermuten, dass dies an der Kürze der Arbeit mit dem Editor lag und Probleme dieser Art mit zunehmender Übung seltener auftreten dürften.

Nur wenige Kinder verwendeten bei Aufgabe 3 auf Anhieb auch das Ausschalten des LED-Felds, um es zum Blinken zu bringen. Fast alle versuchten das Blinken mit dem „Pausiere“-Baustein umzusetzen. Dies wurde möglicherweise durch die Programmierung der Ampel in der Einführung begünstigt, bei der man diesen Baustein benötigt. Zudem könnte der Bezeichner dazu geführt haben, dass die Kinder in den „Pausiere“-Baustein eine andere Semantik hinein interpretierten ("tue eine Zeit lang nichts"), von der sie auch nur schwer abzubringen waren. Die Verwendung eines leeren „Zeige LEDs“-Bausteins in Verbindung mit dem „Pausiere“-Baustein überzeugte die Kinder deutlich mehr.

6 Fazit und abschließende Worte

Die Ergebnisse der Studie zeigen einerseits die geringen Vorkenntnisse der interviewten SchülerInnen und offenbaren zugleich das Potential kindgerechter reaktiver Informatiksysteme – hier in Form des Calliope mini – für das Erlernen informatischer Grundprinzipien. Ein Großteil der 137 TeilnehmerInnen verschiedener Schulen in unterschiedlichen Einzugsgebieten hatte bislang allenfalls rudimentäres Vorwissen über Informatiksysteme. Eine adäquate Vorstellung der Funktionsweise des EVA-Prinzips konnte nahezu überhaupt nicht festgestellt werden.

Die Ergebnisse der abschließenden Programmierarbeit zeigen, dass die Kinder mit den einfachen EVA-Problemstellungen, wie sie in den Aufgaben 1-3 zu lösen waren, kaum Probleme hatten. Dies zeigt auch, dass sie im Laufe der Unterrichtseinheit die Kompetenz erworben haben, mit elementaren Programmierelementen und dem Editor sinnvoll umzugehen. Die 4. Aufgabe erforderte eine Zerlegung in überschaubare Teilprobleme, also eine durchaus fortgeschrittene Kompetenz des "computational thinking". Insofern ist es bewerkenswert, dass neun Teams (ca. 14%) diese Aufgabe vollständig korrekt lösen konnten und fast alle anderen Teams immerhin korrekte Teilergebnisse bei der Wahl der Ein-Ausgabe-Bausteine und der Definition der Sequenzen angeben konnten.

Insgesamt ziehen wir hieraus das Fazit, dass SchülerInnen der 4. Klasse in der Lage sind, einfache Beispiele für das EVA-Prinzip zu verstehen und in einem konkreten Informatiksystem zu implementieren – wie es auch [Sc01] aus kognitionspsychologischer Sicht bestätigt. Der Calliope mini stellte sich als gutes Hilfsmittel zur Entmystifizierung solcher Systeme heraus und ist den Ergebnissen der Studie nach zu urteilen als didaktisches Informatiksystem zum Einsatz in der Primarstufe geeignet. Zu sehr ähnlichen Schlussfolgerungen kommt auch eine Studie von Murmann et al. (vgl. [Mu18]), in der ebenfalls Unterrichtseinheiten für Calliope mini entworfen und durchgeführt worden sind. Während der Bremer Fokus auf Beobachtungen und Befragungen der Kinder lag, die mit den Calliope minis arbeiteten, beinhaltet unsere Studie auch die Auswertung eines Abschlusstests und fokussiert damit auch die Performanz der SchülerInnen. Eine weitere Studie mit der Programmiersprache Scratch findet sich in [FGH17].

Abschließend möchten wir einige persönliche Eindrücke schildern: Viele Lehrkräfte waren im Vorfeld skeptisch und zeigten gegenüber den Geräten Berührungängste. Manche konnten sich kaum vorstellen, dass ihre SchülerInnen im Umgang mit den Minicomputern

etwas Sinnvolles lernen könnten. Die Unterrichtseinheit hat dieses Bild deutlich verändert, weil sie den Kindern Spaß machte und ihre Ergebnisse die Erwartungen der Lehrkräfte schon nach kurzer Zeit deutlich übertrafen – eine Erkenntnis, die bei möglichst vielen Grundschullehrkräften erreicht und gefestigt werden muss.

Danksagung Wir danken Ludger Humbert (Universität Wuppertal), der uns in der Vorbereitungsphase für die Studie unterstützt hat. Ebenso danken wir den beteiligten LehrerInnen und SchülerInnen für ihre Bereitschaft, an der Studie teilzunehmen.

Literatur

- [Be18] Best, A.; Borowski, C.; Büttner, K.; Freudenberg, R.; Fricke, M.; Haselmeier, K.; Herper, H.; Hinz, V.; Humbert, L.; Müller, D.; Thomas, M.: Kompetenzen für informatische Bildung im Primarbereich. LOG IN 38/189/190, S. 1–36, 2018.
- [FGH17] Funke, A.; Geldreich, K.; Hubwieser, P.: Analysis of scratch projects of an introductory programming course for primary school students. In: IEEE Global Engineering Education Conference (EDUCON). S. 1229–1236, Apr. 2017.
- [HWW17] Hoffmann, S.; Wendlandt, K.; Wendlandt, M.: Algorithmisieren im Grundschulalter. In: Informatische Bildung zum Verstehen und Gestalten der digitalen Welt. Gesellschaft für Informatik, Bonn, S. 73–82, 2017.
- [LP17] LPM: Bildungsminister Commerçon stellt Landeskonzept Medienbildung und Förderprogramm für Medienschulen vor, 2017, URL: https://www.lpm.uni-sb.de/typo3/index.php?id=2092&tx_ttnews%5Btt_news%5D=679&cHash=fc7e178c61bb18f79092d46e441cccf0&L=0, Stand: 18.05.2019.
- [Ma02] Mayring, P.: Einführung in die qualitative Sozialforschung. Beltz, 2002.
- [Mi] Microsoft: MakeCode, URL: makecode.calliope.cc, Stand: 18.05.2019.
- [Mo] Modular Robotics: Cubelets, URL: www.modrobotics.com, Stand: 18.05.2019.
- [Mu18] Murmann, P. D. L.; Schelhowe, P. D. H.; Bockermann, D. I.; Engelbertz, S.; Illginnis, S.; Moebus, A.: Calliope mini - Eine Explorationsstudie im pädagogisch-didaktischen Kontext, Abschlussbericht, 2018, URL: <https://elib.suub.uni-bremen.de/edocs/00106848-1.pdf>, Stand: 18.05.2019.
- [Op] Open Broadcaster Software: OBS Studio, URL: <https://obsproject.com>, Stand: 18.05.2019.
- [Sc01] Schwill, A.: Ab wann kann man mit Kindern Informatik machen? - Eine Studie über informatische Fähigkeiten von Kindern. In: Informatikunterricht und Medienbildung. INFOS '01, GI, S. 13–30, 2001.
- [ST11] Starruß, I.; Timmermann, B.: Informatische Bildung in Deutschland. LOG IN 31/1, S. 49–59, 2011.

Bild der Informatik von Grundschullehrpersonen

Ergebnisse eines mehrjährigen Projekts zu informatikbezogenen Vorstellungen

Alexander Best¹

Abstract: Grundschullehrpersonen ($n=11$) wurden mittels leitfadengestützter Einzelinterviews zu ihren Vorstellungen zur Informatik und zum Informatikunterricht befragt. Die Analyse der Transkripte deckte vier Sichtweisen auf Informatik auf: eine mathematische, mediengeprägte, gesellschaftliche und technische. Die Lehrpersonen gehen von einer dichotomen Einteilung in *Informatik-Insider* sowie *Informatik-Outsider* aus und haben eine funktional geprägte Sichtweise auf digitale Artefakte (*black box*). Informatiksysteme nehmen einen hohen Stellenwert ein; es werden aber auch *Unplugged*-Ansätze expliziert. Fachliche Vorstellungen beschränken sich auf sämtliche GI-Prozessbereiche und die GI-Inhaltsbereiche INFORMATION UND DATEN, ALGORITHMEN sowie INFORMATIK, MENSCH UND GESELLSCHAFT. Der Kontakt mit Informatik entstand in der eigenen Schulzeit, Freizeit, im Studium oder beruflichen Umfeld und führte zu unterschiedlichen biografischen Zugängen. Das selbstständige Unterrichten einer informatischen Unterrichtseinheit führte bei Lehrpersonen ($n=3$) sowohl zur Vorstellungsänderung (*conceptual change*) als auch Vorstellungsbeständigkeit und -resilienz (*belief perseverance* oder *conceptual conservatism* bzw. *robustness*), welche in abschließenden Einzelinterviews erhoben wurden.

Keywords: Informatik in der Grundschule; Primarbildung; Bild der Informatik; Lehrervorstellungen; Einzelfallstudien; Leitfadeninterviews; *Grounded Theory*

1 Einleitung

Das Projekt „Informatik in der Grundschule (IGS)“ am Arbeitsbereich Didaktik der Informatik der WWU Münster wurde 2014 initiiert und an ein Dissertationsvorhaben gekoppelt. Die Forschungsfragen lauten:

- RQ1* Welche biografischen Bezüge zur Informatik und zum Informatikunterricht werden von Grundschullehrpersonen assoziiert?
- RQ2* Welche zentralen Begriffe der Informatik sind nach Auffassung der Lehrpersonen für die Grundschule von Bedeutung; warum sind sie es und wie sind diese ihrer Ansicht nach inhaltlich belegt?
- RQ3* Wo begegnen Kinder im Grundschulalter nach Ansicht der Grundschullehrpersonen Informatik in ihrem Alltag?

¹ Westfälische Wilhelms-Universität Münster, Didaktik der Informatik, Corrensstraße 80, 48149 Münster, alexander.best@uni-muenster.de

RQ4 Welche implizit-unbewussten informatikbezogenen Vorstellungen manifestieren sich beim Unterrichten einer informatischen Unterrichtseinheit und inwiefern kommt es hierbei zu Vorstellungsänderung, -beständigkeit und/oder -resilienz?

Auf der INFOS 2015 wurden erste Forschungsergebnisse vorgestellt, welche durch einen hypothesenprüfenden Zugang mittels eines Fragebogens gewonnen wurden [BM15]. Grundlage bildete ein Datensatz von 27 Lehrpersonen des Faches Sachunterricht an 13 Grundschulen. Auf der INFOS 2017 wurden Zwischenergebnisse vorgestellt, die über einen hypothesengenerierenden Zugang mittels leitfadengestützter Einzelinterviews gewonnen wurden [Be17]. Grundlage bildeten zwei kontrastierende Einzelfälle. In diesem Beitrag werden die abschließenden Ergebnisse des Vorhabens vorgestellt, welche zwischen 2014 und 2019 gewonnen wurden. Die Datenerhebung war auf Grundschulen in öffentlicher Trägerschaft im Regierungsbezirk Münster (NRW) beschränkt. Die Population besteht aus 6.469 Lehrpersonen ($\sigma^{\wedge}=534$) an 411 Schulen².

2 Forschungsstand

Lehrpersonen besitzen sowohl fachliche, allgemeindidaktische als auch fachdidaktische Kenntnisse. Besonders Shulman zeigte in seinen Arbeiten, dass Drittes als „besonderes Amalgam“ dient, um Erstes und Zweites zu verknüpfen [Sh87]. Hierauf basierend entwickelte Ernest ein Modell für die Mathematikdidaktik, in dem er neben Kenntnissen (*knowledge*) auch Vorstellungen (*beliefs*) und Einstellungen (*attitudes*) von Lehrpersonen als bedeutsame Faktoren für professionelles Handeln herausstellte und in Anlehnung an Thompsons Vorarbeiten drei Sichtweisen auf Mathematik auswies: *problem-solving view*, *Platonist view* und *instrumentalist view* [Er89]. Das Interesse der Forschung an Lehrervorstellungen stieg zunehmend und erhielt mit Pajares Aufsatz „*Teachers' Beliefs and Educational Research: Cleaning Up a Messy Construct*“ von 1992 ein Manifest, welches die multidisziplinären Ergebnisse und Erkenntnisse bündelte und systematisierte. Für die Fachdidaktik Informatik gelangt Bender zu zahlreichen „professionellen Überzeugungen“³ von Informatiklehrpersonen [Be16]. Lewis, Jackson und Waite zeigen, dass sich Vorstellungen und Einstellungen von Hochschuldozierenden (*faculty*) und Studierenden (*students*) gleichen, unterscheiden, annähern und auseinanderentwickeln [LJW10]. Dengel weist in einer Studie mit 116 Informatiklehrpersonen an unterschiedlichen Schulformen der Sekundarstufe nach, dass eine Mehrheit frühe informatische Bildung sowohl für möglich als auch sinnvoll erachtet [De17]. Funke, Geldreich und Hubwieser führten eine Studie mit sechs Grundschullehrpersonen ($\sigma^{\wedge}=5$) durch [FGH16]. In dieser kommen sie zum Ergebnis, dass die interviewten Lehrpersonen zwar über „kein konkretes Bild der Informatik in der

² Angaben laut Statistischem Landesamt NRW, Landesdatenbank-Codes *B-D06.1-II* und *B-D09.1-II*, Schuljahr 2017/18. Angaben exklusive PRIMUS-Schulen, Freie Waldorfschulen und sonstige Privatschulen.

³ Kirchner weist darauf hin, dass die Begriffe „Überzeugung“ sowie „Vorstellung“ in der Bildungsforschung weitestgehend synonym verwendet werden und empfiehlt die Verwendung des letzteren [Ki16].

Grundschule“, jedoch über einige förderliche Vor- und Einstellungen verfügen [FGH16]. Die Ergebnisse decken sich mit denen der auf der INFOS 2015 und 2017 vorgestellten Beiträge [BM15], [Be17]. Die Veränderbarkeit, Beständigkeit bzw. Resilienz sowie Differenzierung in bewusste und unbewusste Lehrervorstellungen stellen bislang ein Forschungsdesiderat der Fachdidaktik Informatik für sämtliche Bildungsbereiche dar.

3 Theoretischer Rahmen

Kirchner definiert: „Lehrervorstellungen sind subjektive, relativ stabile, wenngleich erfahrungsbasiert veränderbare, zum Teil unbewusste, kontextabhängige Kognitionen von Lehrpersonen. Sie umfassen die theorieähnlichen, wenn auch nicht widerspruchsfreien Gedanken zu verschiedenen fächerübergreifenden und fachspezifischen Gegenstandsbereichen der Profession von Lehrpersonen.“ [Ki16]. Eine Übersicht weiterer Definitionen geben Fives und Buehl [FB12]. Über Anteile von Affektion, Motivation und Volition an Vorstellungen sowie deren Wechselwirkung mit weiteren mentalen Merkmalen, etwa Einstellungen, wird derzeit aktiv geforscht.

3.1 Eigenschaften von Lehrervorstellungen

Aus den Katalogen von Pajares, Fives und Buehl sowie Kirchner lassen sich folgende Eigenschaften zusammentragen [Pa92], [FB12], [Ki16]: Lehrervorstellungen können (1) explizit, also bewusst, oder implizit, also unbewusst, sein, (2) fungieren als Filter auf die Wahrnehmung, (3) haben individuelle Bedeutung und Funktion im Vorstellungssystem⁴, (4) sind entgegen wissenschaftlicher Theorien weder zwingend logisch noch widerspruchsfrei, (5) haben Einfluss auf das Verhalten von Lehrpersonen inner- und außerhalb des Unterrichts, (6) haben Einfluss auf die Lernprozesse von Schülerinnen und Schülern⁵ sowie deren Vorstellungen, (7) lassen sich nicht distinkt von Wissen trennen und (8) sind kontextuell. Die Mehrheit der Forscher geht davon aus, dass Lehrervorstellungen veränderbar sind, jedoch leicht und schwer veränderbare Vorstellungen existieren. Zwei Kriterien hierfür stellen die Zeitspanne sowie Relevanz oder Zentralität einer Vorstellung im Vorstellungssystem dar.

3.2 Systematisierung von Lehrervorstellungen

Taxonomien lassen sich u. a. bei Fives und Buehl, Kunter und Pohlmann sowie Calderhead finden:

⁴ Bryan definiert: „*Belief systems are groups of beliefs that, unlike knowledge systems, do not require general consensus. They are relatively static, and when they change it is not because of sound reasoning but more likely because of a »conversion or gestalt shift«*“ [Br03] in Anlehnung an Nespor.

⁵ „Schülerinnen und Schüler“ bzw. „Schülerinnen und Schülern“ wird im Folgenden mit SuS abgekürzt.

Domänen von Lehrervorstellungen nach...

<i>Fives und Buehl [FB12]</i>	<i>Kunter und Pohlmann [KP15]</i>	<i>Calderhead [Ca96]</i>
<ul style="list-style-type: none"> • Selbst • Kontext oder Umgebung • Inhalt oder Wissen • spezifische Unterrichtskonzepte • Lehransätze • Lernende 	<ul style="list-style-type: none"> • Selbst • bestimmte SuS • Lehren und Lernen 	<ul style="list-style-type: none"> • Selbst und die Lehrerrolle • Lernende und das Lernen • Lehren • Lernen zu Lehren • Fach

Für die Fachdidaktik Informatik gelangt Bender zu folgender Systematisierung [Be16]:
 (1) Lehr-lerntheoretische und epistemologische Überzeugungen in Bezug auf Informatik,
 (2) Überzeugungen in Bezug auf den Umgang mit Daten und (3) Überzeugungen zum Lernen im Kontext von Strategien und Prinzipien.

4 Methodik und Methodologie

Die Ergebnisse des auf der INFOS 2015 vorgestellten Beitrags zeigen große Unsicherheiten bei den Lehrpersonen, sich deduktiv vorgegebenen Aussagen über Informatik und Informatikunterricht zuzuordnen [BM15]. Fischler problematisiert: „Fragebogen besitzen nicht die Flexibilität, die notwendig ist, um situationsbezogene Vorstellungen so auszuloten, dass der Zusammenhang von erklärten Handlungszielen und darauf abgestimmten Handlungen sichtbar wird. Außerdem öffnen sie durch die Vorgabe von Begriffen, über deren Bedeutung im Falle verschiedener Interpretationen wegen Fehlens eines Dialogs zwischen Fragendem und Befragtem kein Konsens hergestellt werden kann, ein weites Feld möglicher Missverständnisse.“ [Fi01]. Methodologischer Rahmen für die Neuausrichtung des Vorhabens bildete die *Grounded Theory (GT)*, welche bereits in mehreren Arbeiten der Fachdidaktik Informatik erfolgreich genutzt werden konnte [Kn11], [Mü17]. Die Wahl des Datenerhebungsinstruments fiel auf das leitfadengestützte Einzelinterview. Es zeichnet sich über den Einsatz eines Gesprächsleitfadens und sogenannter Check-Up-Fragen durch ein mittleres Maß an Strukturiertheit aus, was besonders Novizen den Einstieg in die Interviewführung erleichtert. Die Interviews wurden digital-auditiv aufgezeichnet und mittels Regelsatz transkribiert sowie pseudonymisiert. Daran schloss sich die Analyse der Transkripte über das dreischrittige Kodier-Schema der *GT* an: offene, selektive und axiale Kodierung. Die Zusammensetzung der Stichprobe (*sample*) erfolgte mittels theoretischem Sampling. Hierfür wurden nach Auswertung eines Interviews Lehrpersonen identifiziert, die das Potential kontrastiver (maximale Ausprägung) oder ähnlicher (minimale Ausprägung) Vorstellungen vermuten ließen. Grundlage für diese Auswahl bildeten zumeist quantitative Daten, die zuvor mittels eines

Interviewakquisebogens erhoben wurden. Neben diesem Zugriff auf die expliziten Vorstellungen der Lehrpersonen sollte auch eine Erhebung der impliziten Vorstellungen erfolgen.⁶ Als eine Möglichkeit, auf Letztere zuzugreifen, schlagen Fives und Buehl vor, (geplantes) Verhalten und Gespräche der Lehrpersonen zu beobachten und auf zugrundeliegende Vorstellungen zu schließen (*infer*) [FB12]. Einige der zuvor interviewten Lehrpersonen ($n=3$) erklärten sich bereit, eine informatische Unterrichtseinheit unter Hospitation des Forschers durchzuführen und anschließend an einem Abschlussinterview teilzunehmen. Während der Hospitation wurde ein Protokoll angefertigt. Die Durchführung und Auswertung der Abschlussinterviews gestaltete sich analog zu den Eingangsinterviews.

5 Vorstellungen von Grundschullehrpersonen zur Informatik

Es zeigten sich vier Sichtweisen auf Informatik, die nur geringe Überschneidungen bei den interviewten Lehrpersonen aufwiesen. Eine Sonderrolle kommt der mediengeprägten Sichtweise zu, denn der Medienbegriff wurde von zahlreichen Lehrpersonen zur Kompensation fehlender Fachlichkeit oder Fachsprache situationsbezogen herangezogen, auch wenn sie nicht genuin dieser Sichtweise zugeordnet werden können.

Mediengeprägte Sichtweise: Informatik ist für diese Lehrpersonen eng mit Informatiksystemen verknüpft. Diese nehmen jedoch in ihren Vorstellungen unterschiedliche Funktionen ein: Für manche Lehrpersonen führt bereits die Nutzung von Informatiksystemen zur Assoziation mit Informatik, wobei sie sich stark auf den Medienbegriff stützen.⁷ Andere erwarten eine Gestaltungs- oder Bewertungskomponente.

Mathematische Sichtweise: Diese Lehrpersonen zeigen im Interview Parallelen zwischen Informatik und Mathematik auf. Insbesondere der Algorithmenbegriff nimmt für sie eine wichtige Funktion ein. Rechenstrategien oder die Auseinandersetzung mit Sachaufgaben werden mit diesem verknüpft. Eine Lehrerin sieht sich durch die Teilnahme am Informatik-Biber in ihrer diesbezüglichen Vorstellung bestätigt.

Technische Sichtweise: Diese Lehrpersonen verbinden mit Informatik eine strukturelle Perspektive auf digitale Artefakte [Sc08], bei der sie sich auf Hardware- (bspw. Datenspeicher) oder Softwarekomponenten (bspw. algorithmische Grundbausteine) beziehen. Insbesondere Datenspeicherung, -sicherheit und -schutz werden von ihnen

⁶ Fives und Buehl stellen fest: „*We contend that teachers hold both implicit and explicit beliefs that influence their teaching practice. Attempts to access teachers' implicit beliefs may well bring these conceptions into the explicit realm, thereby changing the nature of the beliefs under examination.*“ [FB12].

⁷ Dies deckt sich mit Borowskis Aussage: „Es gibt zwei entscheidende Gründe, warum sich ein Ansatz der Informatik in der Grundschule mit der Medienbildung und Medienpädagogik auseinandersetzen muss. Der eine besteht darin, dass bei Lehrerinnen und Lehrern häufig informatische Bildung und Medienbildung synonym benutzt wird.“ [Bo14].

genannt. Besonders der GI-Inhaltsbereich INFORMATION UND DATEN wird von diesen Lehrpersonen mit Informatik assoziiert. Während der Datenbegriff aus dem Mathematikunterricht bekannt ist, wird Information meist mit der sogenannten „Informationsrecherche“ im Netz assoziiert.

Gesellschaftliche Sichtweise: Informatik setzt sich für diese Lehrpersonen mit Fragen der Auswirkungen auseinander, die durch die zunehmende gesellschaftliche Durchdringung mit Informatiksystemen entstehen. Die Einstellungen der Lehrpersonen sind durchaus differenziert und reflektiert („keine Rose ohne Dornen“). Sie gehen davon aus, dass sowohl förderliche als auch gefährdende Auswirkungen existieren. Genanntes Beispiel für Erstes bildet der Einsatz von Informatiksystemen in der Medizin, die Vernichtung von Arbeitsplätzen im Niedriglohnssektor durch die Automatisierung von Abläufen für Letztes.

Obwohl die interviewten Lehrpersonen ähnliche Biografien aufweisen, hat der Kontakt mit Informatik sie unterschiedlich geprägt und zu unterschiedlichen Vorstellungen geführt, welche wiederum unterschiedliche Funktion im informatischen Vorstellungssystemen einnehmen. Dies gilt insbesondere für Lehrpersonen, die Informatikunterricht in der eigenen Schulzeit belegten. Während eine Lehrperson bereits Überlegungen angestellt hatte, wie sie Themen aus ihrem damaligen Informatikunterricht auf den Unterricht in der Grundschule übertragen könne, führte bei einer anderen Lehrperson der eigene, als negativ empfundene, Informatikunterricht dazu, dass Informatik im Studium, im Referendariat und auch im Berufsalltag gemieden wurde und weiterhin wird.

6 Vorstellungen von Grundschullehrpersonen zum Informatikunterricht

Informatische Bildung, im Sinne der Vorstellungen der Lehrpersonen zur Informatik, wird als bedeutsam für die Primarstufe und darüber hinaus auch für die Lebens- und zukünftige Berufswelt der SuS erachtet. Der Ort bzw. die Form informatischer Bildung zeichnet sich hierbei jedoch sehr heterogen ab: (1) eigenständiges Fach, (2) fächerintegriert in mehrere Fächer, (3) fachintegriert in ein Fach oder (4) Arbeitsgemeinschaft/Neigungsgruppe. Die Lehrpersonen gehen davon aus, dass SuS bereits über das Elternhaus und *peer-groups* informatikaffin oder nicht-informatikaffin sind. Sie sehen eine Verstärkung dieser Differenzierung in *Informatik-Insider* und *Informatik-Outsider* [Kn11] mit zunehmendem Alter. Sie gehen davon aus, dass Jungen höheres Interesse an Informatik haben als Mädchen. Eine Lehrperson differenzierte hier jedoch zwischen einer qualitativen und quantitativen Begegnung mit Informatiksystemen, die bei Jungen häufig in den letztgenannten Bereich falle. Die Lehrpersonen sind überzeugt, dass genderspezifischen Sozialisationseffekten zur Informatik in den unteren Klassen der Grundschule oder gar in der Elementarbildung

entgegengewirkt werden müsse. Informatikunterricht wurde von sämtlichen Lehrpersonen mit kooperativen Sozial- und Aktionsformen assoziiert.

7 Ergebnisse

Zu *RQ1*: Der Bezug zur Informatik sowie zum Informatikunterricht entsteht überwiegend in der eigenen Schulzeit. Dies geschieht entweder durch die Wahl, Abwahl oder Nicht-Wahl des Informatikunterrichts. Die durch den Informatikunterricht gewonnenen Vorstellungen sind häufig durch einen Theorie-Praxis-Konflikt gekennzeichnet. Zudem beschränken sich diese meist auf einzelne Werkzeuge oder Gegenstände („Programmiersprache C“, „Diskettenlaufwerk“ etc.). Die Einstellungen der Lehrpersonen gegenüber dem Informatikunterricht in der Sekundarstufe sind negativ oder indifferent, was jedoch nur bei einer Lehrperson zur Ablehnung informatischer Bildung in der Grundschule führt.

Zu *RQ2*: Insbesondere die GI-Prozessbereiche sehen die Lehrpersonen als bedeutsam für die Primarbildung an. Mehrfach äußern sie jedoch Befürchtungen, dass, etwa im Fall von BEGRÜNDEN UND BEWERTEN, eine Überforderung der SuS eintreten könne. DARSTELLEN UND INTERPRETIEREN sowie STRUKTURIEREN UND VERNETZEN werden häufig auf den Einsatz von Informatiksystemen zur Unterstützung von Lernprozessen beschränkt. Die Lehrpersonen beziehen sich auf die GI-Inhaltsbereiche ALGORITHMEN, INFORMATION UND DATEN sowie INFORMATIK, MENSCH UND GESELLSCHAFT, deren Bedeutung für die Primarbildung je nach Sichtweise unterschiedlich gewichtet wird. Lediglich Lehrpersonen, die der mediengeprägten Sichtweise zugeordnet werden können, haben bei der Erläuterung der GI-Inhaltsbereiche durchgängig Schwierigkeiten, da ihnen hier der Medienbegriff fehlt. Vorstellungen zu INFORMATIKSYSTEME sowie SPRACHEN UND AUTOMATEN existieren nicht – zumindest nicht unter diesen Bezeichnungen. Die Relevanz der GI-Kompetenzbereiche wird anhand des (1) Alltagsbezugs, (2) des Allgemeinbildungsanspruchs und (3) der Berufsvorbereitung festgemacht.

Zu *RQ3*: Der Kontakt von Kindern mit Informatik wird von den interviewten Lehrpersonen überwiegend über Informatiksysteme wahrgenommen. Hierzu zählt sowohl der bewusste Kontakt (*Phänomenbereich 1*), aber auch der unbewusste Kontakt (*Phänomenbereich 2*) [HP04]. Insbesondere diejenigen Lehrpersonen, die eine mathematische Sichtweise einnehmen, gehen auch auf Kontakte der SuS zur Informatik ohne Informatiksysteme (*Phänomenbereich 3*) ein. Mit zunehmendem Alter wird ein häufigerer und bewussterer Kontakt von SuS mit Informatik wahrgenommen. Darüber hinaus sind die Lehrpersonen der Ansicht, dass genderspezifische Sozialisierungseffekte in den unteren Klassen der Grundschule weniger stark ausgeprägt seien und eine informatische Bildung deshalb bereits hier einsetzen müsse.

Zu *RQ4*: Die Hospitationen ($n=3$) zeigen, dass die Lehrpersonen ihre zuvor in den Interviews explizierten Vorstellungen auch auf Handlungen und Gespräche im

Unterricht übertragen. Vorstellungsänderungen treten insbesondere bei denjenigen Lehrpersonen auf, die eine starke Verknüpfung zwischen Informatik und Informatiksystemen in den Interviews explizieren und eine *Unplugged*-Einheit unterrichten. Eine Lehrperson mit hoher Berufserfahrung zeigt erkennbare Vorstellungsbeständigkeit bzw. –resilienz; obwohl sie die Durchführung einer *Unplugged*-Unterrichtseinheit zur Kryptologie als motivierend für die SuS erachtet, hält sie an ihrer Vorstellung fest, dass Informatikunterricht zu theoriegeladen sei.

8 Implikationen

Anders als bislang vermutet, verfügen die interviewten und beobachteten Grundschullehrpersonen über heterogene informatikbezogene Vorstellungen. Eine Gefahr besteht darin, dass die Lehrpersonen selektiv informatische Inhalte oder Prozesse aufgreifen, die ihre Vorstellungen bestätigen, und solche, die ihren Vorstellungen widersprechen, zurückzuweisen. Lehrpersonen mit einem noch sehr vagen und undifferenzierten Bild der Informatik neigen dazu, spezifische Themen der Informatik als umfassend aufzufassen. Dies könnte dazu führen, dass Informatik bspw. mit Programmierung oder Kryptologie gleichgesetzt wird. Folgende Empfehlungen können aus den gewonnenen Erkenntnissen abgeleitet werden:

Initiativen für die tertiären und quartären Bildungsbereiche sollten...

- (1) ...sämtliche GI-Prozessbereiche und GI-Inhaltsbereiche abdecken, sodass ein differenziertes Bild der Informatik vermittelt werden kann.
- (2) ...förderliche Vorstellungen zur Informatik, etwa den Bezug zu Algorithmen über Ablaufbeschreibungen, Anleitungen und Rätsel, aufgreifen, um die Wahrnehmung für *Unplugged*-Ansätze zu sensibilisieren.
- (3) ...für eine vorgeschaltete, formative, summative und nachbereitende Evaluation informatikbezogener Vorstellungen genutzt werden.
- (4) ...Ansätze und Methoden, welche auf die „Diskreditierung“ resilienter Vorstellungen ausgelegt sind, überprüfen.

9 Ausblick

Erste Ergebnisse und Erkenntnisse konnten im Grundschullehrseminar „Digitale Medien und informatische Bildung (DiMeiB)“ an der WWU Münster integriert werden. In Zukunft sollen vorhandene, sich verändernde und beständige bzw. resiliente Vorstellungen in den Fokus genommen werden. Zudem wird die Forschung auf weitere Regierungsbezirke ausgeweitet und zu einem Mixed-Methods-Ansatz erweitert.

10 Literaturverzeichnis

- [Be16] Bender, E.: Modellierung und Dimensionierung der professionellen Überzeugungen und motivationalen Orientierungen als Aspekte professioneller Kompetenz von Informatiklehrkräften. Kumulative Dissertation, Paderborn, 2016.
- [Be17] Best, A.: Bild der Informatik von Grundschullehrpersonen. Erste Ergebnisse aus qualitativen Einzelfallstudien. In (Diethelm, I. Hrsg.): Informatische Bildung zum Verstehen und Gestalten der digitalen Welt. Köllen Druck+Verlag, Bonn, 2017; S. 83–86.
- [BM15] Best, A.; Marggraf, S.: Das Bild der Informatik von Sachunterrichtslehrern. Erste Ergebnisse einer Umfrage an Grundschulen im Regierungsbezirk Münster. In (Gallenbacher, J. Hrsg.): Informatik allgemeinbildend begreifen. Köllen Druck+Verlag, Bonn, 2015; S. 53–62.
- [Bo14] Borowski, C.: Thesen zur Informatik in der Grundschule. <https://www.grundschulinformatik.de/thesen-zur-informatik-in-der-grundschule/>, 18.05.2019.
- [Br03] Bryan, L. A.: Nestedness of beliefs: Examining a prospective elementary teacher's belief system about science teaching and learning. In *Physical Review Special Topics - Physics Education Research*, 2003, 40; S. 835–868.
- [Ca96] Calderhead, J.: Teachers: Beliefs and Knowledge. In (Berliner, D. C.; Calfee, R. C. Hrsg.): *Handbook of educational psychology*. Macmillan Library Reference USA, New York, 1996; S. 709–725.
- [De17] Dengel, A.: Opinions of CS Teachers in Secondary School Education about CS in Primary School Education. In (Barendsen, E.; Hubwieser, P. Hrsg.): *Proceedings of the 12th Workshop in Primary and Secondary Computing Education*. ACM Press, New York, 2017; S. 97–98.
- [Er89] Ernest, P.: The Knowledge, Beliefs and Attitudes of the Mathematics Teacher: a model. In *Journal of Education for Teaching*, 1989, 15; S. 13–33.
- [FB12] Fives, H.; Buehl, M. M.: Spring cleaning for the “messy” construct of teachers’ beliefs. What are they? Which have been examined? What can they tell us? In (Harris, K. R.; Graham, S.; Urdan, T. Hrsg.): *APA Educational Psychology Handbook. Volume 2. Individual Differences and Cultural and Contextual Factors*. American Psychological Association, Washington (D.C.), 2012; S. 471–499.

- [FGH16] Funke, A.; Geldreich, K.; Hubwieser, P.: Primary School Teachers' Opinions about Early Computer Science Education. In (Sheard, J.; Suero Montero, C. Hrsg.): 16th Koli Calling International Conference on Computing Education Research. Koli Calling 2016. ACM Press, New York, 2016; S. 135–139.
- [Fi01] Fischler, H.: Verfahren zur Erfassung von Lehrer-Vorstellungen zum Lehren und Lernen in den Naturwissenschaften. In *Zeitschrift für Didaktik der Naturwissenschaften*, 2001, 7; S. 105–120.
- [HP04] Humbert, L.; Puhlmann, H.: Essential Ingredients of Literacy in Informatics. In (Schubert, S.; Magenheimer, J. Hrsg.): *Informatics and Student Assessment. Concepts of Empirical Research and Standardisation of Measurement in the Area of Didactics of Informatics*. Köllen Druck+Verlag, Bonn, 2004; S. 65–76.
- [Ki16] Kirchner, V.: *Wirtschaftsunterricht aus der Sicht von Lehrpersonen. Eine qualitative Studie zu fachdidaktischen teachers' beliefs in der ökonomischen Bildung*. Springer VS, Wiesbaden, 2016.
- [Kn11] Knobelsdorf, M.: *Biographische Lern- und Bildungsprozesse im Handlungskontext der Computernutzung*. Dissertation, Berlin, 2011.
- [KP15] Kunter, M.; Pohlmann, B.: Lehrer. In (Möller, J.; Wild, E. Hrsg.): *Pädagogische Psychologie*. Springer, Berlin, 2015; S. 261–281.
- [LJW10] Lewis, C.; Jackson, M. H.; Waite, W. M.: Student and Faculty Attitudes and Beliefs About Computer Science. In *Communications of the ACM*, 2010, 53; S. 78–85.
- [Mü17] Müller, D.: *Der Berufswahlprozess von Informatiklehrkräften*. Überarbeitete Fassung der Dissertation, Universität Wuppertal, 2016. Universitätsverlag Potsdam, Potsdam, 2017.
- [Pa92] Pajares, M. F.: Teachers' Beliefs and Educational Research: Cleaning Up a Messy Construct. In *Review of Educational Research*, 1992, 62; S. 307–332.
- [Sc08] Schulte, C.: Die duale Natur digitaler Artefakte als Kern Informatischer Bildung. In (Thomas, M.; Weigend, M. Hrsg.): *Interesse wecken und Grundkenntnisse vermitteln*. 3. Münsteraner Workshop zur Schulinformatik. ZfL-Verlag, Münster, 2008; S. 7–24.
- [Sh87] Shulman, L. S.: Knowledge and teaching: Foundations of the new reform. In *Harvard Educational Review*, 1987, 57; S. 1–22.

Analyse von Curricula auf Abdeckung der Kompetenzen zur Bildung in der digitalen Welt

Ira Diethelm,¹ Sebastian Glücks²

Abstract: Die Kultusministerkonferenz hat Ende 2016 die Strategie zur Bildung in der digitalen Welt [Ku16] verabschiedet hat, mit der sich die Länder verpflichten dafür zu sorgen, dass Schüler*innen zum Ende ihrer Pflichtschulzeit 61 Kompetenzen besitzen. In dieser Studie wurden daher alle Curricula der Sekundarstufe I, die für niedersächsische Gymnasien im Schuljahr 2017/18 in Kraft waren, auf Abdeckung dieser Kompetenzen untersucht. Dabei wurde sowohl der Frage nachgegangen, welche Kompetenzen bereits jetzt in welchen Fachcurricula enthalten sind als auch, welche Kompetenzen sich vglw. leicht in Bezug auf die Unterrichtsthemen oder die Ausbildung der Lehrkräfte in welche Fächer aufnehmen ließen³. Die Rolle des Faches Informatik wird bei dieser Analyse sehr deutlich. Sie zeigt, dass mindestens ein Drittel der KMK-Kompetenzen keinem anderen Fach zuzuordnen sind und je nach Stundenumfang eine Aufnahme von bis zu zwei Dritteln der Kompetenzen in das Fach Informatik sinnvoll erscheinen. Die Analyse zeigt auch, welche anderen Fächer noch Potenzial aufweisen, welche aber auch nicht, so dass hieraus gut abzulesen ist, welche Curricula überarbeitet werden sollten. Die hier erstellten Zuordnungen und Veränderungsvorschläge sollen keine bundesweite Empfehlung sein. Sie stellen nicht mehr und nicht weniger als eine erste Untersuchung dar. Sie können aber einen wertvollen Ausgangspunkt für Schulentwicklungsgespräche mit Kollegien und erste Orientierung für andere Bundesländer und Schulformen sein und so helfen mit einem konstruktiven Prozess die Relevanz der Informatik im Kollegium zu verdeutlichen.

Keywords: Kompetenzen; digitale Welt; Curricula; Inhaltsanalyse; Rolle der Informatik

1 Ausgangslage und Fragestellung

Der Bildungsauftrag der allgemeinbildenden Schulen ist, die Schülerinnen und Schüler zu der aktiven und mündigen Teilhabe an der Gesellschaft zu befähigen. Mit der fortschreitenden Entwicklung der digitalen Technologien steigen auch die Anforderungen an diejenigen, die mit ihnen umgehen müssen. Ein kompetenter Umgang mit diesen setzt somit den fundierten Erwerb entsprechender Kompetenzen voraus. Dazu zählt neben dem Wissen um die Verwendung auch ein Grundverständnis für die informatischen Grundlagen. Für einen reflektierten Umgang mit Informatiksystemen und der Urteilsfähigkeit über sie sind diese Aspekte gleichermaßen bedeutend. Da das Zurechtfinden in der digitalen Welt inzwischen als Kulturtechnik neben dem Lesen, Schreiben und Rechnen angesehen wird, ist in Bezug auf den allgemeinbildenden Schulunterricht festzuhalten, dass den

¹ Carl von Ossietzky Universität, Didaktik der Informatik, 26111 Oldenburg ira.diethelm@uol.de

² Studienseminar Leer, Bürgerm.-Ehrlehnholz-Str. 15, 26789 Leer (Ostfriesland) sebastian.gluecks@uol.de

³ Dieser Artikel fasst die Masterarbeit [Gl18] zusammen, aus der viele Passagen z.T. wortwörtlich übernommen sind. Aus Gründen der Lesbarkeit wurde auf die umfangreiche Kenntlichmachung verzichtet.

Schülerinnen und Schülern während ihrer Schullaufbahn die benötigten Kompetenzen vermittelt werden müssen. Welche Kompetenzen dies genau sind, wurde schon vielfach diskutiert. Die Gesellschaft für Informatik hat dazu bereits im Jahr 2008 Empfehlungen für die Sekundarstufe I ausgesprochen [Ge08]. Auch seien die Empfehlungen der Länderkonferenz der MedienBildung [LK15] und der europäische DIGCOMP [FPB13] exemplarisch für solche Kompetenzlisten genannt.

Mit der Umsetzung der Strategie sollen laut KMK zwei wesentliche Ziele verfolgt werden: Das erste betrifft die Integration der aufgestellten Kompetenzen in den Unterrichtsalltag. Es wird angestrebt, die digitalen Kompetenzen in die Unterrichtsfächer zu integrieren. Als zweites Ziel formuliert die KMK, dass digitale Lernumgebungen vermehrt pädagogisch sinnvoll eingesetzt werden sollen. Auch dies erfordert einen abgestimmten Katalog an Kenntnissen und Fähigkeiten der Schülerinnen und Schüler, auf den sich die Lehrkräfte in den verschiedenen Fächern verlassen können, damit nicht in jedem Fach erneut erklärt werden muss, wie man beispielsweise an einem gemeinsamen Dokument arbeitet.

Den Anstoß für die hier vorliegende Untersuchung bildet die Anfrage eines niedersächsischen Gymnasiums, das ein Schulkonzept für die digitale Bildung erarbeiten will. Dabei sollen sowohl Pläne für die Infrastruktur und technische Ausstattung als auch für curriculare Vorgaben entstehen. Ähnliche Schulentwicklungsprozesse beschäftigen seit Veröffentlichung der KMK-Strategie viele Schulen bundesweit. Insofern ist die Anfrage der Schule nach entsprechenden Umsetzungskonzepten exemplarisch. Insbesondere stellen Schulen (und Länder) sich die folgenden zwei Fragen: *1. Inwiefern sind die geforderten Kompetenzen jetzt schon in den Curricula enthalten?* und *2. Welche Fächer können bzw. sollten welche Kompetenzen in ihre Curricula aufnehmen?*

Die KMK-Strategie [Ku16] schreibt zur Verteilung der Kompetenzen auf die Fächer: „Die Entwicklung und das Erwerben der notwendigen Kompetenzen für ein Leben in einer digitalen Welt gehen über notwendige informatische Grundkenntnisse weit hinaus und betreffen alle Unterrichtsfächer.“ Hier wird also festgestellt, dass es notwendige informatische Grundkenntnisse gibt, die zum Leben in der digitalen Welt erforderlich sind. Es wird aber auch gefordert, dass alle Fächer ihren Beitrag leisten müssen. Welche Fächer aber genau welchen Beitrag leisten müssen, wird im KMK-Dokument nicht spezifiziert. Informatik ist wie in Niedersachsen in vielen anderen Bundesländern kein Pflichtfach und erreicht daher dort nur einen Teil der Jahrgänge. Informatik kann daher in diesen Ländern zur Umsetzung des für alle Schülerinnen und Schüler verpflichtenden Kompetenzkatalogs der KMK nur eingeschränkt herangezogen werden. Die Frage nach der Verteilung der Kompetenzen ist damit auch eine Frage nach der Stellung des Schulfaches Informatik.

Als erste Schritte auf dem Weg zu einer umfassenden Antwort auf diese Fragen wurde daher eine Analyse aller gültigen Kerncurricula Niedersachsens für Gymnasium und die Sekundarstufe I. Dies entsprach einerseits dem Profil der anfragenden Schule, und andererseits ist dies genau der Bereich, in dem nach der ICILS13 [Bo14] die deutschen Schülerinnen und Schüler am besten abgeschnitten hatten. Daher ist anzunehmen, dass hier bei einer Analyse

ggf. die größte Übereinstimmung zu finden wäre, falls die ICILS-Ergebnisse auf die Lehrpläne zurückgeführt werden können. Die Analyse soll zum einen eine Bestandsaufnahme davon sein, welche dieser Kompetenzen bereits vermittelt werden müssen. Dazu werden im Folgenden das Vorgehen und die Ergebnisse beispielhaft am Fach Informatik und einem weiteren Fach dargestellt. Zum anderen werden mögliche Anpassungen vorgeschlagen, die die Umsetzung der KMK-Strategie befördern würden.

2 Vorgehen

Es wurden alle gymnasialen Kerncurricula der Sekundarstufe I Niedersachsens, die im Schuljahr 2017/18 in Kraft waren, zur Analyse herangezogen. Bei der Wahl der zu analysierenden Fächer wurde keine weitere Einschränkung vorgenommen. Die Analyse wurde in den Fächern auf die offiziellen Dokumente der Kerncurricula begrenzt und keine darüber hinaus gehende Literatur zur Interpretation der Curricula (wie z.B. Schulbücher) herangezogen. Für jedes Fach entstand während der Analyse eine Tabelle, in der jeder Kompetenz der KMK-Strategie die passenden fachspezifischen Kompetenzen zugeordnet wurden. Dabei wurde in zwei Kategorien unterschieden. Zur so definierten „vollen Abdeckung“ zählt eine Fachkompetenz, wenn diese nach Einschätzung der Autoren die Kompetenz zur digitalen Bildung vollständig umschließt. Dazu musste die folgende Frage mit „ja“ beantwortet werden: Wenn Schülerinnen und Schüler die betrachtete Fachkompetenz erworben haben, haben sie damit gleichzeitig auch die Kompetenz zur Bildung in der digitalen Welt erworben? Wurde diese Frage jedoch mit „nein“ beantwortet, wurde in abgeschwächter Form danach gefragt, ob wenigstens ein Teilaspekt der Kompetenz zur Bildung in der digitalen Welt erworben wurde. Fachkompetenzen, für die diese Frage mit „ja“ beantwortet wurde, wurden in die „teilweise Abdeckung“ einsortiert. Zu diesen teilweise abdeckenden Fachkompetenzen wurden darüber hinaus Veränderungsvorschläge erarbeitet, durch deren Umsetzung eine vollständige Abdeckung erreicht werden könnte. An dieser Stelle verschwimmt bereits die Ergebnisdarstellung mit der Deutung, da es sich bei den Veränderungsvorschlägen bereits um Schlussfolgerungen aus der Analyse handelt. Diese Darstellung wurde gewählt, damit die Information zusammenhängend dargestellt werden kann.

Einschränkend ist anzumerken, dass die Autor*innen die Schulfächer Informatik, Musik, Mathematik und Chemie vertreten und die Analyse daher in allen andere Fächern aus fachfremder Sicht durchgeführt wurde. Wie bei jeder qualitativen Inhaltsanalyse ergeben sich aus unterschiedlichen Einschätzungen verschiedener Personen Deutungsunterschiede, die bei einer weiterführenden Diskussion der Ergebnisse zu beachten sind. Dies wird durch die recht allgemeinen Formulierungen der Kompetenzen in der KMK-Strategie, die häufig nicht explizit auf die digitale Welt Bezug nehmen, noch verstärkt. So kann man Kompetenzen wie „Referenzierungspraxis beherrschen“ oder „eine Produktion planen“ auch als erfüllt ansehen, wenn in einem Kerncurriculum diese ebenfalls allgemein erwähnt werden und man diese Kompetenzen somit auch mit analogen Mitteln erreichen kann. Insofern soll hier betont werden, dass die nachfolgenden Ergebnisse und Veränderungsvorschläge *eine* mögliche

Deutung darstellen und nicht als endgültige Empfehlung, sondern als Ausgangspunkt für weitere Diskussionen zu verstehen sind. Ebenso sind die berechneten prozentualen Abdeckungsquoten zu verstehen: Es wurden zur Vereinfachung Spiegelstriche der Kompetenzen gezählt, obwohl verschiedene Kompetenzen natürlich durchaus unterschiedlichen Umfang im Unterricht einnehmen.

3 Ergebnisse

Hier werden nun exemplarisch die Ergebnisse der Analyse der fachspezifischen Curricula für Musik und Informatik dargestellt. Die vollständigen Ergebnisse für alle Fächer sind online verfügbar [G118]. Die Ergebnisse werden tabellarisch dargestellt, vgl. Abb. 1: In der ersten Zeile steht die betrachtete KMK-Kompetenz, die über eine Übereinstimmung in dem Unterrichtsfach verfügt. Fachspezifische Kompetenzen, die die KMK-Kompetenz vollständig abdecken (VA), sind in der zweiten Zeile aufgeführt. In der dritten Zeile, die gelb markiert ist, sind Fachkompetenzen aufgelistet, die die betrachtete KMK-Kompetenz teilweise abdecken (TA). Dies kann der Fall sein, wenn beispielsweise der direkte Bezug zu digitalen Umgebungen fehlt oder sie als nicht-verpflichtende Option genannt ist, ansonsten aber eine ähnliche Kompetenz zu Grunde liegt oder sie sich auf konkrete, fachspezifische Beispiele bezieht. Die vierte und damit letzte Zeile ist für Kommentare und Veränderungsvorschläge (VV) vorgesehen.

KMK	3.1.2. Eine Produktion planen und in verschiedenen Formaten gestalten, präsentieren, veröffentlichen oder teilen		
VA	3.2.1 Arbeitsfeld: Singen <ul style="list-style-type: none"> - tragen Sprechstücke artikuliert vor (Grundlagen) - singen einstimmige Lieder verschiedener Stil- und Herkunftsbereiche – auch auswendig (Grundlagen) - singen zweistimmige Lieder und Kanons in Kleingruppen und im Klassenverband (Grundlagen) - singen ein- und mehrstimmige Lieder nach Noten (Weiterführung) 3.2.2 Arbeitsfeld: Instrumentalspiel <ul style="list-style-type: none"> - notieren und spielen Melodien und Rhythmen (Grundlagen) - spielen Arrangements in Kleingruppen und im Klassenverband – auch nach Notation (Grundlagen) - erfinden und gestalten Musik nach einer Vorgabe (Weiterführung) 3.2.3 Arbeitsfeld: Bewegung <ul style="list-style-type: none"> - gestalten eine Szene mit Musik und Bewegung (Grundlagen) 	KMK 3.3.1. Bedeutung von Urheberrecht und geistigem Eigentum kennen TA 3.3.5 Arbeitsfeld: Musik in ihren historischen und gesellschaftlichen Bezügen - Thema: Urheberrecht (Erweiterung I)	
		KMK 3.3.2. Urheber- und Nutzungsrechte (Lizenzen) bei eigenen und fremden Werken berücksichtigen TA 3.3.5 Arbeitsfeld: Musik in ihren historischen und gesellschaftlichen Bezügen - Thema: Urheberrecht (Erweiterung I)	
VV	Im Bereich des Musikmachens ist der Vortrag ein wichtiger Bestandteil. Jeder musikalische Vortrag ist als Produktion im Sinne der KMK-Kompetenz zu werten, da eine Vorbereitung und Planung für eben jenen notwendig sind.		VV Das Urheberrecht wird lediglich als mögliche thematische Konkretisierung genannt. Es könnte einen höheren Stellenwert erhalten, indem entsprechende Kompetenzen zu dem Wissen um das Urheberrecht und dessen Einhaltung formuliert werden. Die digitalen Kompetenzen 3.3.1 und 3.3.2 können als Anhaltspunkt dienen.

Abb. 1: Vergleich des Curriculums für Musik mit der KMK-Strategie im Bereich 3

Wenn eine Kompetenz aus der KMK-Strategie sowohl vollständig als auch teilweise von unterschiedlichen Fachkompetenzen abgedeckt wird, so ist sie insgesamt als vollständig abgedeckt zu werten. Daraus resultiert, dass keine Veränderungsvorschläge für die teilweise abdeckenden Kompetenzen formuliert werden. Darüber hinaus finden sich in dieser Zeile Erklärungen zu Interpretationen, sofern diese für das Verständnis notwendig sind. Die Kompetenzbereiche, Lernfelder oder ähnlichen Kategorien, zu denen die Fachkompetenzen in den Curricula zusammengefasst wurden, wurden in die tabellarische Darstellung der Ergebnisse übernommen, um ein leichteres Auffinden der Kompetenzen in den Curricula zu ermöglichen. Diese Inhalte sind kursiv gedruckt, damit sie gut von den eigentlichen Fachkompetenzen unterschieden werden können.

3.1 Musik

Das Curriculum für das Fach Musik [Ni17] deckt in seiner jetzigen Form nach unserer Einschätzung fünf Kompetenzen aus dem Kompetenzrahmen zur Bildung in der digitalen Welt der KMK ab. Dies entspricht einer Abdeckungsquote von 8,2%. Weitere vier dieser Kompetenzen lassen sich in der teilweisen Abdeckung durch Fachkompetenzen finden. In Abb. 1 ist exemplarisch der KMK-Kompetenzbereich 3 *Produzieren und Präsentieren* dargestellt.

Der Kompetenzbereich 3.3 *Rechtliche Vorgaben beachten* lässt sich im Fach Musik bei der Thematisierung des Urheberrechts finden. Neben Bereich 3 sind auch im Kompetenzbereich 1 und 2 der KMK Übereinstimmung zu erkennen. Insbesondere ist eine Zuordnung der Kompetenz 2.5.2 *Medienerfahrungen weitergeben und in kommunikative Prozesse einbringen* zum Unterrichtsfach Musik im Kerncurriculum zu finden. Der Kompetenzbereich 6.1 *Medien analysieren und reflektieren* lässt sich im Fach Musik unter anderem im Bereich der Analyse der Wirkung von Musik wiederfinden. Auch gibt es Möglichkeit die Kompetenzen 6.1.3 *Wirkungen von Medien in der digitalen Welt (z.B. mediale Konstrukte, Stars, Idole, Computerspiele, mediale Gewaltdarstellungen)* analysieren und konstruktiv damit umgehen anhand von Komponisten- und Musikerportraits zu vermitteln.

3.2 Informatik

Das Kerncurriculum des Unterrichtsfachs Informatik [Ni14] enthält nach unserer Einschätzung passende Fachkompetenzen zu 20 der 61 Kompetenzen der KMK-Strategie und deckt damit momentan 32,8% der zu vermittelnden Kompetenzen zur Bildung in der digitalen Welt ab. Eine Erhöhung um weitere 26 Kompetenzen auf 75,4% ist mit Blick auf die Ausbildung der dafür nötigen Fachlehrkräfte zur Umsetzung der zuvor genannten Veränderungsvorschläge möglich. Zu den 15 verbleibenden Kompetenzen ließen sich keine passenden Fachkompetenzen aus dem KC Informatik ausmachen.

Im Kompetenzbereich 1.1 *Suchen und Filtern* steht die Verwendung von digitalen Ressourcen für Rechercharbeiten im Vordergrund. Die Vermittlung der Kompetenzen des Kompetenzbereichs 1.2 *Auswerten und Bewerten* wird durch die Kompetenzen des im KC Informatik sogenannten Lernfeldes *Daten und ihre Spuren* umfassend abgedeckt. Der Kompetenzbereich 3.1 *Speichern und Abrufen* wird durch Fachkompetenzen, die in Verbindung zu Datenbanken und weiteren Möglichkeiten der Strukturierung von Daten stehen, abgedeckt. In Bezug zu den Kompetenzen aus 2.1 *Interagieren* ließen sich die Themen Datenschutz und Datensicherheit behandeln. Die Kompetenzen zu 2.4 *Umgangsregeln kennen und einhalten (Netiquette)* ließen sich anhand der Thematisierung sozialer Netzwerke vermitteln. Zu diesem Kompetenzbereich lassen sich im Kerncurriculum in der aktuellen Form jedoch keine zugehörigen Kompetenzen identifizieren. Die Kompetenz 2.5.1 *Öffentliche und private Dienste nutzen* hingegen lässt sich in Ansätzen im Kerncurriculum wiederfinden.

Die Vermittlung der Kompetenzen aus dem Kompetenzbereich 3 *Produzieren und Präsentieren* kann ebenfalls durch den Unterricht im Fach Informatik umfassend sichergestellt werden. Für den Kompetenzbereich 4 *Schützen und sicher Agieren* lassen sich im Kerncurriculum passende Kompetenzen insbesondere aus dem Lernfeld Daten und ihre Spuren ausmachen. Hier erscheint eine Zusammenführung der Fachkompetenzen mit den KMK-Kompetenzen sinnvoll. Dies könnte als Ergänzung zum Kompetenzbereich *I4 Informatik und Gesellschaft* des Kerncurriculums geschehen. Der Kompetenzbereich 5 *Problemlösen und Handeln* lässt sich gänzlich in den Informatikunterricht integrieren. Neben dem Bereich 5.5 *Algorithmen erkennen und formulieren*, der umfangreich durch Fachkompetenzen abgedeckt ist, lassen sich auch Kompetenzen finden, für die weitgehende Anpassungen notwendig werden würden. Zu den Kompetenzen 5.1.1 *Anforderungen an digitale Umgebungen formulieren*, 5.2.2 *Anforderungen an digitale Werkzeuge formulieren*, 5.2.4 *Digitale Umgebungen und Werkzeuge zum persönlichen Gebrauch anpassen*, 5.4.1 *Effektive digitale Lernmöglichkeiten finden, bewerten und nutzen* und 5.4.2 *Persönliches System von vernetzten digitalen Lernressourcen selbst organisieren können* lassen sich in der aktuellen Form des Kerncurriculums derzeit keine passenden Fachkompetenzen ausmachen, obwohl diese Kompetenzen eindeutig dem Lernfeld *Computerkompetenz* zuzuordnen sind. Insofern könnte dieses Lernfeld um den Unterpunkt *Umgang mit digitalen Umgebungen und Werkzeugen* erweitert werden, in dem diese fünf Kompetenzen zusammengefasst werden.

Die Kompetenzen aus dem Bereich 6 *Analysieren und Reflektieren* lassen sich als Teilaspekte der beiden Fachkompetenzen *I 4.1 beschreiben die Auswirkungen des Einsatzes von Informatiksystemen auf die Gesellschaft* und *P 4.4 bewerten die Bedeutung eines Informatiksystems für das Individuum und die Gesellschaft* wiederfinden. Diese beiden Kompetenzen sind sehr umfangreich und allgemein formuliert, sodass die zugeordneten Kompetenzen aus der KMK-Strategie als Konkretisierungen angesehen werden können.

4 Zusammenfassung für alle Fächer

In der Übersichtsgrafik in Abb. 2 sind die Abdeckungsquoten der einzelnen Fächer aufgelistet. Die oberen, grün eingefärbten Balken stellen die prozentuale Abdeckung der Kompetenzen zur Bildung in der digitalen Welt der jeweiligen Fächer dar. Die Werte sind so zu verstehen, dass in dem jeweiligen Unterrichtsfach die KMK-Kompetenzen zu dem aufgeführten Prozentsatz bereits unterrichtet werden müssen. Das Unterrichtsfach Informatik hat mit 32,8% in der jetzigen Form des Curriculums nach unserer Einschätzung die größte fachspezifische Abdeckung. Die zweitgrößte Abdeckung ist beim Unterrichtsfach Deutsch mit 23,0% zu finden. Die geringste Abdeckung mit 1,6% erreicht derzeit das Fach Politik-Wirtschaft.

Die unteren, gelben Balken weisen die Abdeckungsquoten aus, die erreicht werden können, wenn die Veränderungsvorschläge umgesetzt werden würden. Das Fach Informatik erreicht weiterhin die höchste Quote mit 75,4%. Auch das Fach Deutsch behält den zweiten Platz in

der Rangfolge mit einer Erhöhung auf 31,1%. Der geringste Wert wäre dann beim Fach Geschichte mit 9,8% zu finden.

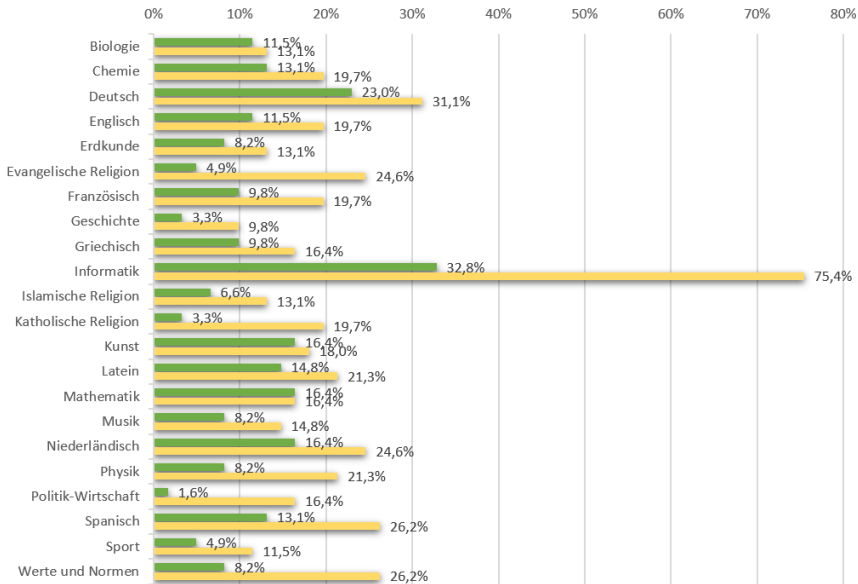


Abb. 2: Abdeckung der KMK-Kompetenzen durch die Fächer

Um eine Aussage darüber treffen zu können, wie viele Kompetenzen aus der KMK-Strategie insgesamt durch die vorhandenen Kerncurricula abgedeckt werden, muss eine Auszählung über mehrere Fächer durchgeführt werden. In der KMK-Strategie ist festgelegt, dass die Vermittlung der Kompetenzen alle SuS erreichen soll. Daher wurde ermittelt, ob dies in den Pflichtfächern möglich ist und welcher Anteil dort bereits vermittelt wird. Den Abbildungen 3 bis 5 kann entnommen werden, welche Kompetenzen in welchen Fächern zu finden sind. Darüber hinaus ist mit der Größe der farbigen Kästchen angedeutet, in welchem Umfang die entsprechende Kompetenz in dem jeweiligen Fach abgedeckt wird. Dazu wird zwischen zwei Fällen unterschieden. Sofern nur eine einzige passende Fachkompetenz gefunden wurde, ist ein kleines Kästchen dargestellt. Bei zwei oder mehr übereinstimmenden Fachkompetenzen wurde ein breites Kästchen eingefügt. Darüber hinaus kann an der Art der farblichen Füllung erkannt werden, ob es sich bei dem Unterrichtsfach um ein Pflichtfach handelt. Diese sind einfarbig ausgefüllt, während alle anderen Fächer eine gestreifte Struktur aufweisen. Zur Menge der Pflichtfächer werden die Unterrichtsfächer Biologie (BI), Chemie (CH), Deutsch (DE), Englisch (EN), Erdkunde (EK), Geschichte (GE), Kunst (KU), Mathematik (MA), Musik (MU), Physik (PH), Politik-Wirtschaft (PW) und Sport (SP) gezählt. Die Abkürzungen für die übrigen Fächer sind: Evangelische Religion (ER), Französisch (FR), Griechisch (GR), Informatik (IF), Islamische Religion (IR), Katholische Religion (KR), Latein (LA), Niederländisch (NI), Spanisch (SN) und Werte und Normen (WN). Außerdem ist an der Farbe zu erkennen, ob die KMK-Kompetenz, auf die die Nummer verweist,

vollständig (grün) oder teilweise (gelb) von dem entsprechenden Unterrichtsfach abgedeckt wird. Aus Platzgründen wurde auf die ausführliche Nennung der Kompetenzen aus der KMK-Strategie verzichtet und nur die Kompetenznummern Bezug genommen.

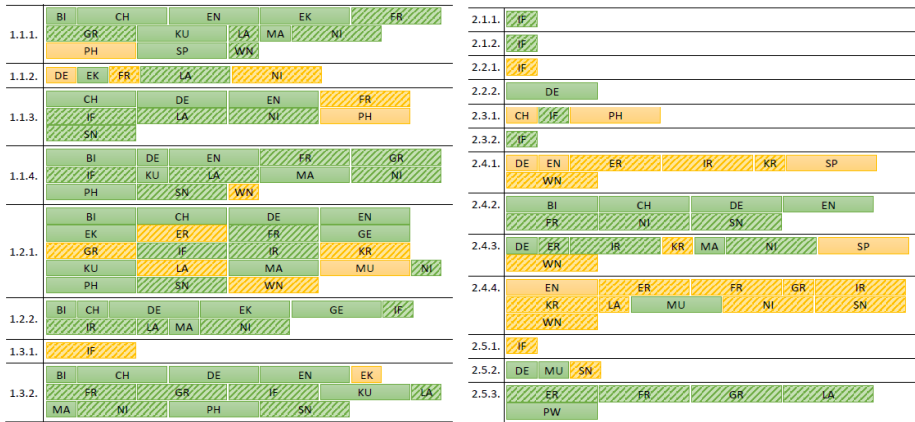


Abb. 3: Abdeckung der KMK-Bereiche 1 und 2

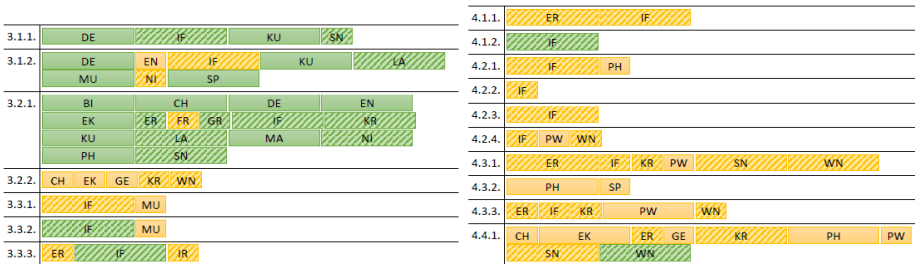


Abb. 4: Abdeckung der KMK-Bereiche 3 und 4

Der Kompetenzbereich 1 *Suchen, Verarbeiten und Aufbewahren* ist jetzt schon vergleichsweise gut in die vorhandenen Curricula integriert. Anders sieht es bei dem Kompetenzbereich 2 *Kommunizieren und Kooperieren*, 4 *Schützen und sicher Agieren* sowie 5 *Problemlösen und Handeln* aus. Insgesamt können 21 der 61 Kompetenzen bereits vollständig von den Pflichtfächern abgedeckt werden. Wird die teilweise Abdeckung hinzugerechnet, ist eine Abdeckung von 40 Kompetenzen möglich. Es bleiben allerdings weitere 21 Kompetenzen übrig, die auf diesem Weg nicht ohne weiteres in die Pflichtfächer integriert werden können. Es handelt sich hierbei um die Kompetenzen 1.3.1, 2.1.1, 2.1.2, 2.2.1, 2.3.2, 2.5.1, 3.3.3, 4.1.1, 4.1.2, 4.2.2, 4.2.3, 5.1.1, 5.1.2, 5.2.2, 5.2.3, 5.2.4, 5.3.1, 5.4.2, 5.5.2, 5.5.3 und 6.2.3. Gründe hierfür sind in erster Linie, dass diese Kompetenzen keine reinen Anwen- derkompetenzen sind, sondern alle ein tieferes Grundverständnis der zugrundeliegenden Funktionsweisen und Strukturen erfordern.

Bei der Betrachtung, welche Fächer diese übrigen Kompetenzen einschließen, fällt insbe- sondere das Fach Informatik auf. In diesem lassen sich alle aufgezählten Kompetenzen in

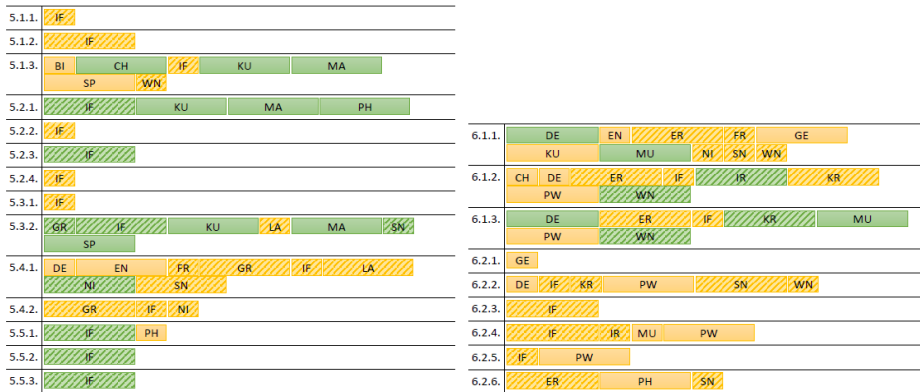


Abb. 5: Abdeckung der KMK-Bereiche 5 und 6

voller oder teilweiser Abdeckung wiederfinden oder leicht hinzufügen. Abbildung 6 zeigt dazu eine grafische Übersicht und damit die Relevanz des Pflichtfaches Informatik.

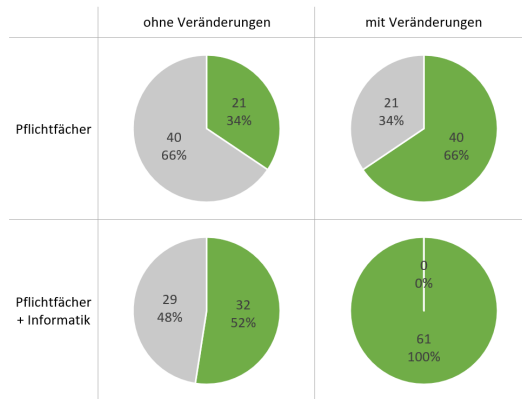


Abb. 6: Abdeckung der KMK-Kompetenzen durch Pflichtfächer mit und ohne Informatik

5 Schlussfolgerungen

Die in den vorangegangenen zwei Kapiteln ausführlich beschriebenen Ergebnisse der Untersuchung zeichnen ein erstes, aber deutliches Bild davon, wie die Umsetzung der KMK-Strategie zur Bildung in der digitalen Welt aussehen kann. Die hier dargestellte Analyse weist auf die Einstufung von Informatik als verpflichtendes Unterrichtsfach hin. Dadurch ließe sich sicherstellen, dass alle Schülerinnen und Schüler genau die informatischen Grundkenntnisse besitzen, die sie für den Erwerb der weiterführenden Kompetenzen benötigen. Diese Maßnahme ist dieser Untersuchung zufolge notwendig, um die verbleibende Lücke bei der Kompetenzvermittlung zu schließen. Gleichwohl lässt

sich auch erkennen, dass Informatik die Last für die Umsetzung der KMK-Strategie nicht allein tragen kann. Es braucht somit beides: Ein eigenes Schulfach für die systematische Bereitstellung der Grundkompetenzen UND eine Aufnahme einiger Kompetenzen, die zu dem jeweiligen Fach passen. Der Informatikunterricht ist demnach der richtige Ort, um das von der KMK als informatische Grundkenntnisse beschriebene Grundverständnis zu vermitteln. Die weiteren Unterrichtsfächer können dann darauf zurückgreifen und die Nutzungskompetenzen vertiefen.

Die Umsetzung der vorgeschlagenen Veränderungsvorschläge kann allerdings nur gelingen, wenn auch die Lehrkräfte entsprechend diesem Anforderungsprofil ausgebildet worden sind. Dazu sind zwei Bedingungen zu erfüllen. Zum einen müssen die Lehrkräfte selbst über die Kompetenzen verfügen, die sie den Schülerinnen und Schülern vermitteln sollen. Zum zweiten müssen sie in der Vermittlung dieser Kompetenzen ausgebildet sein. Daraus ergeben sich neue Anforderungen sowohl an die Ausbildung als auch an die Fortbildung von Lehrkräften. Anzusetzen ist dazu bereits an den Hochschulen. Diese müssen ihrerseits durch entsprechende curriculare Vorgaben sicherstellen, dass die Absolventinnen und Absolventen der Lehramtsstudiengänge die entsprechenden Kompetenzen zur Bildung in der digitalen Welt, die sie in ihren Unterrichtsfächern vermitteln sollen, selbst erworben haben und wissen, wie sie diese weitergeben. Dies betrifft somit sowohl die Fachinhalte als auch die Fachdidaktiken.

Literaturverzeichnis

- [Bo14] Bos, Wilfried; Eickelmann, Birgit; Gerick, Julia; Goldhammer, Frank; Schaumburg, Heike; Schwippert, Knut; Senkbeil, Martin; Schulz-Zander, Renate; Wendt, Heike: ICILS 2013 Computer- und informationsbezogene Kompetenzen von Schülerinnen und Schülern in der 8. Jahrgangsstufe im internationalen Vergleich. Waxmann, Münster, Westf, 2014.
- [FPB13] Ferrari, Anusca; Punie, Yves; Brečko, Barbara N.: , DIGCOMP: A framework for developing and understanding digital competence in Europe, 2013.
- [Ge08] Gesellschaft für Informatik e.V.: , Grundsätze und Standards für die Informatik in der Schule, 2008.
- [Gl18] Glücks, Sebastian: , Analyse der niedersächsischen Kerncurricula der gymnasialen Sekundarstufe I auf Kompetenzen zur Bildung in der digitalen Welt. <https://oops.uni-oldenburg.de/id/eprint/3585>, 2018.
- [Ku16] Kultusministerkonferenz: , Bildung in der digitalen Welt – Strategie der Kultusministerkonferenz, 2016.
- [LK15] LKM - Länderkonferenz MedienBildung: , Kompetenzorientiertes Konzept für die schulische Medienbildung – LKM-Positionspapier Stand 29.01.2015, 2015.
- [Ni14] Niedersächsisches Kultusministerium: , Kerncurriculum für die Schulformen des Sekundarbereichs I Schuljahrgänge 5 - 10 Informatik, 2014.
- [Ni17] Niedersächsisches Kultusministerium: , Kerncurriculum für die Schulformen des Sekundarbereichs I Schuljahrgänge 5 - 10 Musik, 2017.

Beobachtungen zum informatischen Problemlösen im Escape-Adventure-Spiel „Room-X“

Alexander Hacke,¹ Mareen Przybylla,² Andreas Schwill³

Abstract: Informatisches Problemlösen ist einer der zentralen Aspekte der Informatik und ein Forschungsgebiet innerhalb der Informatikdidaktik, das Prozesse, Aufgaben und Einstellungen bezüglich des Problemlösens hinsichtlich informatischer Vorgehensweisen und Inhalte untersucht. Im informatischen Escape-Adventure-Spiel „Room-X“ bieten sich aus Forschungssicht Möglichkeiten, Lernende beim Problemlösen zu beobachten und deren Strategien zu analysieren. In diesem Artikel zeigen wir auf, wie der Room-X zu einer geeigneten Umgebung wird, die Lernenden ein Schaufenster für die Informatik bietet und ihnen ermöglicht, informatisches Problemlösen als eine der Kernkomponenten der Informatik in einem attraktiven und motivierenden Umfeld auszuüben. Mittels Videoanalyse werden Verhaltensweisen analysiert und Schlussfolgerungen für die gezielte Förderung von Problemlösekompetenzen in der Informatik sowie zur Weiterentwicklung des Room-X gezogen.

Keywords: Informatisches Escape-Game; Informatisches Problemlösen; außerschulischer Lernort

1 Einleitung

Problemlösen ist einer der zentralen Aspekte der Informatik und bildet mit jeder ihrer Teildisziplinen Verknüpfungspunkte. Um eine gute Grundlage für ein Informatikstudium beziehungsweise eine informatische Ausbildung zu erhalten, ist es eine Grundvoraussetzung für Schülerinnen und Schüler, sich mit dem Problemlösen in der Informatik explizit auseinanderzusetzen. Jedoch ist die Thematik vergleichsweise wenig erforscht. Folglich ist auch wenig darüber bekannt, wie informatisches Problemlösen in der Schule zielführend unterrichtet werden kann. Trotzdem ist Problemlösen fester Bestandteil der Bildungsstandards Informatik für die Sekundarstufe II [Ar16] und Bestandteil vieler deutscher Rahmenlehrpläne für das Fach Informatik (u. a. [Mi14; Mi18]). Da informatisches Problemlösen in der Schule oft in theoretischem Umfeld behandelt wird und mit wenig schüler nahen Inhalten besetzt ist oder aber nur implizit eine Rolle spielt, ist es für viele Schülerinnen und Schüler nur von geringem Interesse. Mit einem informatisch geprägten Escape-Room können Schülerinnen und Schüler für das Thema motiviert werden und gleichzeitig an einem außerschulischen Lernort die Anwendung von Problemlösestrategien vertiefen. Das informatisch geprägte Escape-Adventure-Spiel „Room-X“ fungiert getreu dem Motto „Informatik für alle“ als Schaufenster für die Informatik und zeigt, dass informatische Bildung auch im spielerischen

¹ Universität Potsdam, Didaktik der Informatik, August-Bebel-Str. 89, 14482 Potsdam, ahacke@uni-potsdam.de

² Universität Potsdam, Didaktik der Informatik, August-Bebel-Str. 89, 14482 Potsdam, przybyll@uni-potsdam.de

³ Universität Potsdam, Didaktik der Informatik, August-Bebel-Str. 89, 14482 Potsdam, schwill@uni-potsdam.de

Umfeld hilfreich sein kann. In diesem Beitrag wird zunächst theoretisch fundiert, was unter informatischem Problemlösen zu verstehen ist. Anschließend wird am Beispiel des „Room-X“ gezeigt, inwieweit Problemlösen sich an außerschulischen Lernorten integrieren lässt. Darauf basierend werden Empfehlungen für Weiterentwicklungen abgeleitet.

2 Informatisches Problemlösen

Problemlösen wird innerhalb der Kognitionspsychologie als der Versuch bezeichnet, *von einem Anfangszustand vorbei an einer Barriere zu einem Zielzustand zu gelangen* [MR17]. Problemlösen verlangt eine Reihe kognitiver Fähigkeiten, welche nach Bloom aufeinander aufbauend in sechs Kategorien eingeteilt werden können, wobei die oberen drei („Analysing“, „Evaluating“, „Creating“) als „higher-order thinking“ betrachtet werden und die unteren drei („Remembering“, „Understanding“, „Applying“) voraussetzen [AK01]. Die Bearbeitung von einfachen Aufgaben lässt sich normalerweise durch die unteren drei Kategorien abbilden. Es ist hierfür erforderlich, die Aufgabenstellung zu verstehen (Understanding), dazu passende Informationen und Prozeduren aus dem Langzeitgedächtnis abzurufen (Remembering) und diese im gegebenen Kontext anzuwenden (Applying). Das Problemlösen benötigt zusätzlich auch die Fähigkeiten des „higher-order thinking“. Das Problem muss analysiert werden, wobei wichtige von unwichtigen Details unterschieden und nicht vordergründig sichtbare Details erkannt werden müssen (Analyzing). Aus der Analyse muss per Heuristik eine zielführende Strategie generiert werden, die gegebenenfalls bekannte elementare Prozeduren mit neuen Zusammenhängen verknüpft (Creating). Dabei ist diese Strategie ständig auf Effektivität zu überwachen und gegebenenfalls zu überdenken (Evaluate). Zum erfolgreichen Problemlösen ist es zusätzlich erforderlich, Zutrauen in die eigenen Fähigkeiten zu haben. Auch die Einstellungen zum konkreten Problem und Problemlösen an sich sind maßgeblich mit dafür verantwortlich, wie gut ein Problemlöser die ihm zur Verfügung stehenden Mittel nutzen will und kann (vgl. [Sc13]). Der Begriff „Problem“ muss in Hinblick auf das Problemlösen eingegrenzt werden. Eine aus der Psychologie stammende Definition besagt, dass ein *Problem dann vorliegt, wenn in einer Situation, in der ein bestimmtes Ziel erreicht werden soll, ein Hindernis oder eine Barriere dies verhindert* [MR17]. Probleme lassen sich auf vielfältige Weise kategorisieren. Man kann sie beispielsweise anhand dessen unterscheiden, wie klar der zu erreichende Zielzustand definiert ist, oder danach, ob es sich um sogenannte *einfache* oder *komplexe* Probleme handelt. Bei *komplexen* Problemen ändern sich im Verlauf des Lösungsversuchs die Rahmenbedingungen, was eine fortwährende Neubewertung des Lösungsansatzes erfordert. Zudem spielen sehr viele Variablen eine Rolle, die sich teilweise gegenseitig bedingen. Zu komplexen Problemen zählen beispielsweise die Steuerung eines Unternehmens oder die Bewältigung einer globalen Krise. *Einfache* Probleme hingegen besitzen stabile Rahmenbedingungen und im Vergleich weniger relevante Variablen, was jedoch nicht bedeutet, dass sie leicht zu lösen wären. Informatische Problemstellungen fallen zumeist in die Kategorie der *einfachen* Probleme. Das bedeutet, die Rahmenbedingungen verändern sich während des Lösungsversuchs nicht oder nur wenig und die Anzahl der zu beachtenden Variablen bewegt

sich im überschaubaren Bereich. Natürlich können sie, eingebettet in eine realweltliche Situation, auch Teil eines *komplexen* Problems sein. Im hier gedachten Zusammenhang soll jedoch der Fokus auf *einfache* Probleme gesetzt werden, da sonst nicht mehr eindeutig ist, ob das Problem informatischer oder sonstiger Natur ist.

Definition: Ein informatisches Problem liegt vor, wenn in einer Situation mit stabilen Rahmenbedingungen, in der ein bestimmtes Ziel erreicht werden soll, ein Hindernis oder eine Barriere dies verhindert, wobei zur Zielerreichung ein informatisches Verfahren benötigt wird.

Nun stellt sich die Frage, ob das vorliegende Problem informatischer Natur ist oder nur der gewählte Problemlöseablauf – oder ob möglicherweise beide Teile der Informatik zuzurechnen sind. Ähnlich der Vorgehensweise durch Humbert und Puhmann bei der Unterteilung informatischer Phänomene in drei Kategorien [HP04] lassen sich auch Probleme nach deren Bezug zur Informatik klassifizieren:

1. *Das Problem ist nicht informatischer Natur.* Probleme rein philosophischen Charakters, bei denen ein informatischer Lösungsansatz nicht sinnvoll bzw. deplatziert ist.
2. *Das Problem ist indirekt informatischer Natur.* Probleme, die realweltlichen Charakter haben, jedoch inhärent informatisch sind und somit durch eine informatische Problemlösestrategie gelöst werden können.
3. *Das Problem ist direkt informatischer Natur.* Probleme, die eine Problemlösestrategie mit informatischen Prinzipien erfordern.

Unterteilt man Probleme auf diese Weise, so wird offensichtlich, dass sich ein Problem der Kategorie drei⁴ durchaus in Kategorie zwei wiederfinden kann, nämlich als informatischer Teil eines realweltlichen Problems. Beispielsweise ist ein Problem, bei dem es um die Verteilung von Aufgaben an Mitarbeiter oder die Umlaufplanung im Nahverkehr geht, oft ein ganzzahliges Optimierungsproblem, welches mit dem Branch-and-Bound-Verfahren aus der Informatik gelöst werden kann. Der Problemraum eines solchen Problems besteht dann nicht nur aus dem informatischen Problem, sondern auch darin, dass der informatische Charakter zunächst erkannt werden muss. Probleme der Kategorie zwei lassen sich aber eventuell auch auf nicht-informatischem Wege lösen. Beispielsweise lässt sich ein Optimierungsproblem auch per zufälliger Verteilung lösen, da die Frage nach der besten Verteilung bei einem realweltlichen Problem nicht immer im Vordergrund steht. Offensichtlich wird auch, dass die Ausgangssituation des Problems über die Wahrscheinlichkeit des Informatikgehalts des Problemlösevorgangs Auskunft gibt.

⁴ Hierzu zählen Probleme wie das Finden des dichtesten Punktepaars, das mit dem Divide-and-Conquer-Verfahren gelöst werden kann oder die Programmierung eines Oberon-Programms zur Berechnung von Fakultäten, wobei hier Programmierkonzepte zur Problemlösung erforderlich sind.

3 Problemlösen in Escape-Adventure-Spielen

Escape-Adventure-Spiele haben ihren Ursprung in den sogenannten Point-and-Click-Adventures, welche sich als Konsolen- oder Computerspiele bereits seit den 1970er Jahren großer Beliebtheit erfreuten und in den frühen 2000er Jahren als Browser-Games wiederbelebt wurden. Escape-Rooms (auch als Live-Escape-Games, Exit-Rooms und unter weiteren, ähnlichen Begriffen bekannt) sind eine spezielle Art der Escape-Adventure-Spiele, in denen die Spieler als Team in einen real existierenden Raum eingeschlossen werden und mit Hilfe der im Raum befindlichen Hinweise und Rätsel in begrenzter Zeit versuchen, aus dem Raum zu entkommen. Meist gilt es zusätzlich noch eine Mission zu erfüllen, wie beispielsweise das Entschärfen einer Bombe, das Lösen eines Kriminalfalls oder das Stehlen eines Gegenstandes. Die Themen für solche Spiele sind dabei äußerst vielfältig und siedeln sich gern in spannenden Settings an, z. B. Chemielaboren, Gefängnistrakten oder Agentenbüros. Escape-Adventure-Spiele bieten auch für Bildungskontexte Anreize. So beschreibt beispielsweise Nicholson [Ni18] als Vorzüge der Verwendung solcher Spiele im Unterricht u. a. die Abwechslung zur Arbeit am Computer, die erforderliche Kooperation im Team und motivationale Aspekte als Grundlage für aktives Lernen und sozialen Konstruktivismus. Escape-Adventure-Spiele bieten eine gute Gelegenheit, Problemlösefähigkeiten zu trainieren. Das Konzept solcher Spiele beinhaltet die wesentlichen Merkmale eines o. g. *einfachen* Problems und macht die Spielenden somit zu Problemlösern. Sie müssen innerhalb einer bestimmten Zeit von einem Ausgangszustand (der Raum und die mitgegebenen Hinweise) zu einem Zielzustand (meist: den Raum verlassen) gelangen. Dies ist nicht ohne Weiteres möglich, da ihnen ein oder mehrere Hindernisse (Rätsel; der Raum lässt sich nicht ohne Weiteres öffnen, etc.) in den Weg gestellt werden. Sie müssen sich daher heuristischer Verfahren (z.B. Bildung von Teilzielen, Suchraumeingrenzung, Visualisierung) bedienen, kreativ eine Lösungsstrategie planen und diese bei der Durchführung stetig auf Sinnhaftigkeit überprüfen. Zudem sind Escape-Adventure-Spiele durch den spielerischen Abenteuer-Charakter gut geeignet, um die Motivation der Teilnehmenden hoch zu halten und eventuell vorhandene negative Einstellungen zum Problemlösen zu überdecken. Eventuell wird das enthaltene Problemlösen sogar gar nicht als solches wahrgenommen.

4 Room-X: Ein Escape-Adventure-Spiel für den Informatikunterricht

Um Schülerinnen und Schülern in begrenzter Zeit und auf motivierende Weise einen Einblick in verschiedene Themengebiete der Informatik zu geben, sie für das Fach zu begeistern und für das Institut für Informatik zu werben, wurde an der Universität Potsdam das Escape-Adventure-Spiel „Room-X“ eingerichtet. In diesem Kontext besuchen uns regelmäßig Schülergruppen, die sich bereitwillig beim Problemlösen beobachten lassen. Anders als bei anderen Escape-Adventure-Spielen, deren Themen meist die Allgemeinheit ansprechen, ist das Szenario des Room-X auf die Zielgruppe ab der zehnten Klasse zugeschnitten. Ihre Mission beinhaltet, die Aufgaben der nächsten Informatikarbeit im Klassenraum von Herrn Schröder auszuspionieren, welche sich auf einem passwortgeschützten Tablet befindet. Die

Klausur muss abfotografiert werden, andernfalls ist die Mission nicht vollständig erfüllt und gilt als verloren. Das Passwort lässt sich mithilfe der Gegenstände im Raum in Erfahrung bringen. Zusätzlich hat der Lehrer die Alarmanlage der Tür des Klassenraums aktiviert. Um unbemerkt wieder hinaus zu gelangen, müssen die Teams den Zahlencode des Schlosses herausfinden, welches den Zugang zu einer Fernbedienung der Alarmanlage versperrt. Die Teams im Room-X werden während der gesamten Spielzeit durch eine Kamera im Raum beobachtet, damit ihnen gegebenenfalls durch Tipps geholfen werden kann. Das Spiel dauert 60 Minuten, nach Ablauf der Zeit löst die Alarmanlage aus. Auch das vorzeitige Öffnen der Tür löst die Alarmanlage aus und führt zu Disqualifikation und zum Abbruch der Mission. Die Nutzung der Tafel und der Blöcke und Stifte im Raum ist ausdrücklich erlaubt.

Zunächst erhalten die Teams in einem separaten Raum sämtliche Informationen über das Szenario, die Abläufe und die Spielregeln des Room-X. Es wird darauf hingewiesen, dass die Spielleitung die Personen im Raum im Laufe des Spiels kontaktieren kann. Die Art der Kontaktaufnahme bleibt zunächst offen. Anschließend wird das Team in den Room-X geführt, der Timer wird gestartet, die Tür geschlossen und die Alarmanlage aktiviert. Um das Spiel zu gewinnen, muss das Team nun verschiedene Rätsel lösen. Um eine Verbreitung des Lösungsweges für den Raum zu verhindern, wird an dieser Stelle darauf verzichtet, die Rätsel im Detail zu beschreiben. Man kann sie sich aber in etwa wie folgt vorstellen:

- Das Tablet ist mit einem Passwort gesichert, daran klebt ein Zettel mit der Aufschrift „PW: Endlich Ferien! (HexHex)“ → Was soll das heißen?
- An der Pinnwand befindet sich ein Zettel mit einer kryptischen Botschaft: „Fcu Htjjuvüem nkgiv okt jgwvg uejygt ko Ocigp.“ → Was hat das zu bedeuten?
- Auf dem Lehrertisch liegt eine SD-Karte mit der Aufschrift *GEHEIM*, legt man sie in eine herumliegende Digitalkamera ein, sieht man Fotos von verschiedenen Gegenständen (z. B. ein riesiges Gerät mit der Aufschrift *Z3* und eine merkwürdig bunt eingefärbte Landkarte) → Befinden sich in den Bildern entscheidende Hinweise?

Durch das Erkennen und Lösen typisch informatischer Problemstellungen und die geschickte Kombination gefundener Hinweise ist es schließlich möglich, das Tablet zu entsperren, den Schlüsseltresor zu öffnen und die Alarmanlage zu deaktivieren. Im Anschluss haben die Teammitglieder die Möglichkeit, ihre Erlebnisse untereinander zu diskutieren und Hintergründe und Lösungen zu einzelnen Rätseln des Raumes zu erfahren.

5 Room-X und informatisches Problemlösen

Mit dem Ziel, das Vorgehen der Schülerinnen und Schüler beim Lösen informatischer Probleme im Room-X zu beobachten, deren Strategien zu identifizieren und daraus Schlussfolgerungen für die gezielte Förderung von Problemlösekompetenzen im Room-X (und

allgemein) abzuleiten, soll zunächst untersucht werden, in welchem Maße das Spiel informatisches Problemlösen (er)fordert. Hierzu wird der Raum mit den dazugehörigen Aufgaben und Rätseln im Folgenden unter Bezugnahme auf die o. g. Definitionen und „Thinking skills“ analysiert. In einer anschließenden Videoanalyse werden die Strategien der Teilnehmer identifiziert und bzgl. der zum Erfolg führenden Eigenschaften analysiert.

5.1 Problem und Problemlösen im Room-X

Die Startsituation, mit der die Teilnehmenden konfrontiert werden, entspricht einem *einfachen* Problem der o. g. Problemdefinition, denn es lassen sich die folgenden Merkmale wiederfinden: Das Team befindet sich am Anfang der Spielphase im Ausgangszustand, der aus dem Raum mit seinen versteckten Hinweisen und den vom Übungsleiter genannten Hinweisen besteht. Die Gruppe kann nicht ohne Weiteres zum Zielzustand übergehen, also die Klassenarbeit abfotografieren und die Tür öffnen, da der Weg dorthin durch verschiedene Hindernisse (Türcode, Tablet-Passwort) versperrt ist. Die Rahmenbedingungen ändern sich während der Suche im Problemraum nicht, wenn man vom sich verstärkenden Zeitdruck absieht. Auch die Anzahl der Variablen, mit denen im Verlauf hantiert werden muss, ist überschaubar groß und die Wechselwirkungen zwischen ihnen sind gering. Dementsprechend lässt sich eine Kategorisierung als *komplexes* Problem ausschließen. Untersucht man den durch Rätsel vorgegebenen Weg durch den Room-X, so zeigt sich, dass viele Elemente aus dem Bereich der „lower-order Thinking Skills“ enthalten sind. Das heißt, es gibt eine Reihe von Aufgaben, die im Raum zu lösen sind. Es fällt auf, dass es selbst mit sorgfältiger Planung teils nicht möglich ist, den Zusammenhang zwischen den Aufgaben vorab herzustellen, beziehungsweise zu erkennen, wozu die Lösung der Aufgabe dienlich sein könnte. Das hat weniger mit der Komplexität der zu findenden Strategie zu tun als damit, dass der vorgesehene Lösungsweg an einigen Stellen zu stark konstruiert ist und Zusammenhänge keinem erkennbaren Muster folgen und somit kaum nachvollziehbar sind. Der Weg hin zum Zielzustand erfordert oft einfach Brute Force, Teamarbeit und Glück.

5.2 Informatik im Room-X

Das gestellte Problem an sich ist zunächst kein informatisches. Es wird nach einer Klausur gesucht und nach einem Weg, um ein Schloss zu knacken. Hinter dem Problem verbirgt sich keine direkte informatische Problemlösestrategie, somit fällt es in die Kategorie der nicht-informatischen Probleme. Der Weg durch den Problemraum enthält jedoch eine Reihe von Aufgaben informatischer Natur, beispielsweise in den Themengebieten Verschlüsselung, Logik und Automatentheorie. Allerdings müssen einige der Aufgaben nicht wegen ihres informatischen Hintergrunds gelöst werden, sondern weil sie beim Lösen einen nicht-informatischen Hinweis preisgeben. Zudem sind die Aufgaben derart konstruiert, dass sie von jedermann gelöst werden können, also keine informatischen Kompetenzen voraussetzen. Auch zusammengenommen ergeben sie keine speziell informatische Problemlösestrategie

nach o. g. Definition, sondern stehen für sich. Die Problemlösestrategie besteht dadurch vor allem darin, nach Hinweisen zu suchen, die Aufgaben richtig zu lösen und die gefundenen Hinweise und Lösungen derart zusammenzubringen, dass sich daraus das Endergebnis ergibt. Zusammenfassend lässt sich feststellen, dass der Room-X zwar informatische Anteile auf Aufgabenlevel enthält, jedoch gesamt gesehen kein informatisches Problemlösen. Andererseits sind die geforderten Teilstrategien beim informatischen Problemlösen unabdingbar. Das korrekte Lösen von Aufgaben und die dazugehörige Evaluation der Ergebnisse sind Grundbausteine auf dem Weg zum erfolgreichen Problemlösen. Lässt man die systematische Suche im Problemraum aus, so werden leicht entscheidende Hinweise übersehen. Ohne die Gewissheit, dass es sich um die korrekte Lösung handelt, kann nicht mit derselben weitergearbeitet werden, ohne Folgefehler zu riskieren. Letztlich ist auch die Kombination von Teilergebnissen ein Grundbaustein beim informatischen Problemlösen. Dieser Schritt setzt eine fortlaufende Dokumentation der Ergebnisse sowie eine zielführende Darstellung voraus, die Zusammenhänge erkennbar machen kann. Somit lassen die im Room-X benötigten Strategien beim Problemlösen Rückschlüsse darauf zu, wie gut die Schülerinnen und Schüler für das informatische Problemlösen gewappnet sind.

5.3 Angewandte Problemlösestrategien im Room-X

In einer qualitativen Videoanalyse soll nun zunächst geprüft werden, ob die oben genannten Teilstrategien beobachtet werden können und ob sich ein Einfluss auf den Erfolg beim Problemlösen ableiten lässt. Für die Videoanalyse ergeben sich daraus folgende Fragestellungen:

1. Welche typischen Verhaltensweisen können beim Problemlösen im Room-X beobachtet werden?
2. Welchen Einfluss haben die beobachteten Verhaltensweisen auf den Erfolg beim Problemlösen im Room-X?

Bezüglich der Fragestellungen lassen sich basierend auf den in Abschnitt 2 dargestellten Überlegungen und Definitionen sowie den Eigenschaften des Room-X folgende Annahmen ableiten: Erwartungsgemäß werden die Teams die Gegenstände im Raum nach Hinweisen aller Art durchsuchen, die Teammitglieder ihren Vorlieben bzw. Vorkenntnissen entsprechend auf die Aufgaben verteilen, versuchen die einzelnen Aufgaben zu lösen, die Tafel bzw. Blöcke als Mittel zur Visualisierung bzw. Repräsentation der Erkenntnisse nutzen, miteinander kommunizieren und Erkenntnisse im Team evaluieren. Dabei wird davon ausgegangen, dass die folgenden Verhaltensweisen zum Erfolg führen: systematisches Suchen nach Hinweisen, korrekte Lösung der Einzelaufgaben, Visualisierung und Repräsentation der Hinweise und Ergebnisse, Einbeziehung aller Teammitglieder und Evaluation und Kombination der Hinweise und Ergebnisse miteinander.

5.4 Durchführung

Für die Videoanalyse liegt Videomaterial von 24 Gruppen à 5-6 Personen vor, was ca. 130 Teilnehmenden entspricht. Das Material ist hochauflösendes Video mit Ton von einer Überwachungskamera an der Decke des Raumes. Diese Überwachung dient dem Spielleiter normalerweise zur Steuerung des Spiels. Das Videomaterial wurde hinsichtlich des Erfolg bringenden Verhaltens der Teilnehmenden beim Problemlösen untersucht. Dabei wurden folgende Verhaltensweisen vorab isoliert und operationalisiert (deduktives Vorgehen):

1. korrekte Lösung der Einzelaufgaben: Ein oder mehrere Teammitglieder lösen eine der Aufgaben und erhalten eine korrekte Lösung.
2. Einbeziehung aller Teammitglieder: Alle Teilnehmer sind fokussiert auf das Problem, d. h. suchen nach Hinweisen, geben Ratschläge, helfen anderen, lösen Aufgaben.
3. systematisches Suchen nach Hinweisen: Der Raum wird von einem Ende zum anderen gründlich auf Hinweise untersucht, idealerweise unabhängig durch mehrere Personen.
4. Visualisierung und Repräsentation der Hinweise und Ergebnisse: Die Tafel bzw. ein Schreibblock wird genutzt, um Zwischenergebnisse, Hinweise, Erkenntnisse, Fragen zu protokollieren, sobald diese zur Verfügung stehen.
5. Evaluation und Kombination der Hinweise und Ergebnisse miteinander: Ergebnisse werden gegenseitig kontrolliert, an Tafel oder mündlich in Bezug gesetzt.

5.4.1 Beobachtete Verhaltensweisen

Nach qualitativer Auswertung von ca. 70% der Videos lassen sich Tendenzen bezüglich der ersten Fragestellung erkennen. Typischerweise beginnt die Gruppe mit Brute-Force-Heuristik: alle Teammitglieder schwärmen aus, verteilen sich im Raum, blättern Bücher durch, etc. Dies entspricht dem zu erwartenden Durchsuchen der Gegenstände im Raum nach Hinweisen. Im Zuge dieses Prozesses werden verschiedene Aufgaben entdeckt und im Normalfall sofort versucht, sie zu lösen. Dabei werden oft Präferenzen berücksichtigt. Aufgaben, die einer Person zu schwer erscheinen, werden liegengelassen oder jemand anderes zu Rate gezogen, beispielsweise passiert das, wenn eine Umrechnung in ein anderes Zahlensystem erfolgen muss. Diese Verhaltensweisen wurden so auch erwartet. Ebenso erfolgt die Nutzung der Tafel, um Einzelergebnisse zu dokumentieren. Jedoch ist die Dokumentation der gefundenen Hinweise oft gering ausgeprägt und eine Visualisierung selten. Hin und wieder gehen Hinweise im Kommunikationsprozess der Teammitglieder zunächst wieder verloren und müssen dann erneut entdeckt werden. Die Annahmen bezüglich der angewandten Problemlösestrategien bestätigen sich also weitgehend. Alle erwarteten Verhaltensweisen lassen sich beobachten. Darüber hinaus werden auch Verhaltensweisen sichtbar, die auf Demotivation einzelner Teammitglieder schließen lassen, d. h. mitunter gibt es Teilnehmer, die oft aus dem Fenster blicken oder teilnahmslos herumstehen.

5.4.2 Erfolgversprechendes Verhalten

Auch die Annahme hinsichtlich erfolgversprechender Verhaltensweisen lässt sich durch das Videomaterial weitgehend bestätigen, wobei folgende Aspekte deutlich wurden: Teams, die die Tafel intensiver für Notizen nutzen, sind meist erfolgreicher. Es zeigte sich beispielsweise, dass oft Striche für die Anzahl der Stellen des Passworts an die Tafel gebracht werden, was einer allgemein verständlichen Repräsentation des Teilziels entspricht. Andererseits entstanden oft auch Kritzeleien, deren Sinn anderen Teilnehmern verschlossen bleibt. Es zeigte sich ferner, dass auch Teams ohne erkennbar strukturierten Tafelanschrieb frühzeitig fertig sein können, sofern sie trotzdem viel aufschrieben. Wenig bis gar nicht zu finden sind informatische Darstellungen, was jedoch auch so zu erwarten ist. Die Zuhilfenahme von Notizen spielt auch bei der Lösung der Einzelaufgaben eine Rolle. Erfolgreiche Teams nutzen hier oft Zettel und Stift. Die Evaluation der Ergebnisse ergibt sich bei den Aufgaben oft einfach dadurch, dass die Lösung ein „sinnvolles“ Wort ist und die Teilnehmer daraus eine Bestätigung der Richtigkeit ableiten. Außerdem zeigt sich, dass es stark unterschiedlich funktionierende Teams gibt. Es gibt Teams, deren Mitglieder viel untereinander kommunizieren, aber auch solche, in denen Mitglieder relativ teilnahmslos erscheinen, teils auch destruktiv arbeiten. Neben diesen nicht im Lösungsprozess involvierten Teilnehmern kristallisierten sich vier Teilnehmertypen heraus: Solche, die Aufgaben verteilen wollen, solche, die Aufgaben annehmen wollen, solche, die autark arbeiten wollen, und solche, die mit anderen zusammenarbeiten wollen. Eine Vorhersage auf Erfolg und Misserfolg bei der Absolvierung des Room-X ließ sich daraus jedoch zunächst nicht ableiten.

6 Fazit und Ausblick

Der Room-X ist ein Escape-Adventure-Spiel mit informatischen Anteilen, der allgemeine Problemlösestrategien erfordert. Vorteilhaft am gegenwärtigen Konzept ist, dass kein Vorwissen bezüglich einer speziellen Vorgehensweise erwartet wird. Es kann also eine breite Zielgruppe angesprochen werden, da nur Fähigkeiten erwartet werden, die die meisten Lernenden ab Klasse 10 mitbringen. Daraus ergibt sich ferner, dass das Problem für höhere Jahrgänge oder Studierende nicht einfacher wird und das Konzept somit auch für sie ohne Veränderungen durchführbar bleibt. Nachteilig ist jedoch, dass gerade erfahrene Teams keine eventuell bekannten informatischen Strategien anwenden können, um das Problem (schneller) zu lösen. Dies kann mitunter frustrieren und entspricht weniger den Erwartungen an einen informatischen Escape-Room. In Bezug auf die einzelnen Aufgaben ist vorteilhaft, dass auch Teams mit keinen oder nur geringen Vorkenntnissen ein motivierender Einblick in informatische Ideen gegeben werden kann, um ihnen so die Informatik schmackhaft zu machen, auch wenn einige Aufgaben derzeit recht konstruiert wirken und nicht notwendig im Sinne eines informatischen Lösungsschritts sind. Um künftig den Informatikgehalt je nach Besuchergruppe anpassen zu können, sind modulare Weiterentwicklungen geplant, die die Nützlichkeit der informatischen Ideen und den Sinn informatischer Verfahren

betonen. Auf diese Weise können einzelne Rätsel ausgetauscht werden, um inhaltliche Schwerpunkte zu verschieben und den Raum an die Vorkenntnisse der Teilnehmer zu adaptieren. Aus den Ergebnissen der Videoanalyse lassen sich erste Hinweise für die Förderung von Problemlösekompetenzen ableiten. In Bezug auf die Dokumentation ist die geringe Tendenz zu strukturierter Darstellung besonders auffällig. Die Ursachen hierfür müssen daher näher untersucht werden. Idealerweise sollten sich die Problemlösenden per geschickt gewählter Strategie zielgerichtet durch den Problemraum bewegen. Dies setzt jedoch voraus, dass der Lösungsweg so gestaltet ist, dass er durch gute Planung, Evaluation und Anwendung einer Strategie zu Fortschritten führt. Die Videoanalyse zeigte jedoch, dass keinerlei Planungsphase stattfindet, sondern stattdessen gleich mit der Suche begonnen wird, da das derzeitige Konzept eine solche Phase nicht erfordert. Um künftig den Problemlöseaspekt zu verstärken, muss der gegenwärtige Ablauf ersetzt werden. Eine mögliche Herangehensweise besteht darin, informatische Konzepte auf ihre Struktur hin zu untersuchen und diese als (Teil-)Strategie zu integrieren.

Literatur

- [AK01] Anderson, L. W.; Krathwohl, D. R.: A taxonomy for learning, teaching, and assessing: a revision of Blooms taxonomy of educational objectives. Longman, 2001.
- [Ar16] Arbeitskreis Bildungsstandards SII: Bildungsstandards Informatik für die Sekundarstufe II, Beilage zur LOG IN 183/184, 2016.
- [HP04] Humbert, L.; Puhlmann, H.: Essential ingredients of literacy in informatics. In (Magenheim, J.; Schubert, S., Hrsg.): Informatics and Student Assessment - Concepts of Empirical Research and Standardisation of Measurement in the Area of Didactics of Informatics. Bd. 1. LNI Seminars, Gesellschaft für Informatik, Bonn, S. 65–76, 2004.
- [Mi14] Ministerium für Schule und Weiterbildung des Landes Nordrhein-Westfalen: Kernlehrplan für die Sekundarstufe II Gymnasium/Gesamtschule in Nordrhein-Westfalen, 2014.
- [Mi18] Ministerium für Bildung, Jugend und Sport des Landes Brandenburg: Rahmenlehrplan für den Unterricht in der gymnasialen Oberstufe im Land Brandenburg, 2018.
- [MR17] Müsseler, J.; Rieger, M.: Allgemeine Psychologie. Springer-Verlag, Berlin Heidelberg New York, 2017, ISBN: 978-3-642-53898-8.
- [Ni18] Nicholson, S.: Creating Engaging Escape Rooms for the Classroom. *Childhood Education* 94/1, S. 44–49, 2018.
- [Sc13] Schoenfeld, A. H.: Reflections on Problem Solving Theory and Practice. *The Mathematics Enthusiast* 10/1, S. 9–34, 2013.

Informatik in der Grundschule – Stellschraube Lehrerbildung

Zur Notwendigkeit nachhaltiger informatischer Bildung für angehende Grundschullehrkräfte

Kathrin Haselmeier¹

Abstract: Die universitäre Lehrerbildung für Grundschullehrkräfte muss Informatik als Gegenstand der allgemeinen Bildung umfassen. Aus entwicklungspsychologischer Sicht begründet der Beitrag, warum Informatik bereits in der Grundschule Bestandteil des Unterrichts sein muss und referenziert dabei aktuelle Studienergebnisse. Im Spannungsfeld Lehren und Lernen wird herausgestellt, dass gut ausgebildete Lehrkräfte zwar zentral für die kompetente Vermittlung informatischer Inhalte und Gegenstände in der Grundschule sind, sie jedoch bislang im Rahmen ihrer Ausbildung bzw. der Weiterqualifizierung wenig Möglichkeiten haben, fundierte informatische Kompetenzen auf- und auszubauen. Fortbildungsmöglichkeiten für Bestandslehrkräfte werden aufgezeigt, hinsichtlich ihrer Möglichkeiten und Grenzen diskutiert und schließlich der Einfluss auf die universitäre Lehrerbildung als maßgebliche Stellschraube für Informatik in der Grundschule identifiziert. Aus der dargestellten Notwendigkeit heraus wurde eine eigene universitäre Veranstaltung für Studierende konzipiert und (bisher zwei Mal) durchgeführt.² Vorläufige Evaluationsaspekte aus der Veranstaltung werden vorgestellt und diskutiert.

Keywords: Grundschule; Informatische Bildung; Lehrerbildung

1 Situation

Informatik wird gebraucht. Diese Aussage ist die Essenz eines wirtschaftlichen, politischen und gesellschaftlichen Diskurses, der inzwischen seit einigen Jahren sehr präsent geführt wird. Allen Hindernissen und allen Bedenken zum Trotz bleibt unterm Strich die Erkenntnis: Ohne Informatikkenntnis ist eine mündige Teilhabe kaum mehr zu realisieren. Informatik findet sich in allen Lebensbereichen, sie ist omnipräsent und wird nicht mehr verschwinden – egal, wie man zu ihr steht. Ebenso ist unbestritten, dass es viel zu wenig Menschen mit informatischer Bildung gibt, die als Experten Gestaltungsaufgaben übernehmen und informatische Umsetzungen sinnstiftend modellieren können. Sobald jedoch der Begriff »Bildung« im Zusammenhang mit »Informatik« genannt wird, rückt auch die Schule als Bildungsinstitution in den Fokus. In den Dreiklang aus Informatik-Bildung-Schule mischt

¹ Bergische Universität Wuppertal, Didaktik der Informatik, Gaußstraße 20, 42119 Wuppertal, Deutschland
khaselmeier@uni-wuppertal.de

² Gefördert im Rahmen der Qualitätsoffensive Lehrerbildung des BMBW – Teilprojekt KoLBi, »Kohärenz in der Lehrerbildung« an der Bergischen Universität.

sich noch eine weitere Komponente: Informatik gilt als Männerdomäne. Genderfragen können also nicht unberücksichtigt bleiben.

1.1 Genderspekte als Indikator für den »richtigen Zeitpunkt« zur Wahrnehmung einer informatischen Bildungsaufgabe

Aus der Geschlechterforschung ist bekannt, dass Kinder bis zum Eintritt in die Grundschule bereits sehr rigide, traditionelle Geschlechtsrollenstereotype ausbilden. Diese Zuweisung weicht ab Mitte der Grundschulzeit wieder auf, wird bis zum Einsetzen der Pubertät modifiziert und dann sehr beständig getroffen [vgl. BK10, S. 181]. Es ergibt sich also ein recht kurzes Zeitfenster, in welchem informatische Bildung dazu beitragen kann, die Zuweisung an die Disziplin als »Männerdomäne« zu überwinden. Wagner gibt darüber hinaus zu bedenken, dass geschlechterstereotype Verhaltensweisen eine deutliche Auswirkung auf Berufswahl und Bildungschancen haben und insbesondere Mädchen daraus resultierend Benachteiligungen erwachsen [vgl. Wa17]. Vor dem Hintergrund, dass Mädchen im Mittel über signifikant höhere computer- und informationsbezogene Kompetenzen verfügen als Jungen, ist diese Pille besonders bitter [vgl. Fr14].

Die Grundschule muss als erste verpflichtende Bildungseinrichtung explizit in den Blick genommen werden und mit ihr auch die überwiegend weiblichen Lehrkräfte.

1.2 Informatische Bildung bei den »Bestandslehrkräften«

Nun erwächst Bildung nicht schlicht aus dem Besuch einer Institution – vielmehr sind es die Menschen, welche dort unterrichten, die den Bildungserfolg und die Rollenzuweisungen ihrer Schüler*innen maßgeblich beeinflussen. Fehlen also im Erwachsenenalter informatikkundige Menschen, so ist ein Blick auf das eingesetzte pädagogische Personal, in die institutionellen Bildungsaufgaben und die -abläufe aufschlussreich: An weiterführenden Schulen wird zu wenig Informatik unterrichtet und erreicht zudem dort überwiegend Jungen. In der Grundschule fehlt Informatik (fast) vollständig, ist nicht im Fächerkanon enthalten und kommt somit im Studium oder in der zweiten Phase der Lehrerbildung nicht vor. Informatische Bildung bleibt individuellen Interessen und Schwerpunktsetzungen vorbehalten.

2 Informatik versus Medienbildung

Da in den Kollegien in der Regel keine informatische Bildung vorhanden ist, können sich die Schulen nur auf ihre Vorstellungswelt zur »Digitalisierung« stützen. So werden überwiegend Angebote genutzt, die den Medieneinsatz im Unterricht zum Thema haben. Dieser Fokus ist für Grundschulen zudem attraktiv, denn der Einsatz digitaler Medien hat das Potential, einige der Schwierigkeiten in den Grundschulen zu adressieren:

Kompensation von zu wenig voll ausgebildeten Lehrkräften in den Kollegien, Unterstützung bei unterschiedlichen Aufgaben in Inklusion und Integration und Möglichkeiten

für individuelle Förderungen auf verschiedenen Anforderungsniveaus.³ Für nachhaltige, aufschließende informatische Bildung jedoch müssten Lehrkräfte nach Bergner et al. [vgl. Bel7a] neben pädagogischer und didaktischer Handlungskompetenz auch über fachliche und fachdidaktische Kompetenzen verfügen, eine positive Einstellung zur Informatik aufweisen sowie offen für neue Sichtweisen sein. Die grundlegende Bereitschaft sich entsprechend fortzubilden darf bei Lehrkräften vorausgesetzt werden, sie ist im Dienstrecht verankert.⁴

2.1 Fortbildungen als Problemlösung?

Fortbildungen werden von Grundschullehrkräften in Nordrhein-Westfalen in der Regel in zwei verschiedenen Modi besucht: Entweder im Rahmen ihrer Tätigkeit für die Fachkonferenzen, um auf dem aktuellen Stand in den Fächern zu bleiben oder vorhandene Fachkenntnisse vertiefen zu können, oder im Kontext einer Fortbildung für das gesamte Kollegium einer Grundschule. Diese Fortbildungen richten sich an die Arbeitsschwerpunkte, die alle Kollegiumsmitglieder vereinen: Pädagogik in allen Ausprägungen und Schwerpunkten, Schulentwicklung, Unterrichtsmethoden, sowie schul- und klassenorientierte Organisationsformen. Neben dem Aspekt der Freiwilligkeit und der interessensgebundenen Auswahl, der oft fehlenden Möglichkeit, überhaupt an Fortbildungen teilzunehmen und dem breiten Aufgabenspektrum an Grundschulen, werden Fortbildungen für Grundschullehrkräfte in Nordrhein-Westfalen maßgeblich schul- und themenbezogen von Kolleg*innen anderer Schulen der gleichen Schulform durchgeführt. Für die Ausbildung und die Fortbildung der Lehrkräfte an Grundschulen bedeutet dies, dass das Fortbildungsportfolio zur »Digitalisierung« nahezu vollständig von *medienaffinen* Grundschullehrkräften angeboten wird. Es entsteht eine Art Perpetuum mobile der sich selbst bestärkenden Vorstellungsbilder bezüglich der »Digitalisierung«.

Von Seiten der Schulträger wird gleichzeitig viel Geld in die Hand genommen, um eine Ausstattung mit Informatiksystemen an den Schulen voranzutreiben, um die »Digitalisierung« in die Schulen zu transportieren. Das Informatiksystem wird in der Regel *benutzt*, um Inhalte anderer Fächer zu erarbeiten, er dient als reines Werkzeug.⁵ Dies ist nicht grundlegend falsch, greift aber zu kurz. Informatiksysteme müssen »entzaubert«, Konzepte, Ideen und

³ Laut ICILS-Studie [vgl. Ei17, S.15] wird der »digitalen« Technik zwar viel Potential zugesprochen, doch wird dieses Potential von den Schulen derzeit nicht ausgeschöpft, es *kann* ohne informatische Bildung nicht ausgeschöpft werden. Derzeit müssen Lehrkräfte an Grundschulen darauf vertrauen, dass Verlage und Unternehmen geeignete Lern- und Diagnoseprogramme entwickeln und diese Werkzeuge zur Verfügung stellen. Da jedoch die individuellen Lernwege der Kinder im Zentrum der pädagogischen Überlegungen der Lehrkräfte stehen, können diese Werkzeuge nicht umfänglich zielführend eingesetzt und an die Schüler*innen der eigenen Klasse angepasst werden. Somit schaffen sie zwar kurzfristig Entlastung, tragen aber nicht zur Lösung des Problems bei. Mehr noch: Es steht zu vermuten, dass individuelle Lernschwierigkeiten sich verschärfen, wenn sie nicht mehr von der Lehrkraft adressiert werden, sondern von einem Werkzeug, das unabhängig von der konkreten Lernsituation in der eigenen Lerngruppe »funktionieren« soll.

⁴ Nachzulesen in der ADO NRW, §11 und im Schulgesetz §17 LVO §57, Absatz 3

⁵ Diese Fehlvorstellung mahnt Eickelmann in der Auswertung der ICILS-Studie an und gibt zu bedenken, dass Ausstattungsprogramme ohne fundiertes pädagogisches Konzept nicht oder nur selten zu nachhaltigen Veränderungen des Lernens führen [vgl. Ei17, S.16].

Grundlagen thematisiert und beherrscht werden, *bevor* sie als Werkzeuge kompetent und mündig eingesetzt werden können.

2.2 KMK-Strategie konkret am Beispiel des Medienkompetenzrahmens Nordrhein-Westfalen

In der KMK-Strategie »Bildung in der digitalen Welt« werden sechs Kompetenzbereiche formuliert, die in den Bundesländern umzusetzen sind [vgl. KM16]. In Nordrhein-Westfalen werden diese Kernbereiche im Medienkompetenzrahmen für alle Schulstufen aufgegriffen und spezifiziert. Die ursprünglichen Vorgaben zur Querschnittsaufgabe »Medienkompetenz« wurden überarbeitet und um eine sechste Spalte ergänzt. Dort werden unter der Kompetenz »Problemlösen und Modellieren« die Teilkompetenzen »Prinzipien der digitalen Welt«, »Algorithmen erkennen«, »Modellieren und Programmieren« und »Bedeutung von Algorithmen« ausgewiesen und knapp vertieft [vgl. MS18]. Hier ist die Informatik klar als Bezugswissenschaft zu identifizieren.

Mit der Überarbeitung des Medienkompetenzrahmens ist die Tür zur informatischen Bildung in der Grundschule aufgestoßen worden. Darüber hinaus soll der Medienkompetenzrahmen auch als Basis für die Anpassung der Kerncurricula in der Lehrer*innenbildung verstanden werden. Es existieren also bereits verpflichtende informatische Bildungsbestandteile für die Grundschule. Es geht daher sowohl aus entwicklungspsychologischer als auch aus schulpolitischer Sicht nicht mehr um die Frage, *ob* Informatik in die Grundschule soll, sondern nur noch darum *wie*.⁶

3 Stellschraube Lehrerbildung

Um vor diesem Hintergrund sicher zu stellen, dass in den Grundschulen Personal zur Verfügung steht, um den notwendigen Informatikunterricht zu erteilen, liegt eine verpflichtende Verankerung von Informatik in den universitären Lehrangeboten für Grundschullehrkräften nahe. Blömeke [vgl. BKL10] identifiziert in ihren fachübergreifenden Ausführungen zu der Studie TEDS-M7 die starke Überschneidung von Lehrer- und Schülerkompetenzen. Dass der Weg Lehrer*innenbildung-Lehrer*innenhandeln-Lernergebnisse der Schüler*innen eine Wirkungskette darstellt, wird unter anderem von Terhart [Te06] beschrieben. Die curriculare Verankerung entsprechender Elemente in der Lehrerbildung (für die Grundschule leider immer noch überwiegend LehrerINNENbildung) kann dazu beitragen, dass Hürden auf Seiten der Lehrkräfte abgebaut werden und das fachliche Selbstkonzept bezüglich Informatik

⁶ Leider wird die Frage nach dem *wie* in Nordrhein-Westfalen jedoch nicht VOR der verpflichtenden Einführung des Medienkompetenzrahmens beantwortet. Vielmehr sind alle einzelnen Schulen dazu aufgefordert, standortbezogen eigene Medienkonzepte zu entwickeln. Es ist zu überlegen, die Schulen bei dieser Aufgabe kompetent zu unterstützen, indem beispielsweise ein allgemeines Medien- und Informatikkonzept zur Verfügung gestellt wird, welches alle Aspekte der KMK-Strategie abdeckt und durch die Schulen entsprechend angepasst und modifiziert werden kann.

⁷ »Teacher Education and Development Study«, Internationale Vergleichsstudie zur Effektivität des deutschen Lehrerausbildungssystems [BKL10]

positiv verändert wird. Lehrkräfte wiederum, die keine Hemmungen gegenüber dem Fach und dem eigenen Können haben UND die in Informatik gebildet sind, könnten dazu beitragen, das Spannungsfeld Informatik-Bildung-Schule-Gender positiv zu gestalten und ihre Kompetenz für den Bildungserfolg der Kinder, insbesondere der Mädchen, gewinnbringend einzusetzen.

Qualifizierte Lehrkräfte können den Herausforderungen informatischer Bildung kompetent und verantwortungsbewusst begegnen. Damit schafft die universitäre Lehrerbildung die notwendigen Voraussetzungen und muss als mögliche Stellschraube identifiziert werden.

4 Lehrveranstaltung »Informatik für die Grundschule«

Noch ist Informatik weit davon entfernt, ein verpflichtender Bestandteil der **Grundschullehrerbildung** zu sein. Weder die Fachdidaktiken noch die angehenden Grundschullehrkräfte sind auf diese Aufgabe vorbereitet. Best kann in den Zwischenergebnissen seiner Forschung zwar informatische Präkonzepte bei den interviewten Grundschullehrkräften identifizieren [vgl. Be17b], Dengel und Heuer berichten jedoch, dass Grundschullehrkräfte häufiger falsche Vorstellungsbilder zur Informatik haben als Lehramtstudierende anderer Schularten [vgl. DH17]. Sie gehen sogar so weit, von einem »speziellen Wissensdefizit bei angehenden Grundschullehrkräften« zu sprechen [vgl. DH17, S.95]. Darüber hinaus weisen Dengel und Heuer darauf hin, dass Menschen mit falschen Vorstellungen bezüglich der Informatik später aufgenommenes, korrektes Wissen nicht in das eigene Vorstellungsbild integrieren können, da eine Assimilation bei sich widersprechenden Vorstellungsbildern unmöglich ist und die Fehlvorstellungen somit persistent sind [vgl. ebenda]. Diese Persistenz muss bei der Konzeption einer Lehrveranstaltung von Informatikdidaktikern berücksichtigt werden, um nachhaltige informatische Kompetenz zu ermöglichen. Während Lehrkräfte an weiterführenden Schulen sich für das Fach Informatik aktiv entscheiden und die universitäre Lehre an ein grundlegendes Interesse anknüpfen kann, ist die Ausgangslage bei Grundschullehrkräften eine völlig andere. Grundschulinformatikdidaktik ist bislang auch kein durchgängiger Bestandteil der Informatikfachdidaktik und wird fast ausschließlich von Menschen beforscht, die Informatik studiert haben – und kein Grundschullehramt. Sie kennen weder Bedingungsgefüge in Schulen, noch die *Zielgruppe* (d. h. nicht nur ein einzelnes Kind) im Klassenverband. Während Nicht-Informatiker oft ein verzerrtes Bild von Informatik haben, haben Nicht-Grundschulpädagogen oft ein verzerrtes Bild von Grundschulen – und auch von Grundschullehrkräften. Beide Perspektiven zu vereinen kann nur im gegenseitigen Austausch auf Augenhöhe gelingen. Einen ersten Ansatz kann das vierstündige Seminar »Informatik für die Grundschule« an der Universität Wuppertal bieten, das im Curriculum verankert wurde und 6 Leistungspunkte für den Optionalbereich der Bildungswissenschaften abdeckt.

Von der Annahme ausgehend, dass Wissen nicht in bestimmten Alterstufen, sondern in bestimmten Kompetenzstufen erworben wird, ist die Veranstaltung für absolute Informatiknovizen konzipiert. Selbstverständlich sind Tempo, Komplexität, Abstraktion, Arbeitsweisen und Theoriegehalt auf Studierende abgestimmt, doch zeigt die vergleichende Erfahrung aus der Arbeit mit den drei Zielgruppen »Kinder«, »Lehrkräfte« und »Studierende«, dass die

Nutzungskompetenz zwar variiert, die informatische Kompetenz jedoch generationsübergreifend auf einem ähnlichen Niveau zu verorten ist.

Es wurde darüber hinaus mit in die Überlegungen einbezogen, dass es den zukünftigen Lehrkräften möglichst leicht gemacht wird, die erarbeiteten Inhalte in der Grundschule tatsächlich einzusetzen. Grundschullehrkräfte erteilen bereits in mehr Fächern Fachunterricht, als sie studiert haben. Sie sind darauf angewiesen, sich auch fachfremde Gegenstände für den Unterricht eigenständig zu erschließen. In der Regel können sie dabei auf ihre in der eigenen Biografie erworbenen Fachkonzepte zurückgreifen und daher einen Einstieg in neue Gegenstände der jeweiligen Disziplin finden und kreativ eigenes Unterrichtsmaterial zu konzipieren. Für die Informatik ist bei der Zielgruppe jedoch überwiegend entweder kein, oder ein falsches Fachkonzept zu identifizieren. Daher muss es einen engen Rahmen geben, innerhalb dessen Studierende ihr fachliches Selbstkonzept hinterfragen und anpassen können, grundlegende Ideen kennenlernen und den Transfer für den Grundschulunterricht bereits vorbereitet finden. Die curricularen Entscheidungen zur Lehrveranstaltung orientieren sich an den Kompetenzempfehlungen der Gesellschaft für Informatik für die Grundschule [GI19]. Die in der Veranstaltung berücksichtigten informatischen Gegenstände genügen den Kriterien, die durch die Methode der fundamentalen Ideen als durchaus grundlegende Elemente der Informatik identifiziert werden können. Die in Tabelle 1 aufgeführten Gegenstände wurden für die Veranstaltung für fachlich nicht vorbereitete Studierende aufgearbeitet, didaktisch reduziert und bilden den Ausgangspunkt für den Aufschluss jeden Lerngegenstandes, sowohl *mit* als auch *über* das Material.

In der vierstündigen Veranstaltung wechseln sich theoretische und praktische Arbeitsphasen ab, d. h. nach dem theoretischen Einstieg z. B. in einen Gegenstand der Kryptologie, werden passend Unterrichtsmaterialien für die Grundschule vorgestellt, erprobt, diskutiert und zurückbezogen auf den theoretischen Ausgangspunkt. Theorie und Praxis ergänzen sich und adressieren die Grundschule.

Die eingesetzten Unterrichtsmaterialien aus dem Grundschulkontext werden von allen Seminarteilnehmern jeweils in einem Klassensatz erstellt und wandern zusammen mit den theoretischen Ausführungen in eine persönliche »Informatikbox«, angelehnt an das Konzept der »Klassenkisten« zum Sachunterricht der Westfälischen Wilhelms Universität Münster [vgl. MST08]). Die in der Veranstaltung gelieferten Definitionen, Erklärungen und Erläuterungen werden auf Wortspeicherkarten verschriftlicht und für den Einsatz im Klassenraum aufbereitet. Darüber hinaus legen die Studierenden ein Glossar an, das sie fortlaufend ergänzen. Passend zum theoretischen Input und zum vorgestellten Material werden von den Studierenden Unterrichtsentwürfe entwickelt, die sich an den Vorgaben der Studienseminare orientieren. Diese Unterrichtsentwürfe werden in den letzten Veranstaltungen im Seminar erprobt und diskutiert. Nach der abschließenden Überarbeitung landen alle Materialien und Unterrichtsentwürfe ebenfalls in der »Informatikbox«. Diese enthält also im Anschluss an das Seminar eine einsatzfähige Materialausstattung für fundierten Grundschulinformatikunterricht, auf die die Studierenden viel Mühe verwendet haben, welche sie erprobt und diskutiert haben und zu der sie über das theoretische Wissen verfügen. Die Begleitforschung zu den Sachunterrichtskisten aus Münster zeigt, dass solcherart einsatzbereite Boxen im

späteren Unterricht auch eingesetzt werden [vgl. MST08]).

Fachgebiet	1. Durchlauf SoSe18	2. Durchlauf WiSe18/19
Technische Informatik	Rechneraufbau Hardwarekomponenten Festplatte öffnen Netzwerke Internet Calliope	Rechneraufbau Hardwarekomponenten Internet
Theoretische Informatik	Kryptologie Binärdarstellung ASCII Kontrollstrukturen Rekursion	Kryptologie Binärdarstellung Kontrollstrukturen
Praktische Informatik	Dijkstra- Algorithmus Routenplanung Suchalgorithmen Sortieralgorithmen	Binäre Suche Bubble-Sort
Angewandte Informatik	Weg einer E-Mail E-Mail-Sicherheit Strukturierung von Daten Betriebssysteme	Weg einer E-Mail E-Mail-Sicherheit
Informatik, Mensch und Gesellschaft sowie Didaktik	Modellierung Phänomenbereiche Fundamentale Ideen Informatik im Alltag Gesellschaftlicher Diskurs Informatik in Grundschulen Geschichte der Informatik	Modellierung Phänomenbereiche Informatik im Alltag Informatik in Grundschulen Geschichte der Informatik

Tab. 1: Gegenstände im ersten und zweiten Durchlauf

5 Darstellung erster Erkenntnisse

Nach dem ersten Veranstaltungsdurchlauf mit 10 Teilnehmer*innen (8 weiblich, 2 männlich) hatten alle Studierenden des Seminars deutliche Hürden bezüglich der Informatik abgebaut. Die Aussage: »Das ist Informatik? Cool!« könnte sinnbildlich stehen für den A-Ha-Effekt, der sich bei ihnen einstellte. Alle Teilnehmer*innen berichteten im Anschluss von einem veränderten Selbst- und Fachkonzept, von einer überwundenen Hürde und einer veränderten Einstellung zum Thema. Nach dem Erfolg des ersten Durchgangs wurde das Seminar entgegen der ursprünglichen Planung einer jährlichen Durchführung im folgenden Semester direkt wieder angeboten. Die Teilnehmerzahl wurde im zweiten Durchgang erhöht und 20 Plätze konnten an 17 weibliche und 3 männliche Teilnehmer*innen vergeben werden. Um allen Studierenden die Möglichkeit einzuräumen, in Partnerarbeit einen Unterrichtsentwurf

zu einem ausgewählten Thema zu erstellen und sich intensiv mit einem selbst gewählten Gegenstand der Informatik auseinanderzusetzen, musste der theoretische Input und der Umgang mit dem Material eingeschränkt werden (vgl. Tabelle 1). Gekürzt wurden Inhalte, bei denen vermutet wurde, dass sich der Lebensweltbezug für Grundschul Kinder erst auf den zweiten Blick oder über Umwege erschließt.

Während bei 10 Personen der persönliche Kontakt gewährleistet und das Lerntempo sehr langsam war (was mitunter auch eine 1 zu 1 – Betreuung ermöglichte), fiel diese enge Betreuung bei 20 Teilnehmer*innen deutlich geringer aus. Neben der engen Anbindung und der Möglichkeit, sich der »Informatik« langsam und in einem kleinen, geschützten Rahmen zu nähern, ist mit dieser Einschränkung auch die Möglichkeit erschwert, Fehlkonzepte bei den Studierenden zu identifizieren und zu korrigieren und auf Fragen, Anregungen und Ängste bezüglich der Informatik zugewandt einzugehen.

Erste Ergebnisse im Zusammenhang mit den durchgeführten Veranstaltungen können in fünf Punkten zusammenfassend dargestellt werden:

- Ängste und Unsicherheit, bzw. falsche oder fehlende Vorstellungen sind durchgängig und bislang ohne Ausnahme bei allen Studierenden in beiden Durchläufen zu beobachten. Es kann nur zum Teil davon gesprochen werden, dass *neue* Gegenstände vermittelt werden. Vielmehr geht es häufig zunächst darum *falsche* Vorstellungen zu revidieren, *bevor* mit der Entwicklung informatischer Kompetenz überhaupt begonnen werden *kann*.
- Der Lernprozess ist sehr zeitaufwändig und erfordert eine sehr enge Betreuung.
- Viele Gegenstände und Inhalte sind für die Studierenden zunächst fremd oder zum Teil negativ konnotiert.
- Eine in Teilnehmerzahl, Tempo und Niveau reduzierte Veranstaltung erreicht es besser, Studierende für die Gegenstände zu gewinnen, als eine gestraffte Veranstaltung für mehr Studierende.
- Die Hinwendung zum einzelnen Studierenden erlaubt die nötige Korrektur der fachlichen Selbsteinschätzung bezüglich der Informatik und eine stabile Orientierung zur Überwindung der eigenen Unsicherheiten.
- Der Schritt, die Teilnehmerzahlen zu verdoppeln, kann als nicht zielführend angesehen werden.
- Die Reduktion des fachlichen Inputs wirkte sich auf die Veranstaltungstiefe negativ aus.

6 Weiterführende Fragen

Durch die dargestellten Rahmenbedingungen und die ersten explorativ gewonnenen Ergebnisse werden eine Reihe von Fragen aufgeworfen, die in dem weiteren Forschungsprozess

adressiert werden müssen. Es gilt zunächst herauszufinden, ob diese Ergebnisse sich stabil abbilden. In einem weiteren Schritt muss abgewogen werden, welche Priorität zu setzen ist: Ist die exclusive Veranstaltung für nur 10 Studierende mit besseren Ergebnissen dem Seminar für 20 Teilnehmer*innen vorzuziehen? Welche Kompetenzen müssen zukünftige Grundschullehrkräfte bezüglich der Informatik in der Grundschule erwerben? . . . und wie kann das überprüft werden? Können wir davon ausgehen, dass auf diesem Weg Informatik überhaupt zukünftig Eingang in den Unterricht dieser Lehrkräfte findet? Kann eine Langzeitbeobachtung eventuell Aufschluss geben? Dürfen die technischen Möglichkeiten der Informatik als zentrales Element zugunsten der Ideenwelt der Informatik zunächst ausgeklammert werden?

Es zeichnet sich ab, dass die Studierenden beides benötigen: Einblick in Konzepte und Ideen UND Einblick in die Systeme. Eine einzelne Lehrveranstaltung in oben angegebener Form kann diesen Spagat nicht leisten. Soll Informatik in der Breite wirken, sollen also alle Studierenden des Grundschullehramtes über fundierte Kenntnisse verfügen, ist über andere universitäre Veranstaltungsformate nachzudenken.

Werden sich die Ergebnisse aus den bisher untersuchten Kontexten durch weitergehende quantitative und qualitative Studien verstetigen / bestärken, wird auch zu prüfen sein, ob (und wie) Seiteneffekte für die Entwicklung der informatischen Bildung der Schülerinnen und Schüler festzustellen sind. In diesem Fall könnte es angezeigt sein, die Umsetzung der Gestaltung der informatischen Bildung z. B. bzgl. der Gruppengröße in den Blick zu nehmen. Zudem muss vor dem Hintergrund dieser Ergebnisse überlegt werden, wie und ob Schüler*innen einen ähnlichen Kompetenzzuwachs in den üblichen großen Lerngruppen überhaupt erreichen können und welche Bedingungen neben fundierter Lehrerausbildung im Schulkontext zusätzlich erfüllt sein müssen.

Literatur

- [Be17a] Bergner, N.; Köster, H.; Magenheimer, J.; Müller, K.; Romeike, R.; Schroeder, U.; Schulte, C.: Zieldimensionen für frühe informatische Bildung im Kindergarten und in der Grundschule. In (Diethelm, I., Hrsg.): Informatik im Unterricht – so geht’s. LectureNotes in Informatics(LNI)-Proceedings, Gesellschaft für Informatik e.V. (GI), Bonn, S. 53–62, Sep. 2017, ISBN: 978-3-88579-668-8, URL: <https://metager.to/9188->, Stand: 17.05.2019.
- [Be17b] Best, A.: Bild der Informatik von Grundschullehrpersonen. Erste Ergebnisse aus qualitativen Einzelfallstudien. In (Diethelm, I., Hrsg.): Informatik im Unterricht – so geht’s. LectureNotes in Informatics(LNI)-Proceedings, Gesellschaft für Informatik e.V. (GI), Bonn, S. 83–86, Sep. 2017, ISBN: 978-3-88579-668-8, URL: <https://metager.to/9188->, Stand: 17.05.2019.
- [BK10] Becker, R.; Kortendieck, B., Hrsg.: Handbuch Frauen- und Geschlechterforschung, VS Verlag für Sozialwissenschaften der Springer Fachmedien, 2010, ISBN: 978-3-531-17170-8.

- [BKL10] Blömeke, S.; Kaiser, G.; Lehmann, R., Hrsg.: TEDS-M 2008 Primarstufe: Ziele, Untersuchungsanlage und zentrale Ergebnisse, Professionelle Kompetenz und Lerngelegenheiten angehender Primarstufenlehrkräfte im internationalen Vergleich, Waxmann Verlag, 2010, S. 11–38, ISBN: 978-3-8309-2281-0.
- [DH17] Dengel, A.; Heuer, U.: Aufbau des Internets: Vorstellungsbilder angehender Lehrkräfte. In (Diethelm, I., Hrsg.): Informatik im Unterricht – so geht’s. LectureNotes in Informatics(LNI)-Proceedings, Gesellschaft für Informatik e.V. (GI), Bonn, S. 87–97, Sep. 2017, ISBN: 978-3-88579-668-8, URL: <https://metager.to/9188->, Stand: 17.05.2019.
- [Di17] Diethelm, I., Hrsg.: Informatische Bildung zum Verstehen und Gestalten der digitalen Welt, INFOS 2017, 17. GI-Fachtagung Informatik und Schule, 13.–15. September 2017, Oldenburg, Germany, LectureNotes in Informatics(LNI)-Proceedings, Bonn: Gesellschaft für Informatik e.V. (GI), 2017, ISBN: 978-3-88579-668-8, URL: <https://metager.to/9188->, Stand: 17.05.2019.
- [Ei17] Eickelmann, B.: Konzepte und Entwicklungsperspektiven Kompetenzen in der digitalen Welt. 2017.
- [Fr14] Fraillon, J.; Ainley, J.; Schulz, W.; Friedman, T.; Gebhardt, E.: Preparing for Life in a Digital Age – The IEA International Computer and Information Literacy Study (ICILS) International Report. Springer Open, Melbourne, Australia, 2014.
- [GI19] Gesellschaft für Informatik e. V., Hrsg.: Kompetenzen für informatische Bildung im Primarbereich, Empfehlungen der Gesellschaft für Informatik e. V. Erarbeitet vom Arbeitskreis „Bildungsstandards Primarbereich“ – Beschluss des GI-Präsidiums vom 31. Januar 2019 – wird in gedruckter Form der LOG IN 39 (2019) Heft 191/192 beigelegt, 7. Feb. 2019, URL: <http://uni-w.de/1gm>, Stand: 09.05.2019.
- [KM16] KMK, Hrsg.: Kompetenzen in der digitalen Welt, Kompetenzbereiche, Dez. 2016, URL: <https://metager.to/r6tqb>, Stand: 17.05.2019.
- [MS18] MSB-NW, Hrsg.: Medienkompetenzrahmen NRW – Webseite, Ministerium für Schule und Bildung des Landes Nordrhein-Westfalen (MSB-NW), 2018, URL: <https://t1p.de/00on>, Stand: 11.05.2019.
- [MST08] Möller, K., Hrsg.: Lernen mit der Klasse(n)kiste, Ergebnisse einer Befragung von grundschullehrkräften, Westfälische Wilhelms Universität, 2008, ISBN: 978-3-00-025409-3.
- [Te06] Terhart, E.: Universität und Lehrerbildung: Perspektiven einer Partnerschaft. In: Lehrerbildung im Diskurs. S. 20–39, 2006.
- [Wa17] Wagner, P., Hrsg.: Handbuch Inklusion, Grundlagen vorurteilsbewusster Bildung und Erziehung. Herder Verlag, Freiburg, 2017.

Interesse an Informatik und Informatikselbstkonzept zu Beginn der Sekundarstufe I des Gymnasiums

Kathrin Haselmeier¹, Ludger Humbert², Klaus Killich³, Dorothee Müller⁴

Abstract: Die Kompetenzentwicklung bei Schülerinnen und Schülern hängt deutlich von dem fachlichen Interesse und dem fachlichen Selbstkonzept ab. Im Rahmen eines forschend begleiteten Modellvorhabens zur Informatik in der Erprobungsstufe des Gymnasiums wurden zu Beginn des Schuljahres 2018/19 über 2 500 Schülerinnen und Schüler der Jahrgangsstufe 5 befragt. Die leitende Forschungsfrage dieser Untersuchung war: Welche motivationalen Merkmale in der Ausprägung *Interesse an Informatik* und *Informatikselbstkonzept* haben Schülerinnen und Schüler zu Beginn des Modellvorhabens am Anfang der Jahrgangsstufe 5?

Die Ergebnisse zeigen, dass bei den Befragten im Durchschnitt das Interesse an Informatik sehr hoch und ihr fachliches Selbstkonzept bezüglich der Informatik positiv ist. Neben diesen, die gesamte Stichprobe betreffenden Ergebnissen wird die Befragung auch unter den beiden Aspekten geschlechtsbezogene Disparität und Wirkung von bereits besuchtem Informatikunterricht differenziert untersucht. Bei den Schüler^{innen} ist sowohl bezüglich des Interesses an Informatik wie des Informatikselbstkonzeptes eine signifikant stärkere Unsicherheit zu erkennen als bei männlichen Schülern. Die Unterschiede zwischen Befragten, die angeben, bereits Informatikunterricht gehabt zu haben, und denjenigen ohne Informatikunterricht sind gering. Dabei weist die Gruppe mit Informatikunterricht ein etwas höheres Interesse am Informatik und ein leicht positiveres Informatikselbstkonzept auf.

Keywords: Einstieg Informatik; Informatikselbstkonzept; Fachliches Interesse Informatik

1 Forschungsstand und Forschungsziele

1.1 Fachliches Interesse und fachliches Selbstkonzept

In der empirischen Bildungsforschung stehen zunehmend motivationale und affektive Einflussfaktoren im Fokus der Forschung. Die Untersuchung dieser Faktoren gewinnt um so mehr an Wichtigkeit, je mehr die Forschungsergebnisse einen deutlichen Einfluss auf die Kompetenzentwicklung und den Lernerfolg belegen. Vor allem die große Bedeutung der motivationalen Faktoren fachliches Interesse und fachliches Selbstkonzept für die schulische Leistungsentwicklung ist unbestritten und für einige Fächer – in erster Linie für mathematisch-naturwissenschaftliche Fächer und Deutsch – durch empirische Untersuchungen belegt [vgl.

¹ Bergische Universität Wuppertal (BUW), Didaktik der Informatik (DdI), Gaußstr. 20, 42 119 Wuppertal (BUW), khaselmeier@uni-wuppertal.de

² BUW – DdI humber@uni-wuppertal.de

³ Bezirksregierung Düsseldorf, Am Bonnheshof 35, 40 474 Düsseldorf, klaus.killich@brd.nrw.de

⁴ BUW – DdI, dmueller@uni-wuppertal.de

z. B. He11; KH06; Kö06; Le08]. Fachliches Interesse und fachliches Selbstkonzept stehen in Wechselwirkung zueinander [vgl St17].

Das *fachliche Interesse* – d. h. im Schulkontext das Interesse an einem Schulfach – manifestiert sich in positiven Emotionen gegenüber den Gegenständen des Faches und in seiner allgemeinen Wertschätzung [vgl. Sc09]. Im Grundschulalter ist das fachliche Interesse im Allgemeinen hoch. Es differenziert und verfestigt sich in der Sekundarstufe I [vgl. Pa13, S. 361]. Das fachliche Interesse ist u. a. entsprechend der geschlechtsstereotypischen Fächerzuordnung ausgeprägt [Le08]. So zeigt sich bei Wahlfächern, dass Fächer aus dem männlich konnotierten MINT-Bereich⁵ von Mädchen seltener gewählt werden [vgl. z. B. MS18a].

Als *fachliches Selbstkonzept* werden die auf ein spezielles Schulfach oder auf einen Kompetenzbereich bezogenen Einschätzungen der eigenen Kompetenzen und Möglichkeiten verstanden. Seine starke Wirkung auf die Kompetenzentwicklung lässt es als „selbsterfüllende Prophezeiung“ [vgl. He11] erscheinen. In der Bildungswissenschaft wird die Förderung eines adäquaten und positiven Selbstkonzeptes daher oft sogar als das zentrale Ziel pädagogischer Bemühungen gewertet [MT09, S. 196]. Das individuelle fachliche Selbstkonzept einer Schülerin oder eines Schülers und ihre/seine tatsächlichen fachbezogenen Kompetenzen und Möglichkeiten entsprechen sich nicht. Wie das fachliche Interesse ist auch das fachliche Selbstkonzept in der Primarstufe eher hoch, differenziert sich in der Sekundarstufe I aus und fällt dabei im Durchschnitt ab [vgl. Pa13, S. 361]. Männliche Schüler verfügen im Allgemeinen, vor allem in den Fächern mit männlicher Geschlechtszuschreibung, bei gleicher Kompetenz bereits in der Grundschule über ein deutlich höheres, oft überhöhtes, fachliches Selbstkonzept als Schülerinnen, die häufig ein zu niedriges fachliches Selbstkonzept besitzen. In der Sekundarstufe I verstärkt sich die geschlechtsbezogene Diskrepanz des fachlichen Selbstkonzeptes [KH06].

1.2 Interesse an Informatik und Informatikselbstkonzept

Die Entwicklung eines hohen Interesses und positiven fachlichen Selbstkonzeptes bezüglich Informatik ist – nicht nur bei Schülerinnen und Schülern – durch weitverbreitete Fehlvorstellungen [vgl. K113] zur Informatik behindert. Zum Interesse an Informatik und zum Informatikselbstkonzept von Schülerinnen und Schülern gibt es kaum empirische Untersuchungen. Bei Befragungen von Studierenden, die retrospektiv den eigenen Informatikunterricht ihrer Schulzeit betrachten [vgl. z. B. Kn11; Ku08; Mü17], finden sich deutliche Hinweise auf Hinderung der Entwicklung des Interesses an Informatik bzw. am Informatikunterricht durch tradierte Fehlvorstellungen zu Informatik (z. B. Männerfach, Nerdfach, reine Computerwissenschaft).

Die frühe und anhaltende Einordnung des Schulfaches Informatik als „Männerfach“ hat Einfluss auf das informatische Selbstkonzept der Schülerinnen. Mädchen entdecken im Gegensatz zu Jungen oft erst im Informatikunterricht die Informatik für sich [vgl. Ku08, S. 120].

⁵ MINT ist ein Akronym aus Mathematik, Informatik, Naturwissenschaften und Technik

Da Informatik in den meisten Bundesländern kein Pflichtfach ist, entfällt diese Möglichkeit für viele Mädchen. Das Kurswahlverhalten belegt eine durch Geschlechtsstereotypen geprägte Interessenverteilung, die eine noch stärkere geschlechtsbezogene Diskrepanz aufweist als bei den anderen MINT-Fächern. So lag z. B. in Nordrhein-Westfalen im Schuljahr 2017/18 der Anteil der weiblichen Teilnehmenden an den Grundkursen Informatik bei 25,39 % und in den Leistungskursen bei 15 % [vgl. z. B. MS18a, S. 106]. Die Zahlen der vorangegangenen Jahre waren ähnlich.

Anders als die erwähnten qualitativen, retrospektiven Untersuchungen durch Befragung von Studierenden, untersucht die 2013 durchgeführte ICILS-Studie Schülerinnen und Schüler der achten Jahrgangsstufe. Sie thematisierte mit dem „computerbezogenen Selbstkonzept“ und den „computerbezogenen Einstellungen“ [vgl. Lo14, S. 263 f] dem Informatikselbstkonzept und dem Interesse an Informatik nahestehende Themen. Ein Ergebnis der ICILS-Studie 2013 ist, dass Mädchen in Deutschland wie auch in den meisten anderen an der Studie teilnehmenden Ländern „über signifikant höhere computer- und informationsbezogene Kompetenzen verfügen als Jungen“, während ihr entsprechendes Selbstkonzept unterdurchschnittlich ist [vgl. Ei17, S. 14]. In dieser untersuchten Altersgruppe sind Interessen und fachliches Selbstkonzept schon weitgehend ausdifferenziert und oft verfestigt, so dass die vorangegangenen diesbezüglichen Entwicklungen von Interesse sind. Die Entwicklung des Informatikselbstkonzeptes und des Interesses bezüglich des Faches Informatik in der Übergangsphase zwischen Primarstufe und der Sekundarstufe I stellt ein Forschungsdesiderat dar. Da sich fachliches Interesse und Selbstkonzept vor allem in der Sekundarstufe I ausdifferenzieren und festigen, ist genau dieser Zeitpunkt in der Bildungsentwicklung und Kompetenzentwicklung für das Schulfach Informatik bedeutsam.

2 Forschungsfeld, -fragen und -durchführung

In einem vierjährigen Forschungsprojekt werden 3 Jahrgänge von Schülerinnen und Schülern, die von der fünften bis zur sechsten Jahrgangsstufe an Informatikunterricht teilnehmen, forschend begleitet. Im ersten Teilprojekt, über das in diesem Beitrag berichtet wird, werden Interesse und Selbstkonzept bezüglich der Informatik zu Beginn der fünften Jahrgangsstufe untersucht.

Forschungsumfeld

Die Bezirksregierung Düsseldorf setzt ab dem Schuljahr 2018/19 über 4 Jahre ein Modellvorhaben um. Das Unterrichtsfach Informatik wird in diesem Modellvorhaben in den Klassen 5 und 6 als Pflichtfach für alle Schülerinnen und Schüler implementiert (Tab. 1). Es sind zunächst 72 Gymnasien der Regierungsbezirke Düsseldorf und Köln beteiligt. Die Schulen nehmen jeweils mit einer, mehreren oder allen Klassen der Jahrgangsstufe 5 teil. In den beiden folgenden Schuljahren werden sich darüber hinaus weitere Schulen an dem Vorhaben beteiligen. Der Unterricht findet im Umfang von einer Wochenstunde auf der Basis eines verbindlichen didaktischen Konzeptes statt. Die Schulen stellen je eine Ergänzungsstunde für den Informatikunterricht der jeweiligen Klassen in den Jahrgangsstufen 5 bzw. 6 zur

Verfügung. Das Modellvorhaben wird durch die Informatikdidaktikgruppe der Bergischen

Schuljahr	Klasse(n) der Jahrgangsstufe 5	Klasse(n) der Jahrgangsstufe 6
2018/19	X	
2019/20	X	X
2020/21	X	X
2021/22		X

Tab. 1: Zeitlicher Ablauf des Modellvorhabens

Universität Wuppertal forschend begleitet. Mit dem Modellvorhaben eröffnet sich die Möglichkeit, die informatikbezogenen motivationalen Tendenzen von Schülerinnen und Schülern zum Zeitpunkt des Übergangs von der Grundschule in die Sekundarstufe I zu erfassen, sowie deren Entwicklung unter dem Einfluss von Informatikunterricht über zwei Jahre forschend zu begleiten.

Forschungsfrage(n)

Die im Folgenden dokumentierten Ergebnisse der quantitativen, empirischen Untersuchungen geben Hinweise zur Beantwortung der Frage:

Welche motivationalen Merkmale in der Ausprägung Informatikselbstkonzept und Interesse an Informatik haben Schülerinnen und Schüler zu Beginn der Jahrgangsstufe 5 des Gymnasiums?

Von besonderem Interesse sind Bezüge zum Geschlecht der Befragten und zum Einfluss von bereits besuchtem Informatikunterricht, wodurch sich zu der Forschungsfrage folgende Unterfragen ergeben:

1. Bestehen geschlechtsbezogene Diskrepanzen bezüglich dieser Merkmale?
2. Hat zuvor besuchter Informatikunterricht Einfluss auf diese Merkmale?

Stichprobe und Befragung

Zu Beginn des Schuljahres wurden die Klassen der Jahrgangsstufe 5, die im Rahmen des Modellvorhabens mit Informatik im Gymnasium beginnen, aufgefordert, an einer Online-Befragung teilzunehmen. Es haben insgesamt 2 570 Schülerinnen und Schüler den Fragebogen vollständig bearbeitet.

Die Befragungen sollten in der knappen Informatikunterrichtszeit durchgeführt und von den Schülerinnen und Schülern möglichst selbstständig bearbeitet werden. Es wurden die folgenden Rahmenbedingungen im Design der Befragung berücksichtigt: Die reine Bearbeitungszeit des Fragebogens sollte deutlich unter 20 Minuten liegen. Und die Fragen sollten kindgerecht formuliert und ohne Erklärungen der Lehrkraft verständlich sein.

Die Auswertung der von den Schülerinnen und Schüler bearbeiteten Fragen sollte Aufschluss zum Interesse bezüglich Informatik und zum Informatikselbstkonzept geben. Die Befragung ist so konzipiert, dass ebenfalls differenzierte Aussagen in Bezug auf diese motivationalen Merkmale unter den Aspekten möglicher geschlechtsbezogenen Disparität und der Wirkung von bereits besuchtem Informatikunterricht gemacht werden können.

Um die angestrebte Bearbeitungszeit zu realisieren und die Verständlichkeit für die Kinder zu erhöhen, wurden für die Datenerhebung statt der oft üblichen Likert-Skalen mit fünf oder sieben Antwortoptionen einfache, umgangssprachliche und meistens in der Ich-Form formulierte Aussagen genutzt, denen die Befragten in zwei Antwortoptionen zustimmen oder widersprechen konnten, wozu bei einigen Fragen noch die Option „Ich weiß nicht“ hinzugefügt wurde (s. z. B.: Abschnitt 3.3 – Frage zum Interesse am Informatikunterricht).

3 Ergebnisse

3.1 Geschlecht der Teilnehmenden

Die prozentuale Geschlechterverteilung entspricht mit 44,09 % weiblichen und 50,70 % männlichen Befragten (Tab. 2) nicht den zu erwartenden Werten. Denn in den Jahren von 2007 bis 2017 lag in Nordrhein-Westfalen der Mädchenanteil in der fünften Jahrgangsstufe des Gymnasiums bei ca. 52 % [vgl. MS18b, S. 45].

Mädchen	Jungen	keine Angabe
1133	1303	134
44,09 %	50,70 %	5,21 %

Tab. 2: Geschlechterverteilung der 2570 Befragten

Selbst die Annahme, dass ausschließlich Mädchen keine Angabe zum Geschlecht⁶ gemacht hätten, kann diese Abweichung nicht erklären. Eine mögliche Erklärung für diese Abweichung besteht darin, dass nach Auskunft/Angabe(n) einiger Schulen nicht alle Klassen der Jahrgangsstufe 5 an dem Modellvorhaben teilnahmen. Die Zuordnung ihrer Kinder zu den „Informatikklassen“ wurde nach Aussage der Schulen von Eltern oft ausdrücklich angestrebt. Damit zeigt sich bereits zu diesem frühen Zeitpunkt der Kompetenzentwicklung im Schulfach Informatik eine leichte Tendenz der Geschlechtersegregation.

⁶ Bei der Frage nach dem Geschlecht wurde den Befragten die neben den Antwortmöglichkeiten „*Ich bin ein Mädchen*“ und „*Ich bin ein Junge*“ mit „*Dazu möchte ich nichts sagen*“ als drittes auch die Möglichkeit keiner Angabe gegeben, wovon 5,21 % Gebrauch machten.

3.2 Informatikunterricht in der Grundschule

Die Aussage „*Ich hatte bereits in der Grundschule Informatikunterricht*“ bestätigen 10,5 % mit „Ja“ und verneinen 89,5 % mit der Auswahlmöglichkeit „Nein“. Die Frage nach dem „Informatikunterricht in der Grundschule“ hat das Ziel, bereits im schulischen Kontext erfahrene Gegenstände, die von den Schülerinnen und Schülern als zur Informatik gehörend eingeordnet werden, zu erfassen. Diese Zuordnung ist nicht in erster Linie von den erworbenen Kompetenzen, sondern vor allem von ihrer Einordnung durch die Lehrkraft der Grundschule abhängig. Solchermaßen zugeordnete Elemente des Unterrichts und die individuellen Lernerfahrungen mit ihnen werden auf ihren Einfluss sowohl auf das Interesse der Schülerinnen und Schüler an der Informatik und als auch auf ihr Informatikselbstkonzept untersucht.

3.3 Interesse an Informatik und Informatikselbstkonzept

Das Interesse an der Informatik wurde mit der einfachen Aussage „*Ich freue mich auf den Informatikunterricht*“ und den drei Wahloptionen „Ja“, „Nein“ und „*Ich weiß nicht*“ erhoben. Die in der Frage implizierten Vorstellungen der Schülerinnen und Schüler von Informatikunterricht und Informatik sind wahrscheinlich von großer Divergenz und von zahlreichen Fehlvorstellungen bestimmt (s. S. 2).

In der gesamten Stichprobe ist das Interesse sehr hoch: Es antworten 2316 der 2570 Befragten (90,12 %) mit „Ja“, 9,03 % mit „*Ich weiß nicht*“ und 0,86 % mit „Nein“ (Abb. 1a). Auffallend ist nicht nur der hohe Anteil derjenigen, die zustimmen, sondern auch, dass weniger als ein Prozent der Befragten mit „Nein“ antworten.

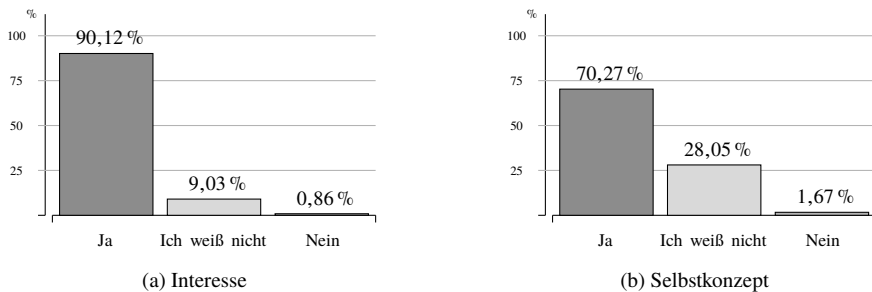


Abb. 1: Informatik: Interesse und Selbstkonzept

Auch das mit der Antwortaufforderung „*Ich erwarte, in Informatik gut zu sein*“ erhobene Informatikselbstkonzept ist positiv. 70,27 % der Befragten erwarten nach eigener Aussage „in Informatik gut zu sein“, 28,05 % geben an, sie wüssten es nicht, und 1,67 % antworteten, dass sie dies nicht erwarteten (Abb. 1b).

In diesem speziellen Fall – der deutlich überwiegende Teil der befragten Schülerinnen und Schüler hat zum Zeitpunkt der Befragung noch keine Erfahrungen mit eigenem Informatikunterricht – sind die positiven Erwartungen an das Fach Informatik deutlich höher als die Erwartungen an die eigenen diesbezüglichen Fähigkeiten. Die sehr positive Wertung, die dem Informatikunterricht entgegengebracht wird, verdeutlicht sich, wenn die Daten der Befragten zu Interesse und Informatikselbstkonzept in Beziehung gesetzt werden. Selbst von den 43 Schülerinnen oder Schülern, die explizit nicht erwarten, „in Informatik gut zu sein“, freut sich dennoch mit 28 der größte Teil auf den Informatikunterricht, 7 wissen es nicht und nur 8 freuen sich nicht.

Geschlechtsbezogene Diskrepanzen bei Interesse und Selbstkonzept bzgl. Informatik

Betrachtet man das Interesse an Informatik (Abb. 2a) und das Informatikselbstkonzept (Abb. 2b) bei Jungen und Mädchen unter dem Genderaspekt, so ist in beiden Gruppen ein sehr hohes Interesse und ein hohes Informatikselbstkonzept zu beobachten. Dennoch treten Unterschiede auf. Vor allem ist bei den weibliche Befragten eine stärkere Unsicherheit zu erkennen. Von den 1 133 befragten Mädchen freuen sich 88,44 % auf den Informatikunterricht, 0,53 % freuen sich nicht, aber 11,03 % wissen nicht, ob sie sich freuen sollen, und bringen damit ihre Unsicherheit bezüglich ihres Interesses an Informatik zum Ausdruck. Von den 1 303 befragten Jungen freut sich ein noch größerer Anteil, nämlich 92,86 %, auf den Informatikunterricht, 0,77 % freuen sich nicht und ein deutlich kleinerer Anteil als bei den Mädchen, nämlich 6,37 %, ist sich unsicher.

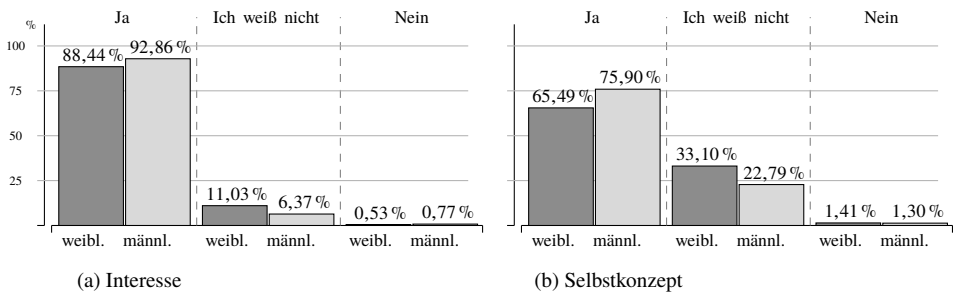


Abb. 2: Informatik: Vergleich Schülerinnen und Schüler

Noch ausgeprägter sind die geschlechtsbezogenen Diskrepanzen bei dem informatischen Selbstkonzept. Von den Mädchen erwarten 65,49 % „gut in Informatik zu sein“, 1,44 % erwarten dies nicht und 33,10 % wissen es nicht. 75,9 % der befragten Jungen, also deutlich mehr als bei den Mädchen, erwarten in „gut in Informatik zu sein“. 1,3 %, ein etwas kleinerer Anteil als bei den Mädchen, erwarten dies nicht. Die Unsicherheit bezüglich des Informatikselbstkonzeptes, „Ich weiß nicht“, ist mit 22,97 % deutlich geringer.

Aus dem Informatik-Biber 2015 wurde eine für die Jahrgangsstufen 5 und 6 als leicht eingestufte Aufgabe zur Mustererkennung als ein Item der Befragung übernommen. Diese Aufgabe wird insgesamt von 92,8 % der Befragten richtig beantwortet. Bei den befragten

Mädchen ist ein deutlich niedrigeres Informatikselbstkonzept vorhanden als bei den Jungen. Dennoch ist der Anteil der richtigen Antworten bei den Mädchen mit 93,65 % nicht niedriger, sondern geringfügig größer als bei den Jungen mit 92,63 %. Dies kann als einen – in weiteren Untersuchungen noch zu überprüfenden – Hinweis auf eine Bestätigung der in der ICILS-Studie 2013 hergestellten Zusammenhänge zwischen genderspezifischen informatischen Kompetenzen und Informatikselbstkonzept (s. Abschnitt 1.2) gewertet werden.

Zur Geschlechtszuordnung in der Informatik wurde die Frage „In Berufen mit Informatik arbeiten . . .“ mit den Ergänzungsoptionen „mehr Frauen“, „mehr Männer“ oder „etwa gleich viele Frauen wie Männer“ gestellt. Das Ergebnis zeigt, dass von vielen der befragten Schülerinnen und Schülern eine männlichen Rollenzuordnung bezüglich der Informatik vorgenommen wird. Fast die Hälfte der Schülerinnen und Schüler, 48,52 %, gibt an, dass mehr Männer in der Informatik arbeiten, während 5,60 % vermutet, dass der Frauenanteil größer wäre. 45,88 % geben an, dass die Anteile ausgeglichen seien.

Einfluss des Informatikunterrichts in der Grundschule

Weniger stark als die Geschlechtszugehörigkeit wirkt sich aus, ob die Schülerinnen und Schüler bereits in der Grundschule Informatikunterricht hatten. 10,5 % der 2 570 Befragten geben an, Informatikunterricht in der Grundschule gehabt zu haben. Der Informatikunterricht in der Grundschule beeinflusst das Interesse an Informatik (Abb. 3a) und auf das Informatikselbstkonzept (Abb. 3b) leicht positiv.

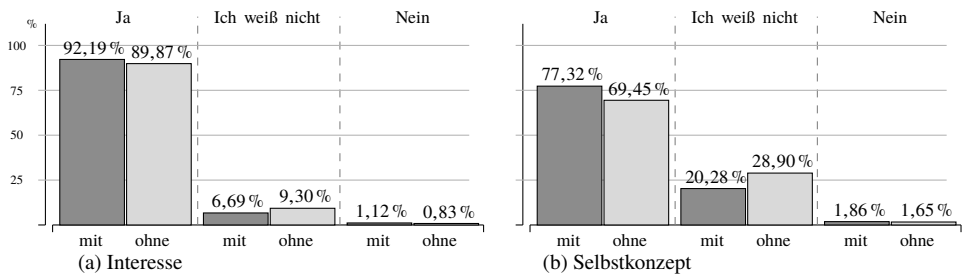


Abb. 3: Grundschulunterricht Informatik – Einfluss

Während sich 92,19 % derjenigen, die Informatikunterricht in der Grundschule hatten, auf den Informatikunterricht freuen, sind es bei den anderen 89,87 % (Abb. 3a). Eine leichte Steigerung bewirkt der Informatikunterricht in der Grundschule auch bei dem sehr geringen Anteil der Verneinung der Aussage „Ich freue mich auf den Informatikunterricht“. Von den befragten Schülerinnen und Schülern mit Informatikunterricht in der Grundschule geben 3 (1,12 %) an, sich nicht auf den Informatikunterricht zu freuen, während es bei den Befragten ohne Informatikunterricht in der Grundschule 19 der 2 301 Befragten (0,89 %) sind. Bei diesen geringen Zahlen ist diese Abweichung allerdings nicht signifikant.

Deutlicher als bei dem Interesse ist der Unterschied in Bezug auf das Informatikselbstkonzept: 77,32 % derjenigen mit Informatikunterricht in der Grundschule erwarten, gut in Informatik

zu sein, während es bei denjenigen ohne Informatikunterricht in der Grundschule 69,45 % sind (Abb. 3b). Die gemachten Erfahrungen mit Informatikunterricht verringern die Unsicherheit bezüglich der Erwartung an den zukünftigen Informatikunterricht: 9,30 % der Schülerinnen und Schüler ohne Informatikunterricht in der Grundschule antworten mit „Ich weiß nicht“ auf die Frage, ob sie sich auf den Informatikunterricht freuen. Hingegen sind es bei den Schülerinnen und Schülern mit Informatikunterricht in der Grundschule 6,69 %.

4 Fazit und Ausblick

Das Interesse an Informatik ist bei den Befragten überraschend hoch. Diese hohen Erwartungen sind bei den Jungen ausgeprägter als bei den Mädchen. Es zeigt sich bei allen Befragten ebenfalls ein im Durchschnitt hohes Informatikselbstkonzept, wobei die geschlechtsbezogene Diskrepanz deutlicher ausgeprägt ist als bei dem Interesse an Informatik. Die Mädchen haben ein niedrigeres Informatikselbstkonzept als die Jungen, wobei dieses bei beiden Geschlechtern eher hoch ist. Informatikunterricht in der Grundschule desillusioniert nicht, sondern stärkt das Interesse und das Selbstkonzept leicht. Die ausgewerteten Daten sprechen für eine frühe Einführung des Informatikunterrichts für alle Schülerinnen und Schüler (möglichst bereits in der Grundschule), um das Interesse der Kinder für Lernerfolge zu nutzen, zur frühen Förderung der Mädchen und zur Verminderung des Gendergaps in der Informatik.

Die forschende Beobachtung der Schülerinnen und Schüler in der Entwicklung ihrer motivationalen Tendenzen bezüglich der Informatik im Verlauf der zwei Jahre Informatikunterricht ist geplant. Entsprechende Daten einer Vergleichsgruppe der gleichen Jahrgangsstufe ohne Informatikunterricht werden ab dem kommenden Schuljahr erhoben und ausgewertet.

Literatur

- [Ei17] Eickelmann, B.: Konzepte und Entwicklungsperspektiven Kompetenzen in der digitalen Welt. 2017.
- [He11] Hellmich, F., Hrsg.: Selbstkonzepte im Grundschulalter: Modelle, empirische Ergebnisse, pädagogische Konsequenzen, Stuttgart: W. Kohlhammer, 2011.
- [KH06] Kessels, U.; Hannover, B.: Zum Einfluss des Image von mathematisch-naturwissenschaftlichen Schulfächern auf die schulische Interessenentwicklung. In (Prenzel, M.; Allolio-Näcke, L., Hrsg.): Untersuchungen zur Bildungsqualität von Schule. Abschlussbericht des DFG-Schwerpunktprogramms. Waxmann, Münster, S. 350–369, 2006.
- [Kl13] Kleinn, K.; Götsch, M.; Heine, Y.; Schinzel, B.: Das DFG-Projekt „Weltbilder der Informatik“. Informatik-Spektrum 36/3, S. 251–256, 2013.
- [Kn11] Knobelsdorf, M.: Biographische Lern- und Bildungsprozesse im Handlungskontext der Computernutzung, Dissertation, Department of Computer Science, Free University Berlin, 2011.

- [Kö06] Köller, O.; Trautwein, U.; Lüdtke, O.; Baumert, J.: Zum Zusammenspiel von schulischer Leistung, Selbstkonzept und Interesse in der gymnasialen Oberstufe. *Zeitschrift für pädagogische Psychologie* 20/1/2, S. 27–39, 2006.
- [Ku08] Kuhl, M.: *Studienkultur Informatik neu denken: Geschlechterkonstruktionen im Informatikstudium an der Universität Dortmund und der Carnegie Mellon University*. Shaker Verlag, Aachen, 2008.
- [Le08] Lechte, M.-A.: *Sinnbezüge, Interesse und Physik. Eine empirische Untersuchung zum Erleben von Physik aus Sicht von Schülerinnen und Schülern*, Dissertation, Opladen; Farmington Hills, Mich., 2008.
- [Lo14] Lorenz, R.; Gerick, J.; Schulz-Zander, R.; Eickelmann, B.: Computer- und informationsbezogene Kompetenzen von Mädchen und Jungen im internationalen Vergleich. In (Bos, W.; Eickelmann, B.; Gerick, J.; Goldhammer, F.; Schaumburg, H.; Schwippert, K.; Senkbeil, M.; Schulz-Zander, R.; Wendt, H., Hrsg.): *Computer- und informationsbezogene Kompetenzen von Schülerinnen und Schülern in der 8. Jahrgangsstufe im internationalen Vergleich. ICILS – International Computer and Information Literacy Study*, Waxmann Verlag, Münster, New York, Kap. VIII, S. 231–263, 2014.
- [MS18a] MSB-NW, Hrsg.: *Das Schulwesen in Nordrhein-Westfalen aus quantitativer Sicht – 2017/18*, Ministerium für Schule und Bildung des Landes Nordrhein-Westfalen (MSB-NW), Düsseldorf, 19. Juni 2018.
- [MS18b] MSB-NW, Hrsg.: *Statistik-TELEGRAMM 2017/18, Schuleckdaten 2017/18. Zeitreihen 2008/09 bis 2017/18. Statistische Übersicht Nr. 397 – 1. Auflage*, MSB-NW – Ministerium für Schule und Bildung des Landes Nordrhein-Westfalen, 15. Feb. 2018.
- [MT09] Möller, J.; Trautwein, U.: Selbstkonzept. In (Wild, E.; Möller, J., Hrsg.): *Pädagogische Psychologie*. Springer-Lehrbuch, Springer Berlin Heidelberg, S. 179–203, 2009.
- [Mü17] Müller, D.: *Der Berufswahlprozess von Informatiklehrkräften*. Überarbeitete Fassung der Dissertation von 2016, Universitätsverlag Potsdam, Potsdam, 2017.
- [Pa13] Pant, H. A.; Stanat, P.; Schroeders, U.; Roppelt, A.; Siegle, T.; Pöhlmann, C., Hrsg.: *IQB-Ländervergleich 2012 – Mathematische und naturwissenschaftliche Kompetenzen am Ende der Sekundarstufe I*, IQB – Institut zur Qualitätsentwicklung im Bildungswesen, Münster, New York: Waxmann Verlag, 2013.
- [Sc09] Schiefele, U.: Motivation. In (Wild, E.; Möller, J., Hrsg.): *Pädagogische Psychologie*. Springer-Lehrbuch, Springer Berlin Heidelberg, S. 151–178, 2009.
- [St17] Stanat, P.; Schipolowski, S.; Rjosk, C.; Weirich, S.; Haag, N.: Zusammenfassung und Einordnung der Befunde. In (Stanat, P.; Schipolowski, S.; Haag, N.; Rjosk, C.; Weirich, S., Hrsg.): *IQB-Bildungstrend 2016*. Waxmann Verlag, S. 387–411, 2017.

Informatikunterricht – Ein Muss zur politischen Mündigkeit

Niko Hausner¹, Katharina Wendlandt², Matthias Wendlandt³

Abstract: Ein erklärtes Ziel schulischer Bildung ist es, die SuS zu (politischer) Mündigkeit zu erziehen. Mündigkeit beschreibt dabei die Fähigkeit, selbstbestimmt und eigenverantwortlich zu handeln. Politische Mündigkeit im Besonderen befähigt dazu, sich begründet am aktuellen öffentlich-politischen Diskurs zu beteiligen und bildet somit die Grundlage aktiven demokratischen Handelns. In dieser Arbeit analysieren wir die Wahlprogramme zur Bundestagswahl 2017 derjenigen Parteien, die im Parlament vertreten sind, im Hinblick auf Themen die Informatik betreffend. Wir erörtern, welche informatischen Grundkenntnisse für die einzelnen digitalen Schwerpunkte nötig sind und inwiefern das Schulfach Informatik dazu beiträgt, eine Beurteilungskompetenz im politischen Wahlprozess zu entwickeln.

1 Einleitung

Im Zuge einer Wahlkampfveranstaltung im Vorfeld zur Wahl des 19. Deutschen Bundestages stellte einer meiner Studenten, der sich aktiv politisch engagiert, einem führenden Landespolitiker die Frage, was man seiner Meinung nach gegen illegale Seiten im Internet machen könne. Die Antwort des Politikers ist amüsiert und alarmierend zugleich. Der Politiker schlug vor, die Einträge und Verweise auf die entsprechenden Seiten bei Google zu löschen - die Webseiten seien dann schließlich nicht mehr erreichbar.

Nachdem der Student diese Anekdote im Rahmen eines Seminars zum Thema „Informatik als Pflichtfach?!“ vorgetragen hatte, entbrannte eine wilde Diskussion unter den Teilnehmenden. Wie soll man politisch selbstbestimmt handeln in einer Gesellschaft, die so extrem von informatischen Inhalten durchzogen ist, wenn man so wenig informatisches Wissen voraussetzen kann und so viele Fehlvorstellungen vorherrschen? Dies gilt nicht nur für Politikerinnen und Politiker, die durch politische Entscheidungen unser aller Leben beeinflussen, sondern es geht auch um die Bürgerinnen und Bürger, die sich mit den Wahlprogrammen auseinandersetzen und sich eine eigene Meinung über die Zukunft bilden müssen. Dies ist die Basis der Demokratie.

Josef Kraus, Präsident des Deutschen Lehrerverbandes, äußerte sich 2013 zur Einführung eines Pflichtfachs Informatik mit den Worten „Die Schüler müssen ja auch nicht wissen, wie eine Schreibmaschine funktioniert. Hauptsache, sie können sie bedienen.“[SP13]

Ähnlich äußerte sich die Schulbehörde Hamburg nach der Abschaffung des Pflichtfachs Informatik im Jahr 2013. Informatik sei als Fach „nicht für alle in gleicher Intensität

¹ Freiherr-vom-Stein-Schule, Dr.-Berthold-Leinweber-Str. 1, 35075 Gladenbach, Deutschland

² Mathematisches Institut, Universität Gießen, Arndtstr. 2, 35392 Gießen, Deutschland

³ Institut für Informatik, Universität Gießen, Arndtstr. 2, 35392 Gießen, Deutschland

vonnöten“, so Behördensprecher Peter Albrecht. Weiter heißt es „Etwas salopp formuliert: Alle sollten in einer mobilen Gesellschaft ein Auto fahren können. Aber nicht alle müssen auch wissen, wie es im Detail funktioniert oder gebaut wird.“[WE13]

Beide Kommentare zeigen, wie wenig fundiert der Blick auf das Schulfach Informatik ist und wie undifferenziert die Thematik analysiert wird. Eine mögliche Ursache mag mangelndes Fachwissen sein. In einem Gastbeitrag von Urs Lautebach in der *Zeit* schreibt er quasi konternd [ZE18]: „Diese Hilflosigkeit führt in eine digitale Unmündigkeit, denn wer von Informatik keine Ahnung hat, kann zwar die Systeme benutzen, die andere entwerfen und bewirtschaften, aber er oder sie kann sie nicht hinterfragen und schon gar nicht an eigene Bedürfnisse anpassen. Zu technisch geprägten Themen kann er sich keine Meinung bilden und die Wahrnehmung bestimmter Rechte bleibt ihm zunehmend verwehrt.“ Er fordert deshalb ein Pflichtfach Informatik, denn „Nur so kann die Schule auch im 21. Jahrhundert ihrem Allgemeinbildungsauftrag gerecht werden.“[ZE18]

In der folgenden Arbeit möchten wir unterstützende Argumente für die Aussage von Urs Lautebach geben, indem wir die Rolle informatischer Bildung im Bereich der Politik im Kontext der Bundestagswahl 2017 analysieren. Die Kernfragen, mit denen sich diese Arbeit beschäftigt, sind: Welchen Stellenwert haben informatische Inhalte im politischen Diskurs? Welche informatischen Themen sind in den einzelnen Wahlprogrammen enthalten? Über welche informatischen Grundlagen sollten Wählerinnen und Wähler verfügen, um die Themenkomplexe verstehen und bei der Bundestagswahl selbstbestimmt über die Parteiprogramme urteilen zu können?

2 Bildung, Mündigkeit und Demokratie

„Aufklärung ist der Ausgang des Menschen aus seiner selbst verschuldeten Unmündigkeit. Unmündigkeit ist das Unvermögen, sich seines Verstandes ohne Leitung eines anderen zu bedienen. Selbstverschuldet ist diese Unmündigkeit, wenn die Ursache derselben nicht am Mangel des Verstandes, sondern der Entschließung und des Muthes liegt, sich seiner ohne Leitung eines andern zu bedienen. Sapere aude! Habe Muth dich deines eigenen Verstandes zu bedienen! ist also der Wahlspruch der Aufklärung.“ [KA84, S.6]

Fritz Reheis [RE14] beschreibt in seinem Einführungswerk zur politischen Bildung die mündigen Bürgerinnen und Bürger als selbstständig in ihrer Lebensführung. Er muss nach Reheis in der Lage sein, selbstständig zu erkennen bzw. zu beurteilen, wann Widerspruch notwendig ist. Dabei stellt er einen Bezug zum Aufklärungspostulat von Immanuel Kant her. Wolfgang Sander [SD08] behandelt den mündigen Bürger als ein mögliches Leitbild der politischen Bildung. Die Mündigkeit der Bürgerinnen und Bürger sei nötig, um das demokratische System eines Staates am Leben zu erhalten. Entsprechend müssen nach Sander auch Schülerinnen und Schüler auf ihre Zukunft innerhalb einer Demokratie

vorbereitet werden. Für Seidel ist Mündigkeit die Fähigkeit „seine eigenen Angelegenheiten selbst regeln zu können“[SE16, S.286].

Eine wichtige Instanz zur politischen Bildung stellt das Bundesamt für politische Bildung dar. Mündigkeit wird dort wie folgt definiert: „Mündigkeit hat noch eine weitergehende Bedeutung. Gemeint ist damit auch Selbstbestimmung und Urteilsfähigkeit. Man spricht oftmals von mündigen Bürgern und meint damit, dass die Bürger und Bürgerinnen nicht nur für sich selbst Verantwortung übernehmen, sondern auch für ihren Staat und ihre Gesellschaft. In einer Demokratie wie in Deutschland ist das besonders wichtig. Die Demokratie braucht mündige Bürger und Bürgerinnen, die sich interessieren und engagieren, die bereit sind, politisch im Staat mitzuwirken.“[BP18]

Alle Definitionen verstehen Mündigkeit als Synonym der Selbstbestimmung auf allen Ebenen des Lebens. Sowohl Wolfgang Sander als auch das Bundesamt für politische Bildung heben darüber hinaus noch das aktive Leben in einer Demokratie hervor, was in den Begriff der politischen Mündigkeit gefasst werden kann. Im Folgenden verstehen wir unter *politischer Mündigkeit* die Fähigkeit, in politischen Entscheidungsprozessen selbstbestimmt urteilen und handeln zu können.

Wolfgang Klafki formulierte folgende grundlegende Fragestellung als Leitziel zu bildenden Inhalten:

„Welche Objektivierungen der bisher erschlossenen Menschheitsgeschichte scheinen am besten geeignet, dem sich Bildenden Möglichkeiten und Aufgaben einer Existenz in Humanität, in Menschlichkeit aufzuschließen, also einer auf wechselseitig anerkannte, damit aber immer auch begrenzte Freiheit, auf Gerechtigkeit, kritische Toleranz, kulturelle Vielfalt, Abbau von Herrschaft und Entwicklung von Friedfertigkeit, mitmenschliche Begegnung, Erfahrung von Glück und Erfüllung hin orientierte, vernunftgeleitete Selbstbestimmung?“[KL86]

Bildung und Erziehung haben also nach Klafki die Aufgabe, dem unmündigen Menschen zur Mündigkeit zu verhelfen. Die Begriffe der Bildung und der Mündigkeit werden dabei verknüpft und bilden die zentralen Begriffe seiner Bildungstheorie.

Wir sehen den Begriff der digitalen Mündigkeit, wie ihn Urs Lautebach im Zitat der Einleitung verwendet, als notwendige Voraussetzung, um informatische Inhalte im politischen Kontext beurteilen zu können. Die digitale Mündigkeit ist eine spezielle Form der Mündigkeit [BE18]. Der souveräne Einsatz von Informations- und Kommunikationstechniken (IKT) wird in Boberach/Neuburger [BO14] als eine wesentliche Voraussetzung für den mündigen, digital-souveränen Bürger hervorgehoben. So zeichnet sich *digitale Souveränität* durch ausreichend inhaltliches Wissen und technische Kompetenzen aus. Hierzu zählen der Autor und die Autorin den Umgang mit Programmen wie Word und Excel sowie die Nutzung technischer Geräte wie Drucker oder Scanner.

Nach Müller, Stecher, Dietrich, Wolf, et al. [MU16, S.7] benötigt die digitale Gesellschaft digital selbstbestimmte Bürgerinnen und Bürger, „die eigenverantwortlich mit den Produkten, Geräten und Technologien der heutigen Zeit umgehen können (Kompetenz) und sich proaktiv mit den Vorteilen und möglichen Risiken der Nutzung auseinandersetzen (Offenheit).“

Versteht man digitale Souveränität als einen Prozess, der sich stetig wandelt und immer an neue Entwicklungen gebunden ist, so ist die zweite Definition, die sich an den allgemeinen Konzepten orientiert, zu bevorzugen. In Bezug auf informatische Bildung in Schulen ist dies besonders wichtig. Schule kann nie so umfassend bilden, dass alle informatischen Inhalte aller zukünftigen Wahlprogramme detailliert im Unterricht behandelt werden. Die allgemeinen Konzepte können aber zum einen dazu beitragen, dass die Themen leichter zugänglich sind und zum anderen, dass die Grundlagen gelegt werden, um sich aktuelle Themen schneller aneignen zu können.

3 Parteiprogramme

In diesem Abschnitt arbeiten wir zunächst den Stellenwert des Themenkomplexes Informatik und Digitalisierung in jedem Parteiprogramm heraus. Die Beurteilung erfolgt relativ zum Umfang des jeweiligen Parteiprogramms. Hierbei werden wir uns ausschließlich auf die Parteien beziehen, die im aktuellen Bundestag vertreten sind.

CDU/CSU Das Regierungsprogramm [PU17] für 2017-2021 der Union erstreckt sich über 76 Seiten im regulären DIN A4 Format. Abgesehen von Einleitung und Vorwort sowie Wahlaufruf ist das Programm in neun verschiedene und auf den ersten Blick gleichberechtigte Themenkomplexe unterteilt. Inhalte zum Bereich Informatik und Digitalisierung finden sich im Themenkomplex mit dem Titel „Chancen im digitalen Zeitalter“. Die Union räumt den Themen mit digitalem Schwerpunkt einen eigenen Bereich in ihrem Regierungsprogramm ein. Damit erlangen diese Themen eine ähnliche Bedeutung wie beispielsweise die Familienpolitik oder die Sicherheitspolitik, welche ebenfalls einen eigenen Themenkomplex in diesem Programm bilden.

SPD Die SPD unterteilt ihr Regierungsprogramm [PS17] mit dem Titel „Es ist Zeit für mehr Gerechtigkeit: Zukunft sichern, Europa stärken“ in zwölf Themenkomplexe, exklusive der Einleitung. Dabei umfasst das Programm 116 DIN A4 Seiten. Im Programm der Sozialdemokraten findet sich zunächst kein eigener Themenkomplex, der der Digitalisierung gewidmet ist. Dennoch finden sich in mehr als der Hälfte der Themenkomplexe Bezüge zu digitalen Themen. Es lässt sich herausstellen, dass auch in diesem Programm den digitalen Themen ein hohes Gewicht zukommt.

Die Linke Auf rund 140 Doppelseiten, die in etwa dem DIN A4 Format entsprechen, beschreibt die Partei Die Linke ihr Programm [PL17] für die Bundestagswahl 2017. Sie setzen dabei auf differenziertere Themenüberschriften. In neun von 18 Themenkomplexen werden informatische Schwerpunkte thematisiert.

FDP Das Wahlprogramm [PF17] der FDP umfasst 150 Seiten. Abzüglich der Einleitung und einem allgemeinen Schluss umfasst es lediglich sechs Themenkomplexe. Erst ein Blick in die Untergliederung dieser sechs Hauptbereiche zeigt digitale Schwerpunkte. Es wird deutlich, dass alle Hauptbereiche, ausgenommen der Steuerpolitik („Politik, die rechnen kann“), informatische Themen beinhalten. Trotz dieser starken Verflechtung digitaler Themen mit dem Gesamtbild des Parteiprogramms ist die absolute Anzahl digitaler Themen geringer als beispielsweise bei SPD und Union, wenngleich die Deckung in diesem Vergleich erstaunlich ist.

Bündnis 90/Die Grünen Das umfangreichste Wahlprogramm [PG17] mit fast 250 Seiten im DIN A4 Format (wenn auch etwas größerem Schriftbild) entstammt der Partei Bündnis 90/ Die Grünen. Trotz des umfangreichen Programms beschränkt sich diese Partei auf wenige und entsprechend weit gefasste Themenkategorien. Man findet im Wahlprogramm von Bündnis 90/Die Grünen lediglich im zweiten Teil einige digitale Schwerpunkte. Die informatischen Themen sind in diesem Programm nicht so stark vertreten wie in anderen, jedoch sind auch hier die bedeutendsten von ihnen, gemessen an ihrem Anteil im öffentlichen Diskurs, durchaus zu finden.

AFD Die geringste Menge an informatischen Inhalten bietet das Wahlprogramm [PA17] der Alternative für Deutschland. Dies kann jedoch auch dem kompakten Umfang des Programms geschuldet sein. So umfasst es zwar 76 Seiten, doch deren Format entspricht in etwa nur DIN A5 im Querformat. Die Aufteilung der AFD sieht 15 Themenkomplexe vor, wobei sieben dieser 15 Komplexe informatische Themen beinhalten. Bezogen auf das kompakte Wahlprogramm findet durchaus eine bemerkenswerte Menge an informatischen Themen Platz, wenngleich es im direkten Vergleich zu anderen Parteien vom Umfang eher weniger ist.

4 Themenübersicht

Wir geben nun eine Auswahl der in den Wahlprogrammen der unterschiedlichen Parteien genannten informatischen Themen an. Dabei beschränken wir uns auf diejenigen Themen, die in mehreren Parteiprogrammen Erwähnung fanden, weshalb kein Anspruch auf Vollständigkeit der Liste aller digitalen Themen im Bundeswahlkampf 2017 bestehen kann. Dies sind die ausgewählten Themen in alphabetischer Reihenfolge:

Algorithmen, Big Data, Cybersicherheit, Datenschutz, Digitalisierung der Arbeitswelt, Fake News, Freies WLAN, Identifizierung und Verschlüsselung, Infrastruktur, Internetkriminalität, Künstliche Intelligenz, Netzneutralität, Software, Tracking Soft- und Hardware, Urheberrecht sowie Wahlfreiheit in den digitalen Märkten.

Im Folgenden werden wir nun einige der bedeutendsten Themen näher analysieren. Wir geben eine kurze Übersicht, inwieweit Berührungspunkte zu den Programmen der Bundestagsparteien bestehen, und begründen, weshalb informatische Grundkenntnisse dazu benötigt werden, um den kompletten Rahmen der politischen Diskussion zu durchdringen.

Kritiker mögen nun anmerken, dass Bürgerinnen und Bürger nicht in jedem Wahlkampfthema Experten sein können bzw. müssen. Dieser Ansicht kann man entgegenhalten, dass es zum einen einer Grundbildung bedarf, um Sachverhalte einschätzen zu können, und zum anderen eine grundlegende Bildung vorhanden sein sollte, damit sich Interessierte ohne Weiteres zusätzliches Wissen zu den genannten Themengebieten aneignen können.

Algorithmen „Ein sogenannter Algorithmen-TÜV soll dafür sorgen, dass niemand durch softwaregestützte Entscheidungen diskriminiert wird oder zu Schaden kommt.“ [PS17] So lautet die konkrete Forderung, die der SPD-Bundesverband in seinem Wahlprogramm für die Bundestagswahl 2017 formuliert hat.

In der *Zeit* [ZE17] wird Heiko Maas zitiert. Er verlangt „eine behördliche Kontrolle, um die Funktionsweisen, Grundlagen und Folgen von Algorithmen überprüfen zu können“. Diese Aufgabe könne eine Digitalagentur übernehmen, eine Aufsichtsbehörde, der Unternehmen wie Google Einblick in ihre Quellcodes und damit Geschäftsgeheimnisse geben müssten. Auch der Bundesverband der Partei Die Linke bezieht sich auf das theoretische Konzept des Algorithmus'. In ihrem Wahlprogramm wird eine transparente Offenlegung derjenigen Algorithmen, welche durch öffentlich-rechtliche Internetangebote genutzt werden, gefordert.

Abgesehen von der Tatsache, dass die Veröffentlichung eines Algorithmus' den Ruin des zugehörigen Unternehmens bedeuten könnte, mag dieses Konzept einen interessanten Ansatz darstellen. Informatisch gebildeten Bürgerinnen und Bürgern wird aber sofort klar, was digital unmündigen Bürgerinnen und Bürgern eventuell nicht bewusst ist: Dass Algorithmen beliebig komplex sind, dass sich Programme über hundertausende oder gar millionen Zeilen von Code erstrecken können und dass die Analyse von Programmcode von Menschenhand in dieser Komplexität praktisch unmöglich ist. Auch der Gedanke an eine automatisierte Analyse von Programmcode verschwindet sofort, da jegliche detaillierte Analyse sehr schnell auf ein nicht-entscheidbares Problem wie das Leerheitsproblem oder das Halteproblem für Turing Maschinen zurückgeführt werden kann.

Dieses Basiswissen wird zum einen beim Programmieren und zum anderen in der Theoretischen Informatik vermittelt. Beide Themengebiete sind in den meisten Curricula, zumindest in den Oberstufenplänen, enthalten.

Big Data Beide sogenannten Volksparteien (CDU/CSU und SPD) nutzen den Begriff Big Data oder eines seiner Synonyme innerhalb ihrer Regierungsprogramme. So schreibt die Union über „Umgang mit großen Datenmengen“, während die SPD den wirtschaftlichen Nutzen von Big Data mit den damit verbundenen Datenschutzfragen in Relation setzt. So heißt es im Wahlprogramm der SPD: „Ziel unserer Datenpolitik ist es, das Recht auf Privatsphäre zu gewährleisten. Gleichzeitig wollen wir das wirtschaftliche Potenzial von Daten nutzen, denn Datenschutz und Big Data schließen sich nicht aus. Wir werden klare Regelungen schaffen, wie Daten verfügbar gemacht werden können und wer welche Daten wann, zu welchem Zweck und zu welchen Bedingungen verwenden darf.“ Die FDP weist auf den Umstand des Datenwertes in ihrem Wahlprogramm hin. So sprechen sie den Verkauf des Messengerdienstes WhatsApp an den Facebook-Konzern an. Der Kaufpreis stehe in keinem Verhältnis zur Größe der Firma bzw. deren Jahresumsatz. Lediglich der Zugang, welchen WhatsApp zu Daten und Kommunikationskanälen habe, rechtfertige den Kaufpreis im zweistelligen Milliardenbereich.

Grundlegende algorithmische Verfahren, bzw. Verfahren aus dem Data Mining, verfeinern hierbei die Sichtweise auf die Grenzen und Möglichkeiten, große Datenmengen zu analysieren. Auch wenn der Themenkomplex Data Mining nicht verpflichtend in den aktuellen Curricula steht, hilft ein grundlegendes Verständnis für Algorithmisierung an dieser Stelle weiter. Die aktuelle Tendenz zeigt, dass die Analyse von großen Datenmengen einen immer größeren Stellenwert einnimmt. Es stellt sich die Frage, ob dieser Bereich aufgrund seiner affektiven und auch fachlich interessanten Seite nicht einen größeren Stellenwert in den Curricula einnehmen sollte. Neben den Möglichkeiten des Data Mining spielt der Wert hinter Daten eine Rolle für den politischen Diskurs um Big Data. Entsprechend wird die Fähigkeit, den Wert hinter Daten abschätzen zu können, für ein selbstbestimmtes Meinungsbild benötigt. Hierzu könnte Informatikunterricht beitragen.

Freies WLAN Die SPD verspricht, dass „öffentliche Einrichtungen offene und kostenfreie WLAN Hotspots verfügbar machen“ [PS17, S.38], die FDP erweitert diese Forderung um den öffentlichen Nahverkehr. Damit stoßen die Parteien auf großen Zuspruch in der Bevölkerung. Laut einer repräsentativen Umfrage der YouGov Deutschland AG [EC16] wünschen sich 74% der deutschen Bevölkerung über 18 Jahren flächendeckendes freies WLAN. Doch sind weiten Teilen der Bevölkerung die Gefahren öffentlich zugänglicher Netzwerke nicht bewusst. Auch wenn die Betreiber der Netze mit Sicherheit werben, so bleibt unerwähnt, dass die jeweiligen Anbieter den Datenverkehr und die Bewegungsprofile der Nutzer überwachen und ausspionieren können. Ist einem Bürger nicht bewusst, wer der Betreiber desjenigen Netzwerks ist, in das er sich eingewählt hat und welche Daten von ihm ausgelesen werden können, so besteht die Gefahr eines fahrlässigen Verhaltens in Verbindung mit dem Diebstahl sensibler Daten. Grundlegende Kenntnisse über Netzwerke, wie sie im Themenkomplex Internet unterrichtet werden, ermöglichen einen kritischen Blick auf dieses Wahlversprechen.

Identifizierung und Verschlüsselung SPD und FDP befassen sich nur oberflächlich mit der Thematik, fordern aber sichere Lösungen für elektronische Signaturen und verschlüsselte Kommunikation. Die Linke hingegen fordert ein „Recht auf Ende-zu-Ende-Verschlüsselung in persönlicher Kommunikation“. Eine elektronische Identifizierung findet Anwendung im Bereich E-Government. Sofern Anträge oder auch Verträge in einer Art Bürgerportal (so eine Forderung der Union) abgewickelt werden sollen, muss sichergestellt werden, dass immer der rechtmäßige Antragsteller oder Vertragspartner Zugriff auf personenbezogene Daten erhält. Eine Vorstellung davon zu haben, wie genau eine elektronische Signatur aussehen kann, welche Voraussetzungen sie erfüllen muss und was jeder einzelne Bürger und jede einzelne Bürgerin hierfür benötigt, sind Grundvoraussetzungen, um sich an der politischen Diskussion beteiligen zu können. Darüber hinaus stellt sich die Frage der Sicherheit, welche die meisten Bürgerinnen und Bürger beschäftigt. Die Wahlprogramme schreiben immer wieder, dass diese Verfahren sicher sind. Urteilen wir nicht fremdbestimmt, wenn wir uns blind darauf verlassen?

Ohne eine informatische Grundbildung sind weder das grundlegende Verfahren von E2EE (End-to-End-Encryption), noch dessen Vor- und Nachteile oder dessen Sicherheit bekannt. Ähnlich sieht es bei der elektronischen Signatur aus. Aktuell finden sich die Bereiche Verschlüsselung, sichere Kommunikation und elektronische Signaturen meistens nur fakultativ in den Curricula wieder. Jedoch werden diese Themengebiete meistens im Themengebiet Internet angeschnitten. Unter Berücksichtigung der Tatsache, dass diese Themenkomplexe einen immer höheren Stellenwert im alltäglichen Leben einnehmen, stellt sich die Frage, ob dieser Themenbereich nicht einen höheren Stellenwert in den Curricula einnehmen sollte.

Künstliche Intelligenz Die Parteiprogramme beschäftigen sich nur mit einem Ausblick auf die möglichen Anforderungen, die Entwicklungen im Bereich Künstliche Intelligenz für die Gesellschaft bereithalten könnten. Rechtliche und ethische Fragen stehen dabei für die SPD im Vordergrund. Bei der Union sind es vor allem Chancen in Forschung und Wirtschaft. Die FDP erachtet es als wichtig in ihrem Programm, dass Einblicke und Funktionsweisen in diesen Bereichen im Kindesalter vermittelt werden.

Im Bereich der Künstlichen Intelligenz gibt es viele Fehlvorstellungen, welche durch verschiedene Romane und Blockbuster noch verstärkt werden. Der Bereich der Künstlichen Intelligenz wird damit beworben, dass er ungeahnte Funktionalitäten und Möglichkeiten böte. Um Chancen und Gefahren richtig einordnen zu können, ist es sinnvoll, die Grenzen der Informatik, speziell die Grenzen der Künstlichen Intelligenz, zu thematisieren. Der Themenbereich ist fakultativ in vielen Curricula enthalten.

5 Diskussion

Die Analyse in Kapitel 3 hat gezeigt, dass informatische Inhalte in großem Umfang in den Parteiprogrammen zur Bundestagswahl 2017 enthalten sind, digitale Themen somit von großer Bedeutung für den öffentlich-politischen Diskurs sind. Gerade die sogenannten Volksparteien thematisieren sehr vielfältig Themenbereiche, in denen eine informatische Bildung notwendig ist, damit Bürgerinnen und Bürger selbstbestimmt urteilen können. Aber auch die Parteiprogramme der FDP und Die Linke beschäftigen sich an vielen Stellen mit Digitalisierung. Die FDP fordert unter anderem, dass „das Fach Informatik wesentlicher Bestandteil der berufsschulischen Ausbildung sein“ muss [PF17].

Die einzelnen Themengebiete sind sehr unterschiedlich und auch mit einer sehr stark unterschiedlichen Gewichtung in den Parteiprogrammen enthalten. Ein Hauptthemengebiet ist der Bereich Internet und verschlüsselte Kommunikation. In verschiedenen Programmen wurde die Offenlegung von Algorithmen, bzw. die Analyse von Funktionalitäten von Algorithmen gefordert. Grundlagenwissen, wie es in den aktuellen Curricula abgedeckt wird (auch wenn die genaue Interpretation der Themen von der jeweiligen Lehrkraft abhängt), ermöglicht hier eine kritische Auseinandersetzung mit den Inhalten und Forderungen. Es bleibt die Frage, ob diese Themen nicht doch flächendeckend stärker mit Bezug auf die aktuellen politischen oder gesellschaftlichen Gegebenheiten unterrichtet werden sollten.

Peter Euler schreibt in einem Artikel [EU13] über Bildung und Schule: „Stattdessen gibt es ständig neue Reformen, die alle ein Ziel haben: möglichst schnelle Verwertbarkeit für den Markt. Ist das Bildung? Ursprünglich bedeutete Bildung mehr: Mündigkeit für alle - die Basis für Demokratie.“ Schule erhebt den Anspruch, die SuS auf zukünftige Lebenssituationen vorzubereiten, sie möchte ihnen die Grundlagen zum selbstbestimmten Handeln und Urteilen in unserer Demokratie mitgeben. Das selbstbestimmte Urteilen über politische Entscheidungen und Ideen ist dabei der wesentliche Faktor. Die vorliegende Arbeit zeigt somit auf, dass eine informatische Grundbildung für alle notwendig ist.

Literatur

- [BE18] Roman Beck et al. *Eine Analyse der Fähigkeiten der Bürger in Deutschland zum konstruktiven und souveränen Umgang mit digitalen Räumen*. abgerufen (2019) unter: [negz.org/wp-content/uploads/2018/06/NEGZ-ISPRAT-Studie-Dig-M%C3%BCnd-Abschlussbericht.pdf](https://www.negz.org/wp-content/uploads/2018/06/NEGZ-ISPRAT-Studie-Dig-M%C3%BCnd-Abschlussbericht.pdf) [2018].
- [BO14] Michael Boberach, Rahild Neuburger. *Zukunftspfade Digitales Deutschland 2020*. Springer. 2014.
- [BP18] Gerd Schneider, Christiane Toyka-Seid. *Mündige Bürgerinnen. Das junge Politik-Lexikon*. In: Bundeszentrale für politische Bildung. 2017.
- [EC16] Verband der Internetwirtschaft e.V. *Das Haftungsrisiko fällt weg*. abgerufen (2019) unter: https://www.eco.de/wp-content/blogs.dir/freie-hotspots_infografik-2.jpg [2016]
- [EU13] Peter Euler. *Bildung heißt Mündigkeit*. In: Metallzeitung (Mitgliederzeitung der IG Metall) Heft 4. S. 15-16. 2013.

- [KA84] Immanuel Kant. *Beantwortung der Frage: Was ist Aufklärung?* In: Berlinische Monatschrift Band 12. S.481-494. 1784.
- [KL86] Wolfgang Klafki. *Die Bedeutung der klassischen Bildungstheorien für ein zeitgemäßes Konzept allgemeiner Bildung.* In: Zeitschrift für Pädagogik. 32. Aufl. S. 455-475. 1986.
- [MU16] Lena-Sophie Müller et al. *D21-DIGITAL-INDEX. Jährliches Lagebild zur Digitalen Gesellschaft.* abgerufen (2019) unter: initiated21.de/app/uploads/2017/01/studie-d21-digital-index-2016.pdf [2016].
- [PA17] Alternative für Deutschland. *Programm für Deutschland.* Wahlprogramm der Alternative für Deutschland für die Wahl zum Deutschen Bundestag am 24. September 2017. Köln. 2017.
- [PF17] Freie demokratische Partei. *Denken wir neu.* Das Programm zur Bundestagswahl 2017 der Freien Demokraten. Berlin. 2017.
- [PG17] Bündnis 90/ Die Grünen. *Zukunft wird aus Mut gemacht.* Bundestagswahlprogramm 2017. Berlin. 2017.
- [PL17] Die Linke. *Sozial. Gerecht. Frieden. Für alle. Die Zukunft für die wir kämpfen!* Die Linke. Langfassung des Wahlprogramms zur Bundestagswahl 2017. Hannover. 2017.
- [PS17] Sozialdemokratische Partei Deutschland. *Es ist Zeit für mehr Gerechtigkeit: Zukunft sichern, Europa stärken.* Das Regierungsprogramm 2017-2021. Berlin. 2017.
- [PU17] Die Union. *Für ein Deutschland, in dem wir gut und gerne leben.* Regierungsprogramm 2017-2021. Berlin. 2017.
- [RE14] Fritz Reheis. *Politische Bildung. Eine kritische Einführung.* Springer. 2014.
- [SD08] Wolfgang Sander. *Politik entdecken - Freiheit leben: Didaktische Grundlagen politischer Bildung.* Wochenschau Verlag. 2008.
- [SE16] Christian Seidel. *Selbst bestimmen. Eine pholosophische Untersuchung personaler Autonomie.* De Gruyter. 2016.
- [SP13] Hilmar Schmundt. *Wir machen eine Rolle rückwärts.* abgerufen (2019) unter: www.spiegel.de/lebenundlernen/schule/erziehungswissenschaftler-wollen-informatik-als-pflichtfach-einfuehren-a-903096.html [2013].
- [WE13] Jens Meyer-Wellmann. *Hamburg schafft trotz Kritik das Pflichtfach Informatik ab.* abgerufen (2019) unter: www.welt.de/print/welt_kompakt/hamburg/article116640160/Hamburg-trotz-Kritik-das-Pflichtfach-Informatik-ab.html [2013].
- [ZE17] Christopher Lauer. *Gesetzesbrecher im Netz.* abgerufen (2019) unter: www.zeit.de/2016/02/internet-charta-heiko-maas-grundgesetz [2016].
- [ZE18] Urs Lautebach. *Informatik für alle.* abgerufen (2019) unter: www.zeit.de/gesellschaft/schule/2018-02/digitalisierung-informatikunterricht-schulen-bildung [2018].

Informatische Bildung für alle Lehramtsstudierenden

Reformprozess einer allgemeinbildenden Informatikveranstaltung in der universitären Lehrerbildung

Daniel Losch,¹ Ludger Humbert²

Abstract: Alle Lehrpersonen benötigen – unabhängig von ihren konkreten Fächern – angesichts einer von Informatik durchdrungenen Welt einen Zugang zu allgemeinbildenden Elementen der Informatik. An der Bergischen Universität Wuppertal werden seit 2009 regelmäßig Veranstaltungen durchgeführt, die dieser Anforderung Rechnung tragen. Befragungen und Rückmeldungen der Studierenden sowie der Lehrenden aus der mehrjährigen Durchführung der Veranstaltung legten nahe, dass eine für alle Lehramtsstudierenden adäquate Optimierung der Veranstaltung möglich ist. Daraus ergibt sich als eine zu untersuchende Hypothese: Das bestehende Lehrangebot muss – hinsichtlich der intendierten Kompetenzen für die Studierenden – als nicht vollständig befriedigend bezeichnet werden. Durch die Rekonfiguration der Veranstaltung werden Möglichkeiten zur Qualitätsverbesserung des Lehrangebots diskutiert. Die Umsetzung wird prozessbegleitend untersucht, um gegebenenfalls zeitnah zu intervenieren. Durch qualitative Forschungselemente konnten studentische Belange inhaltlicher und organisatorischer Art auf die Hypothese bezogen werden und diese in erster Näherung bestätigen. Die Hinzunahme zweier informatikfachwissenschaftlicher und -fachdidaktischer Perspektiven gewährleistet die Triangulation des konzeptuellen Reformprozesses. Das entwickelte Konzept einer allgemeinbildenden Informatikveranstaltung für alle Lehramtsstudierenden berücksichtigt eine verstärkte Verzahnung von Theorie und Praxis, inhaltliche Modularität und konkreten Berufs- bzw. Lebensweltbezug.

Keywords: Allgemeinbildung; Informatik; Lehrerbildung; Literalität; Reform; Phänomene

1 Einleitung

1.1 Motivation und Forschungskontext

Dass informatische Bildung in der Gesellschaft eine nachhaltige Verankerung finden muss, ist Konsens in der Informatikdidaktik [bspw. Br16]; dass es dazu einen Lernort durch das verpflichtende Schulfach Informatik in der Schule braucht, findet ebenfalls breite Zustimmung. Hierzu bedarf es zahlreicher, grundständig in erster und zweiter Phase der Lehrerbildung ausgebildeter Informatiklehrkräfte sowie darüber hinausgehender Informatikfortbildungen für Lehrkräfte. Um aber Aspekten der Medienbildung [etwa Ha99] angemessen Rechnung

¹ Bergische Universität Wuppertal, Gaußstraße 20 42119 Wuppertal, d.losch@uni-wuppertal.de

² Bergische Universität Wuppertal, Gaußstraße 20 42119 Wuppertal, humbert@uni-wuppertal.de

zu tragen, müssen allgemeinbildende Elemente der Informatik für Lehrkräfte aller Fächer aufbereitet werden. So wie Schülerinnen und Schüler erst durch die Kombination von Informatik und integrativer Medienbildung zur Mündigkeit befähigt werden, so bedürfen auch Lehramtsstudierende in der ersten Phase ihrer Ausbildung der Synergien von eigenständigem Informatikangebot und fachlich konkretisierender Mediendidaktik. An der Bergischen Universität Wuppertal werden seit 2009 regelmäßig Veranstaltungen durchgeführt, die dies zu gewährleisten versuchen. Sie wurden bis 2016 hinsichtlich ihrer inhaltlichen Tiefe sukzessive weiter ausgerollt. Zu den seither etablierten Grundfesten der Veranstaltung gehört die Repräsentation der fachlichen Breite der Informatik durch Dozierende, die jeweils aus ihrem jeweiligen Fachgebiet Aufschluss zur Informatik bieten. Für die Orientierung informatischer Lehrangebote für Studierende anderer Fachkontexte klassifizieren Seegerer und Romeike vier Grundtypen: Denkweisen, Fluency, Wissenschaft und Gesellschaft [SR18, S. 20]. Die Veranstaltung kann vornehmlich der Kategorie »Wissenschaft« zugeordnet werden:

Kurse sollen einen Überblick über das Fachgebiet Informatik geben. Den Studierenden werden zentrale Ideen und Schlüsselkonzepte der Wissenschaft Informatik aufgezeigt. Es geht häufig auch darum, ein (breites) Bild der Disziplin zu vermitteln und gleichzeitig das Verständnis für dessen grundlegende Konzepte zu schaffen. [SR18, S. 20]

Zwölf Lehrende gestalten die Veranstaltung durch Beiträge; mit dieser Vielzahl wird auch versucht, die Vielheit in der Wissenschaft Informatik abzubilden. Die Veranstaltung wendet sich an zwei Gruppen bezüglich der zu erwerbenden ECTS: zwei respektive sechs ECTS. Durch curriculare Restriktionen der betroffenen Studiengänge ist für die Lehramtsstudierenden der Sekundarstufe I der geringere Umfang von 60 Zeitstunden vorgesehen; alle weiteren Lehramtsstudierenden sowie Nicht-Lehramtsstudierende studieren 180 Zeitstunden. Die Studierenden wenden zur Erfüllung der Anforderungen vier zusätzliche ECTS im Zusammenhang mit einem begleitenden Projektseminar auf. Wir betrachten in diesem Zusammenhang die Veranstaltung, die zwei ECTS umfasst; diese Veranstaltung verfolgt das Ziel, bei den Studierenden einen universitären »Mindeststandard« informatischer Kompetenzen zu entwickeln.

Befragungen und Rückmeldungen der Studierenden sowie der Lehrenden aus der mehrjährigen Durchführung der Veranstaltung legten bereits nahe, dass eine für alle Lehramtsstudierenden adäquate Optimierung der Veranstaltung möglich ist. Insbesondere die Interessenlage der Studierenden, die Freiwilligkeit der Veranstaltung und eine vermeintliche und »gefürchtete« Abstraktheit der informatischen Fachgegenstände bedürfen einer informatikdidaktischen Erörterung.

1.2 Fragestellung

In der bundesdeutschen Hochschullandschaft gibt es ein ausgewiesenermaßen umfangreiches Angebot an Medienbildung, was anhand der Zahl hochschulischer Medienzentren bereits deutlich wird. Universitäre Angebote, die eine Entwicklung informatischer Selbstkompetenz im Zusammenhang mit geeigneten Veranstaltungen ermöglichen, sind im Vergleich kaum auffindbar; Seegerer und Romeike haben neun Hochschulkurse für Studierende anderer Fachkontexte in Deutschland in ihre Datenbasis aufgenommen [SR18, S. 18]. Die Frage nach der konkreten Gestaltung eines universitären Informatikangebots für alle Lehramtsstudierenden wurde nach unserer Kenntnis bisher nicht wissenschaftlich untersucht. Im Folgenden stellen wir ein informatikdidaktisches Forschungselement aus dem Reformprozess eines bestehenden Konzeptes vor. Dabei wird folgende Hypothese formuliert: Das bestehende Lehrangebot muss – hinsichtlich der intendierten Kompetenzen für die Studierenden – als nicht vollständig befriedigend bezeichnet werden. Wenn das Lehrangebot sich als nicht vollumfänglich zielführend erweist, ist insbesondere die Beantwortung folgender Fragen bedeutsam: Was muss inhaltlich angeboten werden? Welche Formen der Umsetzung sind zielführend?

2 Vorgehensweise

Um eine multiperspektivische Auseinandersetzung des konzeptuellen Reformprozesses zu gewährleisten, wird ein triangulatives Forschungsdesign gewählt:

1. Die *studentische* Perspektive auf die Veranstaltung wird mittels einer qualitativen, schriftlichen Befragung erhoben; die Befragung wird dabei als Reflexionselement nach etwa zwei Dritteln der Veranstaltung implementiert. Im Anschluss fand eine Diskussion zur Gesamtveranstaltung in dialogischer Hörsaalsituation statt; dazu wurden zwei Leitfragen formuliert – zum einen nach dem Verhältnis von inhaltlicher Tiefe und thematischer Breite zum anderen nach Aufwand, Ablauf sowie Organisation der Veranstaltung. Abschließend konnten die Forschungselemente hinsichtlich aktueller Umgestaltungen und Gesamtkonzept unter Zuhilfenahme qualitativer Inhaltsanalyse [Ma99] ausgewertet werden; konkret wurden Zusammenfassungen der einzelnen Erhebungen vorgenommen und induktive Kategorien gebildet, welche schließlich einer Häufigkeitsanalyse unterzogen wurden [Ma99, S. 67 ff.].
2. Der empirische Befund (s. Abschnitt 4) wird aus der Perspektive zweier *fachwissenschaftlich* orientierter Aufschlüsse der Informatik betrachtet ([CI91; De08] – s. Abschnitt 5.1), die strukturgebende Impulse für die Veranstaltungskonzeption setzen.
3. Die Ergebnisse werden über zwei *fachdidaktische* Aufschlüsse diskutiert (s. Abschnitt 5.2): Das Konzept informatischer Literalität bietet die Phänomenorientierung,

bei der einerseits Phänomene als Erscheinungen und Konsequenzen des alltäglichen Lebens aufgefasst werden [HP04]; andererseits werden informatische Handlungsdimensionen in Anwendung, Gestaltung und Entscheidung deutlich. In diesem Zusammenhang soll die von Engbring erweiterte Form des sogenannten »Dagstuhl Dreiecks« einen Aufschluss aus der gemeinsamen Perspektive eines Diskurses von Medienpädagogik, Informatik und Medienwissenschaft zu. Um den organisatorischen Rahmen der Veranstaltung angemessen zu erfassen, sollen zur Unterstützung zugeschnittene Elemente einer empirischen Studie herangezogen werden; dieser Studie liegt eine Ein-/Ausgangsbefragung in der Veranstaltung mit quantitativem Schwerpunkt zugrunde.

3 Durchführung

Vorlesungsthema	Kürzel	SWS	Dozent*in
Was ist Informatik?	was	4	A
Geschichte der Informatik	hist	4	A
Zahlensysteme und Codierung	code	4	B
Eingebettete Echtzeitsysteme	ees	4	C
Rechnerarchitektur	arch	2	D
Modellierung und Implementierung	imp	6	E
Mensch-Maschine-Schnittstellen	mms	4	F
Technische Informatik	tech	2	D
Betriebssysteme	os	2	G
Grundlagen Internet	net	4	H
Kryptographie	crypt	2	I
Sicherheit im Internet	sec	2	H
Recht und Internet	law	2	J
Hörsaalreflexion	refl	2	–
Datenbanken	db	4	B
Maschinelles Lernen	ml	2	L
Grenzen der Informatik	theo	4	K
Informatikallgemeinbildung	did	4	A

Tab. 1: Gegenstände der Vorlesung

Die Umgestaltung der Veranstaltung im Semester der Durchführung umfasste folgende Aspekte:

- Restrukturierung nach fachdidaktischer Maßgabe der thematischen Reihung
- Ergänzung um eine Vorlesung zum maschinellen Lernen
- Erweiterung der Teilveranstaltung {imp}³ um Programmierpraxis

³ Die in { . . . } gesetzten Veranstaltungskürzel sind stets in der Tabelle 1 aufgelöst.

- Erweiterung der Teilveranstaltungen {was, code, tech, net, crypt, sec, theo} um enaktive Elemente⁴

Im Durchschnitt waren 62 Studierende in der Veranstaltung anwesend. An der Evaluations-sitzung, die auch die qualitative Befragung einschließt, nahmen 54 Studierende teil, wobei 49 Bögen ausgefüllt wurden. Die von den teilnehmenden Studierenden belegten Fächer sind in Tabelle 2 dargestellt. Die Teilveranstaltung {refl} wurde als reflexives Element eingebettet, um die genannte Hypothese einer Überprüfung zuführen zu können:

1. Die Studierenden konnten – über die Implementierung der »Think-Pair-Share«-Methode – ihre Reflexion zur Begegnung mit dem *Fach* Informatik und mit der *Veranstaltung* als solcher schriftlich dokumentieren, anhand von Leitfragen mit Kommiliton*innen diskutieren und schließlich vor dem Plenum zu Gehör und zu Protokoll geben.
2. Den Studierenden wurden Hintergründe zur Veranstaltung dargestellt und erhielten einen Ausblick auf Reformüberlegungen der Veranstaltenden.
3. Die Studierenden waren aufgerufen, angesichts des neu gewonnenen Hintergrundwis-sens zur Veranstaltung weitere Argumente vorzubringen.

Fach	Teilnehmende	Fach	Teilnehmende
Mathematik	53,45 %	Geschichte	6,90 %
Englisch	18,97 %	Physik	6,90 %
Biologie	17,24 %	Psychologie	6,90 %
Musik	17,24 %	Evangelische Religion	5,17 %
Politik	17,24 %	Französisch	3,45 %
Chemie	12,07 %	Spanisch	3,45 %
Deutsch	12,07 %	Katholische Religion	1,72 %
Pädagogik	6,90 %		

Tab. 2: Teilnehmende der Veranstaltung nach studierten Fächern⁵

4 Ergebnisse und Diskussion der empirischen Forschungselemente

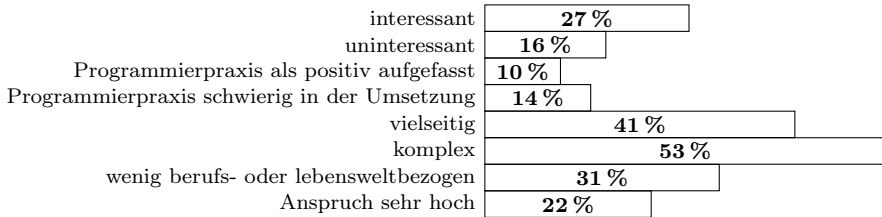
4.1 Struktur und erste Ergebnisse der Befragung der Studierenden

Die *Interessenlage* der Studierenden an der Informatik kann als ausgeglichen mit leicht positivem Überhang bezeichnet werden; Informatik ist also für einige (27 %, s. Abb. 1)

⁴ Bei diesen Elementen handelt es sich um Rollenspiele oder Gruppenarbeiten im Hörsaal mit haptischer Aktivität.

⁵ Lehramtsstudierende qualifizieren sich in zwei respektive drei Fächern, daher sind die Summen größer als 100 %.

⁶ Das qualitative Format sah eine *offene* Beantwortung der Fragen vor. Daher ist die Summe größer als 100 %

Abb. 1: Häufigkeiten der induktiven Kategorien⁶

ein Thema, in das ein Einblick lohnenswert scheint; 16 % bekundeten explizites Desinteresse, sodass etwas über die Hälfte von sich aus keine Stellung dazu bezieht. Zehn Studierende hoben *explizit* die Teilveranstaltungen {was, hist, law, crypt, net, sec} als sehr positiv hervor; dies kann bei ersteren dreien auf ein hohes Interesse an der Reflexion gesellschaftlich-kultureller Wechselwirkungen [En18, S. 7] zurückgeführt werden, was auch an anderer Stelle bereits belegt werden konnte [SR18, S. 26]. Für {crypt, net, sec} sehen die Studierenden die positive Markierung in der Integration enaktiver Phasen begründet: »Beispiel Programmieren erst Theorie und anschließend Praxis (aber dann auch mehr) Mehr Interaktion mit der Hörerschaft wie bei den Entschlüsselungen und dem »Paket-Verschicken« hält (mich persönlich) näher am Geschehen und macht Konzepte greifbarer«. Hier wird sogar eine Steigerung der Anteile an Programmierpraxis gefordert. 41 % der Befragten bekundeten unmittelbar die *Vielseitigkeit* der Informatik in dieser Veranstaltung (davon hoben acht Personen die Vielfalt explizit als positiv hervor); dies bestätigt die geplante Darstellung der fachlichen Breite; die Ringvorlesung ist dafür vermutlich weiterhin geeignetes Gestaltungsformat, wenngleich sich eine Person explizit gegen das Format aussprach. Eine Beobachtung einer befragten Person lautet: »Zur Einordnung der entsprechenden Themen sollten meiner Meinung nach mehr verbindende Elemente betont werden, sodass man den zu Beginn der Veranstaltung vorgestellten ›roten Faden‹ häufiger wiedererkennt.« Dies zeigt, dass die Absicht, die Veranstaltung unter ein Leitmotiv zu stellen, erkennbar war, jedoch in der Umsetzung eine eher geringe Kohärenz⁷ der Teilveranstaltungen aufwies. Die häufigste Zuschreibung an die Informatik ist ihre *Komplexität*⁸; damit kann vermutet werden, dass Studierende sich informatische Kompetenzen also nicht »nebenbei« aneignen können; demnach würden integrative Informatikangebote allein nicht ausreichen. Dies ist auch ein wichtiges Element für den Reformprozess: Primär sollten in dieser fachwissenschaftlich orientierten Informatikveranstaltung für Nicht-Informatikstudierende Prozess- und Gegenstandsbereiche der Informatik thematisiert werden. Der Bezug zu Lehramt, Studienfächern und Lebenswelt der Teilnehmenden sollte möglichst berücksichtigt werden sein. Die bisher durchgeführte Veranstaltung und ihre Struktur führen offensichtlich bei 31 % der Befragten der Veranstaltung dazu, dass derartige Bezüge nur eingeschränkt hergestellt werden können; dazu gesellt sich, dass 22 % der Befragten den Anspruch als sehr hoch einstufen.

⁷ In diesem Beitrag wird Kohärenz als Bezeichnung für die einheitliche Orientierung an einer fachdidaktischen Gestaltungslinie verstanden.

⁸ Die Erwartungen bezüglich der Komplexität und Vielseitigkeit wurden oftmals als überraschend aufgefasst.

4.2 Interaktives Reflexionselement

Die dialogische Hörsaalsituation bestätigt die Ergebnisse der schriftlichen Befragung: So habe es mit Blick auf inhaltliche Tiefe und thematische Breite eine gelungene Übersicht gegeben, die jedoch nicht überschneidungsfrei geblieben sei. Die Kohärenz der Teilveranstaltungen sollte in diesem Sinne stärker herausgestellt werden. Wenngleich praktische Elemente in die Teilveranstaltungen {was, code, imp, tech, net, crypt, theo} eingewoben wurden, sei der Vorlesungsanteil zu umfangreich gewesen; statt des Programmierens seien andere bzw. weitere Praxisbeispiele informatischen Handelns zu gestalten (ein genanntes, studentisches Anliegen: rudimentäre Netzwerkkonfiguration). Die stärkere, *praktisch* orientierte Einbindung der Studierenden in die jeweiligen informatischen Themen ist eine bedeutsame Auflage für den Reformprozess. Insgesamt wurde durch die Diskussionsbeiträge das Spannungsfeld zwischen den Dimensionen Breite, Tiefe und Anwendbarkeit der informatischen Themen deutlich.

5 Triangulation

5.1 Fachwissenschaftliche Perspektive

Im Rahmen wissenschaftstheoretischer Überlegungen zur Informatik wirft Denning folgende, grundlegende Fragen auf: »What is computation? What is information? What can we know through computing? What can we not know through computing? How can we build complex systems simply?« [De08, S. 2]. Für eine geeignete Darstellung der Wissenschaft Informatik in didaktischen Zusammenhängen empfiehlt Claus folgende Zielbereiche: (a) wissenschaftsbezogener Bereich, (b) Zukunftsaspekte, (c) Einsatz–Anwendungen–Auswirkungen bis heute, (d) Nutzen (und Schaden), (e) Grundtechniken und Unterstützungscharakter [CI91, S. 149]. Mit den Teilveranstaltungen {was, hist} wird dem wissenschaftsbezogenen Bereich bereits begegnet; dieses Element ist auch laut empirischem Befund fundamental für den Aufschluss der Informatik und wurde explizit positiv konnotiert. Die Teilveranstaltungen {ees, os, net, crypt, sec, dbs, ml} sind den Zielbereichen »Zukunftsaspekte«, »Einsatz–Anwendungen–Auswirkungen bis heute«, »Nutzen (und Schaden)« dienlich. Der Beantwortung der von Denning aufgestellten Fragen kann insbesondere über die *kerninformatischen* Fachgebiete abgedeckt werden, die in »reiner Form« bislang zu wenig Berücksichtigung in der Veranstaltung gefunden haben. Der bisherige Ansatz ist stark an den Fachgegenständen einzelner (den Studierenden nicht transparenten) Disziplinen orientiert gewesen; damit ist auch der zu Tage getretene Mangel an Kohärenz erklärbar.

5.2 Fachdidaktische Perspektive

Wird nun – als zweiter Zugang – die fachdidaktische Perspektive der Phänomenorientierung hinzugenommen, so ist zu konstatieren, dass Beispiele aus allen drei »Phänomenbereichen«

[HP04, S. 4] in die Veranstaltung eingebettet werden sollten und exemplarisch in je einer Teilveranstaltung informatisch erörtert werden. Dadurch könnte sichergestellt werden, dass die erhobene informatische Komplexität und der zu hohe Anspruch lebensweltbezogen aufgelöst wird. Die Teilveranstaltungen {ees, os, net, crypt, sec, dbs, ml} eignen sich dazu in besonderer Weise. Das in [Pu03, S. 138] vorgestellte Konzept informatischer Literalität ermöglicht folgende, erhellende Aufstellung:

Handlungs- dimension	Veranstaltung (s. Tab. 1)	fachwissenschaftliche und fachdidaktische Referenz
Anwendung	{ees, os, net, crypt, sec, dbs, ml}	(c),(e) (s. Abschnitt 5.1)
Gestaltung	{tech, mms, imp, code}	[De08]; Strukturen und Funktionen [En18]
Entscheidung	{hist, theo, law, did}	(b),(d) (s. Abschnitt 5.1); gesellschaftlich-kulturelle Wechselwirkungen [En18]

Tab. 3: Zuordnung von Elementen informatischer Literatitat zu Teilen der Veranstaltung

Eine Betonung der Handlungsdimension »Anwenden« begegnet den nachgewiesenen Problemzonen beim Berufs- und Lebensweltbezug und adressiert Claus' Zielbereiche »Einsatz–Anwendungen–Auswirkungen bis heute« (c) und »Grundtechniken und Unterstutzungscharakter« (s. Abschnitt 5.1). Die Handlungsdimension »Gestalten« hingegen sollte in einem kerninformatischen Modul einen Schwerpunkt bilden. Damit kann dem Problem hoher Komplexitat Rechnung getragen werden. Der Programmierpraxis und der technischen Informatik wird in praktischer Ausfuhrung mehr Zeit eingeraumt. Das groe Interesse an der Reflexion gesellschaftlich-kultureller Wechselwirkungen spiegelt sich in der Handlungsdimension »Entscheiden« wieder. Die Teilveranstaltung {was} muss beim Uberblick uber die Informatik einen »roten Faden« aufzeigen, der die einzelnen Veranstaltungselemente koharent in den Modellierungsprozess einbindet. Somit wird einerseits die jeweilige Schwerpunktsetzung (Anwendung–Gestaltung–Entscheidung) einer Teilveranstaltung fur die Studierenden transparent; andererseits wird der prozessuale Zusammenhang der Handlungsdimensionen verdeutlicht, der in den Teilveranstaltungen stets mitgedacht werden sollte.

6 Konsequenzen

Die Veranstaltung ist bisher durch eine starke informatikfachwissenschaftliche Orientierung gepragt. Damit erfahren die Studierenden an einigen Stellen die groe fachliche Komplexitat und nur geringe Lebensweltbezug. Die ersten Ergebnisse scheinen damit vorerst die oben dargestellte Vermutung zu bestatigen. Um diesem Eindruck zu begegnen, versuchen wir, die Veranstaltung durch den vorgestellten Losungsansatz (s. Abb. 2) so anzureichern und umzugestalten, dass die Studierenden informatische Kompetenzen in allen in [SR18, S. 20] aufgestellten Kategorien entwickeln konnen. Dabei sollen Theorie und Praxis starker verzahnt werden, indem in jeder Woche einer Vorlesung eine praktische Horsaalubung zugeordnet wird. Die Veranstaltung als Ganzes wird modularisiert (s. Abb. 2):

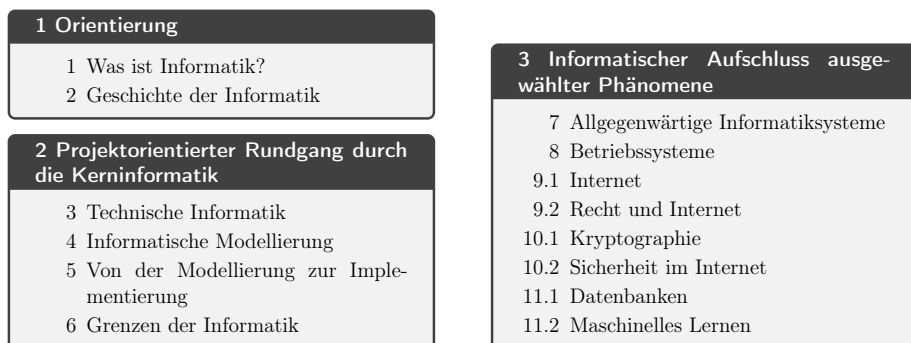


Abb. 2: Vorläufiges Konzept zur Planung einer allgemeinbildenden Informatikveranstaltung

Auf die eingängliche Orientierungsphase, folgt ein projektorientierter Rundgang durch die Kerninformatik. Diese Exposition möge als Grundlage für die Gestaltungs- und Entscheidungskompetenz der Studierenden am Beispiel des Mikrocontrollers Calliope [Ge18] gelten. Der informatische Aufschluss ausgewählter Phänomene bietet die Möglichkeit – auf Basis der kerninformatischen Kompetenzen –, Aspekte der Anwendung mit konkretem Berufs- bzw. Lebensweltbezug zu erarbeiten. Nächste Schritte sind die interne Revision der Veranstaltung auf Grundlage der hier ausgeführten Forschungselemente im kollegialen Kreise und die Untersuchung des projektorientierten Begleitformates (für Studierende mit vier zusätzlichen Leistungspunkten). Dieses birgt ein Potential zur Schaffung von noch stärkerem, studentischen Berufs- und Lebensweltbezug zu informatischen Denk- und Handlungsweisen – auch und gerade bezüglich individueller Studienfachrichtungen. Letztlich ergeben sich Herausforderungen im Hinblick auf die Zieldimension des gesamten Reformprozesses: Es wird angestrebt, durch Vereinheitlichung und Dokumentation die Portabilität des Studienmoduls auf andere Hochschulstandorte voranzutreiben. Wird eine allgemeinbildende Informatikveranstaltung angeboten, so ist nach den bisherigen Überlegungen in der Fachdidaktik eines jeden Faches eine fruchtbare Diskussion über fachlich-informatische Wechselwirkungen anzuregen.

Literatur

- [Br16] Brinda, T.: Stellungnahme zum KMK-Strategiepapier »Bildung in der digitalen Welt«, 2016, URL: <https://pygy.co/zw>, Stand: 17.02.2019.
- [Cl91] Claus, V.: Die Rolle der Sprache – Anforderungen an den Informatikunterricht. In (Burkert, J.; Peschke, R., Hrsg.): Weiterentwicklung des Informatikunterrichts – Folgerungen aus der Sicht von Lehrerbildung und Wissenschaft. Materialien zur Schulentwicklung 16, Hessisches Institut für Bildungsplanung und Schulentwicklung (HIBS), Wiesbaden, S. 148–158, 1991, ISBN: 3-88327-241-8.

- [De08] Denning, P. J.: *Great Principles of Computing*, 2008, URL: <https://pygy.co/NE>, Stand: 17.02.2019.
- [En18] Engbring, D.: Überlegungen zu einem Beitrag zur Lehrerbildung in der digital vernetzten Welt, Ein auf Erfahrungen gestützter Bericht und Diskussionsbeitrag. In (Thomas, M.; Weigend, M., Hrsg.): *Informatik und Medien – 8. Münsteraner Workshop zur Schulinformatik*. Books on Demand GmbH, Norderstedt, S. 95–106, Mai 2018, ISBN: 9783752849424, URL: <https://t1p.de/3k9u>, Stand: 17.02.2019.
- [Ge18] Gesche Joost and Franka Futterlieb and Stephan Noller and Jørn Alraun and Klaus Buß and Maxim Loick: *Calliope mini Dokumentation v1.0* auf github.io, 31. Jan. 2018, URL: <https://pygy.co/NF>, Stand: 17.02.2019.
- [Ha99] Hauf-Tulodziecki, A.; Bartsch, P. D.; Becker, K.-H.; Herzig, B.; Lehmann, G.; Magenheim, J.; Schelhowe, H.; Siegel, C.; Wagner, W.-R.: *Informatische Bildung und Medienerziehung. Empfehlung der Gesellschaft für Informatik e. V. erarbeitet von einem Arbeitskreis des Fachausschusses »Informatische Bildung in Schulen« (7.3)*, Informatik Spektrum, Band 23, Heft 2, 2000 und LOG IN Nr. 6 1999, Okt. 1999, URL: <https://pygy.co/NG>, Stand: 17.02.2019.
- [HP04] Humbert, L.; Puhlmann, H.: *Essential Ingredients of Literacy in Informatics*. In (Magenheim, J.; Schubert, S., Hrsg.): *Informatics and Student Assessment. Concepts of Empirical Research and Standardisation of Measurement in the Area of Didactics of Informatics*. Bd. 1. GI-Edition – Lecture Notes in Informatics (LNI) – Seminars S-1, Dagstuhl-Seminar of the German Informatics Society (GI) 19.–24. September 2004, Köllen Druck+Verlag GmbH, Bonn, S. 65–76, 2004, ISBN: 3-88579-435-7, URL: <https://pygy.co/NI>, Stand: 17.02.2019.
- [Ma99] Mayring, P.: *Einführung in die qualitative Sozialforschung*. Psychologie Verlags Union, München, 1999.
- [Pu03] Puhlmann, H.: *Informatische Literalität nach dem PISA-Muster*. In (Hubwieser, P., Hrsg.): *Informatik und Schule – Informatische Fachkonzepte im Unterricht INFOS 2003 – 10. GI-Fachtagung 17.–19. September 2003*, München. GI-Edition – Lecture Notes in Informatics – Proceedings P 32, Gesellschaft für Informatik, Köllen Druck + Verlag GmbH, Bonn, S. 135–144, 2003, ISBN: 3-88579-361-X, URL: <https://pygy.co/NJ>, Stand: 17.02.2019.
- [SR18] Seegerer, S.; Romeike, R.: *Was jeder über Informatik lernen sollte – Eine Analyse von Hochschulkursen für Studierende anderer Fachrichtungen*. In (Bergner, N.; Röpkke, R.; Schroeder, U.; Krömker, D., Hrsg.): *Hochschuldidaktik der Informatik HDI 2018 – 8. Fachtagung des GI-Fachbereichs Informatik und Ausbildung/-Didaktik der Informatik*. Bd. 8. *Commentarii informaticae didacticae (CID)*, Universitäts-Verlag Potsdam, Potsdam, S. 15–28, 2018, ISBN: 978-3-86956-435-7, URL: <https://t1p.de/k8cu>, Stand: 17.02.2019.

Debuggen im Unterricht – Ein systematisches Vorgehen macht den Unterschied

Tilman Michaeli,¹ Ralf Romeike²

Abstract:

Selbstständig Fehler in Programmcode zu finden und zu beheben stellt sowohl eine wichtige Fähigkeit als auch eine große Herausforderung beim Programmierenlernen dar. Debuggen unterscheidet sich von allgemeinen Programmierfertigkeiten und muss explizit gelehrt werden. Dennoch gibt es überraschend wenige Studien, Materialien und Konzepte, die sich mit der expliziten Vermittlung von Debuggingfähigkeiten beschäftigen. Eine relevante Debuggingfähigkeit ist ein systematisches Vorgehen bei der Fehlersuche. Dieser Beitrag analysiert die Wirksamkeit einer expliziten Vermittlung eines solchen systematischen Vorgehens im Unterricht, insbesondere hinsichtlich der Selbstwirksamkeitserwartungen und der resultierenden Debuggingleistung der Schülerinnen und Schüler. Zu diesem Zweck haben wir eine Intervention entwickelt, pilotiert und anschließend in einem Pre-Post-Kontrollgruppen-Test-Design untersucht. Die Ergebnisse zeigen sowohl einen signifikanten Anstieg der Selbstwirksamkeitserwartungen als auch der Debuggingleistung in der Versuchsgruppe.

Keywords: Debugging; Interventionsstudie; Systematisches Vorgehen

1 Einleitung

Programmieren erfordert eine Vielzahl an Kompetenzen, entsprechend stellt deren Vermittlung eine zentrale Herausforderung des Informatikunterrichts dar. Dabei müssen Schülerinnen und Schüler nicht nur Programmierkonzepte erlernen, sondern auch dazu befähigt werden, Lösungen zu finden, wenn sie mit Fehlern konfrontiert werden. Programme systematisch auf Fehler zu untersuchen, sie zu finden und zu beheben, stellt eine zentrale Kompetenz professioneller Entwicklerinnen und Entwickler dar, die zwischen 20 und 40 Prozent ihrer Arbeitszeit dafür aufwenden [Pe17]. Allerdings haben gerade Programmieranfängerinnen und -anfänger große Probleme im Umgang mit Fehlern. Dies stellt ein erhebliches Hindernis beim Programmierenlernen dar.

Darüber hinaus wird Debuggen im Kontext von Computational Thinking diskutiert [Ya11], und findet sich prominent in neueren Curricula wie dem britischen „Computing Curriculum“.

¹ Friedrich-Alexander-Universität Erlangen-Nürnberg, Didaktik der Informatik, Martensstraße 3, 91058 Erlangen
tilman.michaeli@fau.de

² Freie Universität Berlin, Didaktik der Informatik, Königin-Luise-Str. 24-26, 14195 Berlin ralf.romeike@fu-berlin.de

In deutschen Lehrplänen kommt der Begriff „debuggen“ nur selten vor, allerdings fordern fast alle die Fähigkeit, „mit Fehlern umzugehen“. Dennoch gibt es überraschend wenige Studien, die sich mit der expliziten Vermittlung von Debuggingfähigkeiten beschäftigen. Gleichzeitig mangelt es Lehrerinnen und Lehrern an Konzepten und Materialien für den Unterricht [MR19]. Eine relevante Debuggingfähigkeit ist ein systematisches Vorgehen bei der Suche und beim Beheben von Fehlern.

Das Ziel dieser Interventionsstudie ist es, den Einfluss einer expliziten Vermittlung eines systematischen Vorgehens zum Debuggen zu untersuchen. Im Fokus stehen dabei der Einfluss auf *(RQ1) Selbstwirksamkeitserwartungen* und *(RQ2) Debuggingleistung*. Dazu haben wir eine Intervention entwickelt, pilotiert und anschließend in einem Pre-Post-Kontrollgruppen-Test-Design untersucht.

2 Hintergrund

Debuggen beschreibt den Prozess der Suche und des Behebens von Fehlern. Debuggingfähigkeiten unterscheiden sich von allgemeinen Programmierfertigkeiten, wie [AEH05] oder [Mu08b] zeigen. Während ausgeprägte Debuggingfähigkeiten in der Regel auf entsprechende Programmierfähigkeiten schließen lassen, gilt die Umkehrung nicht unbedingt: Gute Programmierer sind nicht notwendigerweise auch gute Debugger. Dies wirft die Frage auf: Was macht einen „guten“ Debugger aus?

Als eine relevante Debuggingfähigkeit wird die Anwendung eines **systematischen Vorgehens**, also die systematische Verfolgung eines zielgerichteten Plans zur Fehlersuche und -behebung, gesehen. Dazu werden wiederholt Hypothesen formuliert, in Experimenten überprüft und gegebenenfalls verfeinert, bis die Ursache des Fehlers gefunden ist (vgl. z.B. [Gi91]). Oftmals kann dieser Prozess abgekürzt werden: Durch Erfahrung und damit die **Anwendung von Heuristiken und Pattern**, sind typische Fehler und deren mögliche Ursachen bekannt. Um dieses „Lernen aus früheren Fehlern“ zu unterstützen, führen viele professionelle Entwickler ein Debugging-„Tagebuch“ mit dem sie ihre Debugging-Erfahrung dokumentieren [Pe17]. Darüber hinaus spielt die Anwendung von **Debuggingstrategien** eine große Rolle im Debuggingprozess: Durch Strategien, wie beispielsweise dem Tracen des Kontrollflusses durch *print-f*-Debugging, dem Auskommentieren von Code oder Slicing, können Informationen gewonnen werden, die bei der Formulierung von Hypothesen und damit der Lokalisierung des Fehlers helfen [Sp18]. Ähnliches wird durch die Verwendung von **Werkzeugen** wie dem Debugger ermöglicht.

Murphy et al. [Mu08a] genauso wie Kessler and Anderson [KA86] argumentieren, dass Debuggingfähigkeiten explizit unterrichtet werden sollten. Nichtsdestotrotz gibt es überraschend wenige Studien sowohl in Bezug auf die universitäre Lehre als auch auf den Unterricht, die sich mit der expliziten Vermittlung von Debugginginhalten beschäftigen.

2.1 Debuggen in der Hochschullehre

Chmiel und Loui [CL04] verwendeten freiwillige Debuggingaufgaben, um die Debuggingfähigkeiten der Studierenden zu fördern. Es stellte sich heraus, dass Studierende, die die freiwilligen Debuggingaufgaben bearbeitet hatten, deutlich weniger Zeit für das Debuggen ihrer eigenen Programme benötigten. Dieser Zusammenhang spiegelte sich jedoch nicht in den Klausurergebnissen wider, die entgegen der Erwartungen nur leicht besser waren.

Katz und Anderson [KJ87] untersuchten den Effekt der Vermittlung verschiedener Vorgehensweisen (forward-reasoning, backward-reasoning) beim Debuggen. Unterschiedlichen Studierendengruppen wurde zunächst jeweils eine der Vorgehensweisen vermittelt, ehe sie ihr Vorgehen frei wählen konnten. Dabei zeigte sich, dass Studierende weiterhin das ihnen vermittelte Vorgehen anwendeten.

Allwood und Björhag [AB91] untersuchten, inwieweit schriftliche Debugging-Hinweise den Prozess unterstützen können. Während sich die Anzahl der Fehler zwischen Versuchs- und Kontrollgruppe nicht unterschied, war die Anzahl der beseitigten Fehler (insbesondere semantischer und logischer Art) bei Verfügbarkeit schriftlicher Hinweise signifikant höher. Da gleichzeitig keine Unterschiede in den verwendeten Strategien zwischen den Gruppen erkennbar waren, folgerten die Autoren, dass die Unterschiede auf einer höheren Ebene liegen müssen und vor allem ein systematisches Vorgehen beim Debuggen entscheidend sei.

Böttcher et al. [Bo16] vermittelten ein systematisches Debuggingvorgehen sowie die Verwendung des Debuggers in einer expliziten Einheit. Dabei wurde das Debuggingverfahren in einer Live-Demonstration verdeutlicht und eine Übung mit Debuggingaufgaben durchgeführt. Die Auswertung zeigte, dass nur wenige Studierenden den vermittelten systematischen Ansatz anwendeten, sondern schnell zu einem unsystematischen „Herumstöbern“ zurückkehrten.

2.2 Debuggen im Unterricht

Carver und Risinger [CR87] vermittelten einen Debuggingprozess mit LOGO mit vielversprechenden Ergebnissen: Sie gaben den Schülerinnen und Schülern eine Stunde Debugging-Training als Teil eines größeren LOGO-Curriculums. Sie nutzten ein Flow-Chart, das den Debuggingprozess charakterisiert, „Bug Mappings“ und Debugging-Tagebücher, die während der gesamten Zeit im Klassenzimmer vorhanden waren. Die Ergebnisse (ohne Kontrollgruppe) zeigten einen Wechsel von Brute-Force hin zu einem systematischen Vorgehen bei der Suche nach Fehlern. Darüber hinaus wurde für die Fehlersuche deutlich weniger Zeit benötigt. Die Schülerinnen und Schüler formulierten vor dem Ausprobieren des Codes mehr Hypothesen, achteten stärker auf den Kontrollfluss, nahmen weniger Code-Änderungen vor (insbesondere an fehlerfreien Stellen) und machten weniger neue Fehler.

In einer qualitativen Interviewstudie unter Informatiklehrkräften untersuchten [MR19] die aktuelle Rolle, die Debuggen im Unterricht spielt. Dabei ist festzustellen, dass Lehrkräfte bei Programmierproblemen viel und oft individuelle Hilfestellung leisten müssen. In der Folge eilen sie häufig von Schüler-Rechner zu Schüler-Rechner und versuchen zu helfen (gemeinhin als „Turnschuhdidaktik“ bekannt). Weiterhin mangelt es Lehrkräften an Konzepten und Materialien für die Vermittlung von Debugging. Insbesondere ist die Vermittlung eines systematischen Vorgehens – obgleich dessen Bedeutung – in der Schule unterrepräsentiert.

Zusammenfassend zeigt sich, dass nur wenige Untersuchungen zur expliziten Vermittlung von Debuggingfähigkeiten existieren und ein Großteil der Studien älter als 25 Jahre ist. Nichtsdestotrotz demonstrieren die vorhandenen Ergebnisse, dass sich Debuggen explizit vermitteln lässt. Sie deuten zudem darauf hin, dass ein systematisches Vorgehen eine entscheidende Rolle in einem erfolgreichen Debuggingprozess spielen kann. Eine empirische Untersuchung, inwieweit Programmieranfängerinnen und -anfänger innerhalb eines Unterrichtssettings von der Vermittlung eines solchen systematischen Vorgehens profitieren, fehlt bislang.

3 Vorgehen

Ziel dieser Untersuchung ist es, den Einfluss einer expliziten Vermittlung eines systematischen Vorgehens zum Debuggen im Unterricht zu untersuchen. Wir adressieren damit folgende Forschungsfragen:

- **(RQ1)** Hat die Vermittlung eines systematischen Vorgehens einen positiven Effekt auf die Selbstwirksamkeitserwartungen der Schülerinnen und Schüler?
- **(RQ2)** Hat die Vermittlung eines systematischen Vorgehens einen positiven Effekt auf die Debuggingleistung der Schülerinnen und Schüler?

3.1 Untersuchungsdesign

Um diese Forschungsfragen zu beantworten, haben wir ein Pre-Post-Kontrollgruppen-Test-Design gewählt. Zunächst wurde die Intervention in einer 10. Klasse für besonders leistungsstarke Schülerinnen und Schüler ($n = 14$, Greenfoot und Stride) pilotiert, um ausgehend von den gewonnenen Erkenntnissen der Durchführung Anpassungen vorzunehmen. Ergebnisse aus einer solchen Untersuchung ohne Kontrollgruppe helfen uns bei der Beantwortung der Forschungsfragen allerdings nur eingeschränkt, da mögliche Zuwächse in Selbstwirksamkeitserwartungen und Leistung der Schülerinnen und Schüler auch lediglich auf die zusätzliche Übung im Debuggen zurückzuführen sein könnten. Um den Einfluss der Intervention im Gegensatz zum reinen Üben von Debuggen, z.B. durch Debuggingaufgaben, zu untersuchen, haben wir zwei 10. Klassen als Versuchs- ($n = 13$) und Kontrollgruppe

($n = 15$) herangezogen. Dabei wurden explizit zwei Klassen ausgewählt, die von derselben Lehrkraft mit dem identischen Unterrichtskonzept (unter Verwendung von BlueJ und Java) unterrichtet wurden und die im Curriculum zum Untersuchungszeitpunkt gleich weit fortgeschritten waren.

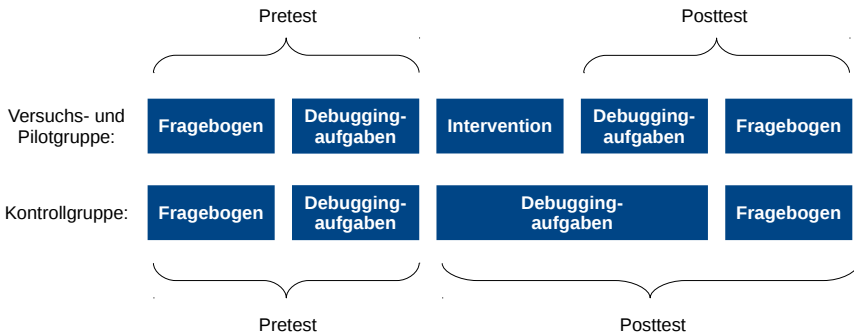


Abb. 1: Untersuchungsdesign

Der Ablauf der jeweils 90-minütigen Unterrichtseinheiten bestand aus einem Pretest, der Intervention (außer in der Kontrollgruppe) und einem Posttest. Wie in Abbildung 1 dargestellt, unterteilten sich Pre- und Posttest jeweils in einen Fragebogen zur Erhebung der Selbstwirksamkeitserwartungen (vier Items mit fünfstufiger Likert-Skala) und der Lösbarkeit der Aufgaben (nur im Posttest) sowie in Debuggingaufgaben zur Erhebung der Leistung der Schülerinnen und Schüler. Für Letzteres wurde die Anzahl der behobenen Fehler (analog zu [Mu08b]) herangezogen, wofür sowohl die Arbeitsblätter, auf denen Fehler und deren Behebung von allen Gruppen notiert werden sollten, als auch der Code ausgewertet wurde.

3.2 Intervention

Die Intervention vermittelt ein systematisches Vorgehen für die Suche und das Beheben von Fehlern. Das Vorgehen lehnt sich an die sogenannte *wissenschaftliche Methode* [Ze05] an, die von professionellen Entwicklerinnen und Entwicklern typischerweise implizit angewendet wird [Pe17]: Es werden wiederholte Hypothesen formuliert, in Experimenten verifiziert und gegebenenfalls verfeinert, bis die Ursache gefunden ist. Wir verwenden dabei eine didaktisch angepasste Variante dieses Vorgehens und unterscheiden explizit zwischen unterschiedlichen Fehlertypen: Compilezeit-, Laufzeit- und logischen Fehlern (vgl. Abbildung 2). Es wird das *Rückgängigmachen* von Änderungen betont, falls Maßnahmen zur Fehlerbehebung nicht erfolgreich sind – gerade da dieses Vorgehen für Schülerinnen und Schüler unnatürlich ist [Si08]. Hierdurch soll vermieden werden, dass die Schülerinnen

und Schüler im Zuge einer fehlgeschlagenen Fehlerbehebung zusätzliche Fehler einbauen – ein typisches Phänomen für Programmieranfängerinnen und -anfänger [GO86].



Abb. 2: Vermitteltes Vorgehen

Zum Üben des Debuggens werden typischerweise Debuggingaufgaben verwendet. Allerdings konfrontieren diese die Schülerinnen und Schüler mit einer großen Menge an fremdem Programmcode. Das Verständnis und die Einarbeitung in fremden Code stellt jedoch eine große Herausforderung dar [AB90]. Um sich dem eigentlichen Lern- und Untersuchungsziel – dem Debuggen *eigener* Programme – anzunähern, verwenden wir daher mehrere aufeinander aufbauende Prototypen eines Programms. Auf diese Art und Weise sehen sich die Schülerinnen und Schüler in jedem neuen Prototypen nur mit vergleichsweise wenig „fremdem“ Code konfrontiert und kennen sich im „alten“ Code bereits aus. Beispielsweise ist im ersten Prototypen des in der Pilotgruppe verwendeten Pongspiels lediglich die Bewegung des Balles umgesetzt, und im nächsten werden zusätzlich die Schläger und deren Steuerung eingefügt. Da die Debugging- und nicht die Testfähigkeiten der Schülerinnen und Schüler untersucht werden sollten, war die Anzahl der vorhandenen Fehler je Prototyp gegeben. Aus dem gleichen Grund wurde darauf geachtet, dass das Fehlverhalten des Programms schnell ersichtlich war, sodass direkt mit der Fehlerlokalisierung begonnen werden konnte.

4 Ergebnisse

4.1 (RQ1) Hat die Vermittlung eines systematischen Vorgehens einen positiven Effekt auf die Selbstwirksamkeitserwartungen der Schülerinnen und Schüler?

Zunächst untersuchen wir den Zuwachs der Selbstwirksamkeitserwartungen für Pilot-, Versuchs- und Kontrollgruppe pre und post, der sich als Mittelwert der vier Items ergibt. Die Antworten der fünfstufigen Likert-Skala wurden auf die Skala 0 (stimme nicht zu) bis 4 (stimme zu) abgebildet. Die Mittelwerte bewegen sich folglich zwischen 0 und 4.

Wir ermitteln, ob ein signifikanter Anstieg der Selbstwirksamkeitserwartungen zwischen Pre- und Post-Test innerhalb der einzelnen Gruppen feststellbar ist. Aufgrund der Stichprobengrößen verwenden wir dazu stets nicht-parametrisierte Verfahren zur Prüfung auf Signifikanz [Ra10]. Mithilfe des Wilcoxon-Vorzeichen-Rang-Tests – einem nicht-parametrischen Test für abhängige Stichproben – analysieren wir die Rangfolgen im Pre- und Posttest. In Tabelle 1 sind die jeweiligen Mediane und der p-Wert des Wilcoxon-Vorzeichen-Rang-Tests (H_0 : Kein oder negativer Versuchseffekt) dargestellt ³.

	Median pre	Median post	Wilcoxon-Test
Pilotgruppe	2,75	3,25	$p = 0,044^*$
Kontrollgruppe	2,25	2,50	$p = 0,083$
Versuchsgruppe	2,25	2,75	$p = 0,001^*$

Tab. 1: Einfluss auf Selbstwirksamkeitserwartungen

Wir sehen also in allen drei Gruppen einen Anstieg der Selbstwirksamkeitserwartungen. Dieser ist allerdings nur für die Pilot- und die Versuchsgruppe signifikant auf einem Signifikanzniveau von $\alpha = 0,05$. Die Effektstärken nach Cohen liegen bei $d = 0,56$ (Pilot) bzw. $d = 0,54$ (Versuch), dies entspricht einem mittleren Effekt [Co88].

Obleich das aktive Üben von Debugging die Selbstwirksamkeitserwartungen verbessert, scheint ein systematisches Vorgehen die Selbstwirksamkeitserwartungen stärker positiv zu beeinflussen.

4.2 (RQ2) Hat die Vermittlung eines systematischen Vorgehens einen positiven Effekt auf die Debuggingleistung der Schülerinnen und Schüler?

Für Unterschiede in der Debuggingleistung vergleichen wir Versuchs- und Kontrollgruppe in Pre- und Posttest. Die Debuggingleistung wird anhand der Anzahl der behobenen Fehler gemessen. Ein pre-post-Vergleich der Debuggingleistung innerhalb der einzelnen Gruppen analog zur Untersuchung der Selbstwirksamkeitserwartungen ist nicht zielführend, da in Pre- und Posttest unterschiedliche Fehler zu beheben waren.

Um festzustellen, ob sich die Leistung der Schülerinnen und Schüler der Versuchsgruppe signifikant von der Leistung der Schülerinnen und Schüler der Kontrollgruppe unterscheidet, prüfen wir, ob die beiden Stichproben derselben Grundgesamtheit entstammen. Nur wenn dies nicht der Fall ist, können wir von einem signifikanten Unterschied ausgehen. Auch hier greifen wir ob der Stichprobengröße wieder zu einem nicht-parametrisierten Test, dem Mann-Whitney-U-Test. Im Gegensatz zum Wilcoxon-Vorzeichen-Rang-Tests ist dieser Test für unabhängige Stichproben ausgelegt. Die p-Werte des Mann-Whitney-U-Tests (H_0 : Stichproben kommen aus derselben Grundgesamtheit) sind in Tabelle 2 dargestellt.

³ Signifikante Testergebnisse zu einem Signifikanzniveau von $\alpha = 0,05$ sind durch ein * gekennzeichnet.

	Mann-Whitney-U-Test
Versuch- vs. Kontrollgruppe Pre	$p = 0,191$
Versuch- vs. Kontrollgruppe Post	$p = 0,049^*$

Tab. 2: Einfluss auf Debuggingleistung

Dementsprechend können wir die Nullhypothese für den Vergleich der Pre-Tests auf einem Signifikanzniveau von $\alpha = 0,05$ nicht ablehnen: Die Debuggingleistung der Schülerinnen und Schüler unterscheidet sich vor Durchführung der Intervention nicht signifikant. Im Gegensatz dazu zeigt sich ein signifikanter Unterschied im Posttest: Die Schülerinnen und Schüler der Versuchsgruppe weisen eine höhere Debuggingleistung (Median = 4, bei insgesamt 9 zu behebbenden Fehlern) auf als die Schülerinnen und Schüler der Kontrollgruppe (Median = 2). Im Posttest wurden für das Ermitteln der Debuggingleistung Aufgaben mit höherem Schwierigkeitsgrad herangezogen, da in beiden Gruppen ein Lerneffekt zwischen Pre- und Posttest anzunehmen ist. Die Effektstärke nach Cohen liegt bei $d = 0,69$ und entspricht einem mittleren Effekt [Co88].

Die höhere Debuggingleistung spiegelt sich auch in der wahrgenommenen Schwierigkeit der Aufgaben durch die Schülerinnen und Schüler wider. Diese wurde ex post im Fragebogen mit Hilfe einer fünfstufigen Likert-Skala erhoben. Wiederum auf die Skala 0 (stimme nicht zu) bis 4 (stimme zu) abgebildet, ergeben sich folgende Mittelwerte:

	Aufgaben Pre	Aufgaben Post
Kontrollgruppe	3,07	1,47
Versuchsgruppe	3,23	2,92

Tab. 3: Mittelwerte für *Aufgaben gut lösbar*

Die Ergebnisse lassen darauf schließen, dass ein systematisches Vorgehen den Unterschied machen kann: Wird Schülerinnen und Schülern ein solches systematisches Vorgehen an die Hand gegeben, so können diese ihren Erfolg beim Lokalisieren und Beheben von Fehlern signifikant verbessern.

5 Diskussion und Fazit

Welche Bedeutung haben diese Ergebnisse für den Informatikunterricht? Nach [MR19] fehlt es Informatiklehrkräften an geeigneten Konzepten für den Themenkomplex *Debuggen*: Zwar sind verschiedene Debuggingstrategien sowie der Umgang mit Werkzeugen wie dem Debugger teils Unterrichtsgegenstand, allerdings spielt die Vermittlung eines systematischen Vorgehens im Unterricht bisher kaum eine Rolle. Diese Studie unterstreicht dabei, wie wichtig es ist, ein systematisches Vorgehen zu vermitteln und liefert Ansatzpunkte für eine Umsetzung. Dabei ist die Unabhängigkeit von Werkzeugen und (textbasierten)

Programmiersprachen – die in der Schulpraxis eine große Heterogenität aufweisen – zu betonen, so wurde dieselbe Intervention sowohl mit Java und BlueJ, als auch mit Stride und Greenfoot erfolgreich durchgeführt. Weiterhin legt dieser Ansatz einen Fokus auf die *Selbstständigkeit* der Schülerinnen und Schüler in der Fehlersuche und -behebung, um das Problem der „Turnschudidaktik“ zu adressieren.

Die vorgestellte Intervention stellt einen ersten Baustein für die Förderung von Debuggingfähigkeiten dar. Dieser sollte um die Vermittlung von konkreten Debuggingstrategien, Werkzeugen und Heuristiken erweitert werden.

Eine mögliche Einschränkung der Validität dieser Studie stellt die geringe Stichprobengröße und die nicht vorhandene Randomisierung der Schülerinnen und Schüler dar. Sie wurden von derselben Lehrkraft nach demselben Konzept unterrichtet und stammen von derselben Schule. Dies könnte die Aussagekraft bezüglich der Verallgemeinerung der Ergebnisse auf die Grundgesamtheit einschränken. Wir planen daher, diese Studie auf eine größere Stichprobe auszuweiten.

Zusammenfassend zeigt unsere Untersuchung, dass eine solche Intervention einen vielversprechenden Ansatz für die Vermittlung von Debuggingfähigkeiten darstellt. Die Vermittlung eines systematischen Vorgehens zum Finden und Beheben von Programmierfehlern hat einen positiven Einfluss auf die Debugging-Selbstwirksamkeitserwartungen. Schülerinnen und Schüler, die ein systematisches Vorgehen vermittelt bekommen haben, zeigen zudem höhere Leistungen im Debuggen als Schülerinnen und Schüler, die Debuggen ausschließlich geübt haben: Ein systematisches Vorgehen der Schülerinnen und Schüler macht den Unterschied.

Literaturverzeichnis

- [AB90] Allwood, Carl Martin; Björhag, Carl-Gustav: Novices' debugging when programming in Pascal. *International Journal of Man-Machine Studies*, 33(6):707–724, 1990.
- [AB91] Allwood, Carl Martin; Björhag, Carl-Gustav: Training of Pascal novices' error handling ability. *Acta Psychologica*, 78(1-3):137–150, 1991.
- [AEH05] Ahmadzadeh, Marzieh; Elliman, Dave; Higgins, Colin: An analysis of patterns of debugging among novice Computer Science students. *Proceedings of the 10th annual SIGCSE conference on Innovation and Technology in Computer Science Education (ITiCSE '05)*, 37(3):84–88, 2005.
- [Bo16] Bottcher, Axel; Thurner, Veronika; Schlierkamp, Kathrin; Zehetmeier, Daniela: Debugging students' debugging process. *Proceedings - Frontiers in Education Conference, FIE, 2016-Novem, 2016*.
- [CL04] Chmiel, Ryan; Loui, Michael C: Debugging: from Novice to Expert. *Proceedings of the 35th SIGCSE technical symposium on Computer science education - SIGCSE '04*, 36(1):17, 2004.
- [Co88] Cohen, Jacob: *Statistical power analysis for the behavioural sciences*. Hillsdale, NJ: erlbaum, 1988.

- [CR87] Carver, McCoy Sharon; Risinger, Sally Clarke: Improving children's debugging skills. In: Empirical studies of programmers: Second workshop. Ablex Publishing Corp., S. 147–171, 1987.
- [Gi91] Gilmore, David J: Models of debugging. *Acta psychologica*, 78(1-3):151–172, 1991.
- [GO86] Gugerty, Leo; Olson, G.: Debugging by skilled and novice programmers. *ACM SIGCHI Bulletin*, 17(4):171–174, 1986.
- [KA86] Kessler, Claudius M; Anderson, John R: A model of novice debugging in LISP. In: Proceedings of the First Workshop on Empirical Studies of Programmers. S. 198–212, 1986.
- [KJ87] Katz, Irvin R.; John R. Anderson: Debugging: An Analysis of Bug-Location Strategies. *Human-Computer Interaction*, 3(4):351–399, 1987.
- [MR19] Michaeli, Tilman; Romeike, Ralf: Current Status and Perspectives of Debugging in the K12 Classroom: A Qualitative Study. In (IEEE, Hrsg.): Global Engineering Education Conference (EDUCON). 2019.
- [Mu08a] Murphy, Laurie; Lewandowski, Gary; McCauley, Renée; Simon, Beth; Thomas, Lynda; Zander, Carol: Debugging: the good, the bad, and the quirky – a qualitative analysis of novices' strategies. Proceedings of the 39th SIGCSE technical symposium on Computer Science Education (SIGCSE '08), 40:163, 2008.
- [Mu08b] Murphy, Laurie; Simon, Beth; Fitzgerald, Sue; Lewandowski, Gary; Thomas, Lynda; Zander, Carol: Debugging: finding, fixing and flailing, a multi-institutional study of novice debuggers. *Computer Science Education*, 18(2):93–116, 2008.
- [Pe17] Perscheid, Michael; Siegmund, Benjamin; Taeumel, Marcel; Hirschfeld, Robert: Studying the advancement in debugging practice of professional software developers. *Software Quality Journal*, 25(1):83–110, 2017.
- [Ra10] Rasch, Björn; Friese, Malte; Hofmann, Wilhelm; Naumann, Ewald: , Quantitative Methoden 2: Einführung in die Statistik für Psychologen und Sozialwissenschaftler (3., erweiterte Auflage), 2010.
- [Si08] Simon, Beth; Bouvier, Dennis; Chen, Tzu-yi; Lewandowski, Gary; McCartney, Robert; Sanders, Kate: Common sense computing (episode 4): debugging. *Computer Science Education*, 18(2):117–133, 2008.
- [Sp18] Spinellis, Diomidis: Modern debugging: the art of finding a needle in a haystack. *Communications of the ACM*, 61(11):124–134, 2018.
- [Ya11] Yadav, Aman; Zhou, Ninger; Mayfield, Chris; Hambrusch, Susanne; Korb, John T: Introducing Computational Thinking in Education Courses. In: Proceedings of the 42Nd ACM Technical Symposium on Computer Science Education. SIGCSE '11, ACM, New York, NY, USA, S. 465–470, 2011.
- [Ze05] Zeller, Andreas: Why Programs Fail: A Guide to Systematic Debugging. Elsevier, 2005.

Zur Relevanz eines Prozessbereiches Interaktion und Exploration im Kontext informatischer Bildung im Primarbereich

Kathrin Müller,¹ Carsten Schulte,² Johannes Magenheim³

Abstract: Im Zusammenhang mit der aktuellen Diskussion über eine informatische Bildung für alle ab der Primarstufe wurden im vergangenen Jahr zwei Empfehlungen für Kompetenzen im Bereich der frühen informatischen Bildung veröffentlicht. Ein wesentliches Unterscheidungsmerkmal beider Dokumente ist die Einführung eines neuen Prozessbereiches P0 "Interagieren und Explorieren" in der Expertise der Stiftung Haus der kleinen Forscher. Im Rahmen der GI-Empfehlungen wird dieser Prozessbereich nur indirekt aufgenommen. Darüber hinaus gibt es zahlreiche Diskussionen über die Sinnhaftigkeit dieses neuen Prozessbereichs. Übergreifendes Ziel dieses Artikels ist es, die Bedeutung und Anwendung des Prozessbereichs P0 zu erläutern: Was verbirgt sich hinter diesem Prozessbereich und wie kann er mit anderen Inhaltsbereichen zu relevanten Kompetenzfeldern kombiniert und methodisch adäquat umgesetzt werden? Es wird u.a. gezeigt, dass mit dem Prozessbereich P0 ein der naturwissenschaftlichen Methodik ähnlicher experimenteller empirischer Zugang zum Verständnis von Informatiksystemen und Algorithmen ermöglicht wird.

Keywords: Interaktion und Exploration; Kompetenzen; Explorieren

1 Einleitung

Im Zuge der Diskussion um eine informatische Bildung für alle hat auch die Frage nach informatischer Bildung ab der Primarstufe stark zugenommen - und damit nach passenden Curricula. Im Zuge dieser Diskussion wurden zuletzt mit der Expertise der Stiftung Haus der kleinen Forscher (HdkF)[St17] und den GI-Standards [Ge19] für die Primarstufe zwei Empfehlungen für Kompetenzen im Bereich früher informatischer Bildung veröffentlicht. Beide Dokumente basieren auf den bisherigen Inhalts- und Prozessbereichen der GI-Standards. Damit liegt ihnen gemeinsam auch ein Kompetenzbegriff in Orientierung an Weinert [We01] zugrunde, nach dem sich die Performanz von Kompetenzen als konkrete Tätigkeiten (Prozessdimension) von Lernenden in einem spezifischen Kontext (Inhaltsdimension)

¹ Universität Paderborn, Didaktik der Informatik, Fürstenallee 11, 33102 Paderborn, Deutschland kathrin.mueller@uni-paderborn.de

² Universität Paderborn, Didaktik der Informatik, Fürstenallee 11, 33102 Paderborn, Deutschland carsten.schulte@uni-paderborn.de

³ Universität Paderborn, Didaktik der Informatik, Fürstenallee 11, 33102 Paderborn, Deutschland johannes.magenheim@uni-paderborn.de

beobachten und ggf. messen lassen. Die Empfehlungen des HdKFs enthalten einen neuen Prozessbereich P0 “Interagieren und Explorieren” (I&E) der in den GI Empfehlungen nur indirekt aufgenommen ([Ge19] S. VI,3,7) wird. In diesem Zusammenhang stellen sich eine Reihe von Fragen, zu deren kritischer Diskussion und Beantwortung der Artikel beitragen möchte:

- Kann I&E als eigenständiger Prozessbereich einer informatischen Kompetenz angesehen werden oder handelt es sich lediglich um eine Vermittlungsmethodik?
- Bis zu welchem Grad kann vom I&E eines Informatiksystems auf dessen innere, informatische Struktur und seine handlungsstrukturierende Wirkung auf dessen NutzerInnen geschlossen werden?
- Welche pädagogischen Interventionen seitens der Lehrkräfte sind für das zielgerichtete und erkenntnisfördernde I&E von Informatiksystemen durch Schülerinnen und Schüler erforderlich?
- Kann systematisches I&E im Informatikunterricht auf methodische Erfahrungen des naturwissenschaftlichen Unterrichts bezüglich der Beobachtung von Naturphänomenen und der Gestaltung von Experimenten zurückgreifen?
- Welchen Bildungswert beinhaltet systematisches I&E von Informatiksystemen im Informatikunterricht und welchen Beitrag zur informatischen Bildung kann damit geleistet werden?

Übergreifendes Ziel dieses Artikels soll es sein, zu erläutern was I&E im Rahmen informatischer Bildung für die Primarstufe bedeutet, welche Möglichkeiten dieser Prozessbereich bietet und welche Grenzen mit ihm verbunden sind. Dazu wird zunächst dargestellt, was sich hinter I&E als Kompetenzdimension verbirgt und wie der Prozessbereich mit den Inhaltsbereichen kombiniert werden kann. Im weiteren Verlauf wird insbesondere die Methodik des Explorierens im Bereich der informatischen Bildung theoretisch und anhand von Beispielen erläutert.

2 I&E als Prozessdimension eines Kompetenzmodells

Um den Stellenwert der Prozessdimension I&E im Rahmen des Kompetenzmodells der GI-Empfehlungen einzuordnen, ist zunächst festzuhalten, dass sich die GI-Empfehlungen nicht auf ein empirisch validiertes Kompetenzstrukturmodell beziehen, das mit Methoden der Kompetenzforschung entwickelt wurde. Es handelt sich vielmehr um ein im Diskurs mit ExpertenInnen entwickeltes normatives Modell, das einen hohen Grad an Praxisbezug zur Fachwissenschaft und Fachdidaktik Informatik aufweist, einen wesentlichen Beitrag zur Allgemeinbildung der SchülerInnen postuliert und mittlerweile im Hinblick auf die Gestaltung einschlägiger Curricula von großer bildungspolitischer Bedeutung ist. In den folgenden Abschnitten dieses Beitrags wird, analog zur Begründung der Prozess- und

Inhaltsdimensionen des GI-Kompetenzmodells, dargelegt, weshalb I&E als eigenständige Prozessdimension das GI-Modell sinnvoll erweitern kann.

2.1 Bezüge zu Fachwissenschaft und Fachdidaktik Informatik

In der Fachwissenschaft Informatik ist das I&E von Informatiksystemen von großer Bedeutung. Zahlreiche Vorgehensmodelle zur Softwareentwicklung weisen entsprechende Arbeitsphasen im Kontext der Anforderungsanalyse und dem Re-engineering von Informatiksystemen aus. Sie sind der Phase des Modellierens und Implementierens vorgeschaltet. Die Exploration bereits bestehender und die (Neu-)gestaltung von Informatiksystemen sind dabei wesentliche Phasen eines iterativen, zyklischen Entwicklungsprozesses von Informatiksystemen [Sc17]. Die in dieser Phase erforderliche Analyse von Anwendungsfällen (Use Cases) basiert zunächst auf dem systematischen Testen und Beobachten von Funktionen des Informatiksystems und der Interaktionen der involvierten Nutzer mit dem System bzw. deren Interaktionen untereinander. Interagieren und Explorieren eines Informatiksystems beschränkt sich daher nicht auf seine technischen Funktionen, sondern berücksichtigt auch dessen sozio-technischen Aspekte und trägt damit seinem Charakter als hybrides Interaktionssystem Rechnung [SB18]. Für Experten wäre ein weiterer Schritt einer derartigen Dekonstruktion des Informatiksystems die Analyse des vorhandenen Sourcecodes sowie der zugehörigen Dokumentation inkl. der evtl. vorhandenen (grafischen) Architektur- und Modellbeschreibungen.

In der Fachdidaktik gehören derartige Explorationsphasen ebenfalls zum fachlichen und methodischen Repertoire einiger fachdidaktischer Ansätze.

Beispiele bzw. Vorläufer des I&E mit Bezug zu den zuvor genannten fachwissenschaftlichen Aspekten existieren im Bezug auf die Betrachtung komplexer Systeme (Benutzung, Wartung und Erweiterung von Softwaresystemen) [Mo], die Analyse und Modellierung von Standardsoftware [Mo] und Benutzen-Analysieren-Gestalten [HH15]. Entsprechende Ansätze aus dem internationalen Raum sind beispielsweise "Consume before Produce" [In] und PRIMM (Predict-Run-Investigate-Modify-Make)[SW17].

Je nach Kenntnisstand der Lerngruppe können durch geeignete Aufgabenstellungen und Interventionen der Lehrkräfte von den Lernenden wichtige Kenntnisse und Kompetenzen gewonnen werden, die zum Verständnis und der Gestaltung von Informatiksystemen beitragen. Dieser Anspruch an den Informatikunterricht schlägt sich auch in entsprechenden empirisch fundierten Kompetenzstruktur- und -entwicklungsmodellen nieder [MSS].

Dabei dient das I&E mit Informatiksystemen aus dem Erfahrungsbereich der SchülerInnen nicht nur als Anknüpfungspunkt und motivierender Einstieg in einen informatischen Themenbereich im Unterricht. Es geht beim Erwerb dieser Kompetenzdimension vielmehr darum, Informatiksysteme aus dem Alltag der SchülerInnen als solche zu identifizieren, über ihre Funktion deren inhärente informatische Prinzipien zu erschließen und ihre Wirkungsweise als hybrides Interaktionssystem mit seinen Auswirkungen auf das soziale Handlungsumfeld der NutzerInnen zu verstehen. Wenn I&E in diesem Sinne im Informatikunterricht realisiert

wird, leistet er auch einen wichtigen Beitrag zur Allgemeinbildung der SchülerInnen, da sie mit dieser Kompetenz befähigt werden, auch zukünftige Informatiksysteme als solche zu identifizieren und sich wesentliche Fragen zu deren Funktion, ihren in Praxis umgesetzten informatischen Prinzipien und deren gesellschaftlichen Auswirkungen zu beantworten.

2.2 I&E im Kontext der Prozessdimensionen der GI-Empfehlungen

Nachdem die fachliche und fachdidaktische Bedeutung der Prozessdimension I&E (P0) diskutiert wurde, soll nun ihr Stellenwert im Zusammenhang mit den anderen Prozessdimensionen des GI-Modells erörtert werden. Diese Prozessdimensionen sind: Modellieren und Implementieren (P1), Begründen und Bewerten, (P2) Strukturieren und Vernetzen (P3), Kommunizieren und Kooperieren (P4) sowie Darstellen und Interpretieren (P5). Beim Blick auf diese Aufzählung wird deutlich, dass nur P1 einen genuin informatischen Bezug aufweist, während die anderen Prozessdimensionen eher generisch sind und auch in anderen Fächern als wichtige Kompetenzdimensionen angesehen werden könnten. Hauptsächlich durch ihre Kontextualisierung in den verschiedenen informatischen Inhaltsbereichen erhalten sie ihre Bedeutung für die Informatik. Daraus ergibt sich zweierlei: Einerseits zeigt sich, dass I&E eine wichtige Ergänzung bei den Prozessdimensionen des GI-Modells mit explizitem informatischem Bezug darstellt. Andererseits gilt es zu überprüfen, ob die Prozessdimension P0 auch in den verschiedenen Inhaltsbereichen I1 – I5 zu sinnvollen kontextualisierten Kompetenzanforderungen führt, wie es bei P1-P5 der Fall ist. Dies soll jetzt für die fünf Inhaltsbereiche Information und Daten (I1), Algorithmen (I2), Sprachen und Automaten (I3), Informatiksysteme (I4) sowie Informatik, Mensch und Gesellschaft (I5) in knapper Form geschehen.

Information und Daten (I1): Thematischer Schwerpunkt dieses Inhaltsbereiches bildet nach den GI-Empfehlungen für die primäre informatische Bildung u.a. die Codierung und Decodierung von Information sowie das Ver- und Entschlüsseln. Im folgenden Beispiel konzentrieren wir uns auf den Aspekt "Kommunikation und Sicherheit".

Kindern ist das Verschicken von Nachrichten z.B. in Form von Briefen bekannt. Einige haben sicherlich bereits erste Erfahrungen mit Geheimschriften oder Sprachen im Sinne einer vertraulichen Kommunikation unter Freunden gemacht. An diesem Punkt kann im Unterricht zunächst angeknüpft werden: Kinder verschicken unterschiedliche Nachrichten und überlegen, wie sie sicherstellen können, dass jemand Fremdes diese Nachricht nicht versteht. Dabei entwickeln sie z.B. selbst altersentsprechende Geheimschriften oder Sprachen und analysieren diese oder gegebene auf Sicherheitsmerkmale. Fragestellung ist hier beispielsweise: Was ist notwendig, damit sowohl Sender als auch Empfänger die Nachricht verstehen, aber kein Fremder? Im Anschluss kann ein Übergang zu bekannten Verschlüsselungsverfahren wie z.B. Skytale und Caesar stattfinden.

Ein weiterer, primär explorativer Aspekt, kann anschließend die Untersuchung von unterschiedlichen Passwörtern z.B. mit dem Verschlüsselungsverfahren Caesar sein. Fragestellung

gen während der Exploration sind beispielsweise: Welche Passwörter lassen sich schneller knacken als andere? Wie unterscheiden sich die unterschiedlichen Passwörter (z.B. in Länge oder Anzahl unterschiedlicher Zeichen etc.)? Somit können, wie bereits zuvor erwähnt, aus der Exploration heraus Aussagen über die Sicherheit von Passwörtern und Anforderungen an geeignete Passwörter gewonnen werden.

Folglich unterstützt die Prozessdimension P0 hier unter Anderem die Entwicklung der in den GI-Standards definierten Kompetenzen: "nutzen und entwickeln Vereinbarungen, um Daten zu verschlüsseln und zu entschlüsseln", "nutzen und entwickeln Vereinbarungen zur Übermittlung von Nachrichten" sowie "geben an, dass Vereinbarungen notwendig sind, um Daten zu codieren und zu decodieren" [Ge19].

Algorithmen (I2): Dieses Kompetenzfeld weist eine hohe Affinität zu dem im Hinblick auf Algorithmen beschriebenen Kompetenzbereich 5.5 der "KMK-Kompetenzen in der digitalen Welt" auf [Ku17]. Insbesondere der Kompetenzbereich 5.5.2 "Algorithmische Strukturen in genutzten digitalen Tools erkennen und formulieren" eröffnet im Zusammenhang mit I&E einen empirisch-analytischen Zugang zum Verständnis von Algorithmen in alltäglich verwendeten digitalen Werkzeugen und Medien. Dies betrifft eine große Bandbreite von Tools und Anwendungssituationen, wie etwa die Berechnung von Ergebnissen in der Tabellenkalkulation, die Suchfunktion in Texten, die Preisgestaltung bei der Produkt- und Servicesuche im Internet in Abhängigkeit von Tageszeit, verwendetem Gerät und Browser, etc. (zur Methodik vgl. Abschnitt 2.3).

Aus dem I&E von Informatiksystemen können über systematische Beobachtungen, z.B. mit entsprechenden Beobachtungsprotokollen, "Wenn-Dann-Beziehungen" gefolgert werden, welche die Systemreaktionen als Folge von User-Interaktionen mit dem System beinhalten. Gleiches gilt für das beobachtbare "Verhalten" eines Informatiksystems beim Input von unterschiedlichen Messwerten über Sensoren (z. B. bei Robotern). Auf diese Weise kann auf interne Verarbeitungsmechanismen des Systems und den ihnen zugrundeliegenden Algorithmen geschlossen werden. An diesem Punkt existiert eine enge Verzahnung mit dem Inhaltsbereich Informatiksysteme, die sich aus der Tatsache ergibt, dass Algorithmen immer in einem Informatiksystem implementiert sind. Ferner kann z.B. die empirische Beobachtung des Laufzeitverhaltens unterschiedlicher Algorithmen in Abhängigkeit von der Dateneingabe einen Zugang zu Komplexitätsbetrachtungen von Algorithmen eröffnen.

Die Prozessdimension I&E trägt hier unter anderem zum Aufbau der in den GI-Standards definierten Kompetenzen "beschreiben Algorithmen alltagssprachlich" und "vergleichen Algorithmen unter Verwendung der Fachsprache" [Ge19] bei.

Sprachen und Automaten (3): Hier können den SchülerInnen aus ihrem Alltag bekannte Automaten (z.B. Getränkeautomat, Verkehrsampel) als Explorationsgegenstand eines Informatiksystems dienen. Systematisches Beobachten und eine Variation der Eingabemöglichkeiten erschließen dann den Zugang zu wesentlichen Aspekten der informatischen Konzepte von Automat und Sprache, wie Ein- und Ausgabealphabet, Systemzustände,

akzeptierte Eingaben etc.. Durch die Untersuchung der Lebenswelt im Bezug auf Automaten trägt die Exploration somit zur in den GI-Standards definierten Kompetenz: "beschreiben Automaten in ihrer Lebenswelt als selbsttätig arbeitende Maschinen" [Ge19] bei.

Informatiksysteme (I4): Das I&E mit/von Informatiksystemen ist, wie zuvor bereits beschrieben, ein zentraler Inhaltsbereich dieser Prozessdimension und kann auch in der Grundschule mit unterschiedlichen altersadäquaten Systemen, z.B. Robotern, durchgeführt werden. Wir zeigen hier am Beispiel des Lernroboters Ozobot⁴ wie das Informatiksystem Roboter exploriert werden kann.

Ozobots (Ozobot bit) besitzen lediglich einen Farbsensor, sowie als Aktoren eine Farb-LED und zwei Motoren zum Antrieb der Räder. Sie folgen schwarzen Linien und reagieren auf vordefinierte Farbcodes. Darüber hinaus sind sie mit einer Blocksprache programmierbar. Aufbau und Funktionsweise können z.B. wie folgt exploriert werden:

Ein Unterricht mit Ozobots kann z.B. mit der Exploration der Linienverfolgung starten: Wie verhält sich der Ozobot an Kreuzungen? Nimmt er immer die gleiche Richtung? Ändert sich die Richtung? Gibt es ein Muster? Welche Hypothesen gibt es zur Ursache dieses Verhaltens? Im weiteren Verlauf kann dann exploriert werden, welche Eigenschaften die Linien haben müssen, auf denen der Roboter fährt. Hier können unterschiedliche Farben, unterschiedlich breite Linien, unterschiedliche Kurven etc. getestet werden. Die Ergebnisse können auf entsprechenden Arbeitsblättern protokolliert werden. Aus diesen Protokollen können die Kinder anschließend Schlussfolgerungen ziehen, wie z.B. welche Farben der Ozobot "erkennt" und wie er auf die Farben reagiert. Darauf basierend können dann Hypothesen über die Architektur des Ozobots aufgestellt werden. Wie kann es dazu kommen, dass die LED in der Farbe der Linie leuchtet? Welche Bauteile sind dazu notwendig und welche Prozesse innerhalb des Ozobots? Farbcodes können exploriert werden, indem das Verhalten des Ozobots bei unterschiedlichen Farbcodes protokolliert wird, oder indem die Reaktion des Roboters vorgegebenen Reaktionsmöglichkeiten zugeordnet werden. Auf diese Weise können sich die Kinder zunächst die Funktionsweise erschließen. Anschließend können, z.B. auf Basis der bereits aufgestellten Hypothesen, Überlegungen zur Architektur aufgestellt werden, wie beispielsweise: Der Roboter braucht etwas, mit dem er die Farben und Linien erkennen kann, er besitzt einen Motor und Räder, eine LED, etc. Über diese Überlegungen kann dann ggf. mit Unterstützung der Lehrperson auf die einzelnen Bauteile des Roboters und das EVA-Prinzip geschlossen werden.

Ein Anwendungsbeispiel über die Ozobots hinaus ist die Möglichkeit, im Alltag mittels protokollierter Beobachtung nach Informatiksystemen zu suchen. Wo ist mir in meinem Alltag ein Informatiksystem begegnet? Warum ist dies ein Informatiksystem?

Die Prozessdimension I&E trägt hier ebenfalls zum Aufbau von in den GI-Standards definierten Kompetenzen bei. Dies sind unter anderem: "interagieren zielgerichtet mit Informatiksystemen" und "geben grundlegende, allgemeingültige Beschreibungen der Funktion und Arbeitsweise von Informatiksystemen an (EVA-Prinzip)" [Ge19].

⁴ <https://ozobot.com/products>

Informatik, Mensch und Gesellschaft (I5): Die Prozessdimension I&E betrachtet Informatiksysteme nicht nur als technische Artefakte, sondern als sozio-technische Systeme mit einer hybriden Interaktionsstruktur. Insofern werden die mit dem Informatiksystem verbundenen Interaktionen stets auch im sozialen Kontext seiner Nutzung gesehen. So kann mit der Prozessdimension I&E beispielsweise das Nutzungsverhalten unterschiedlicher User bei einer gegebenen Benutzungsoberfläche beobachtet und diese Beobachtungen im Hinblick auf softwareergonomische Fragestellungen bewertet werden. Daraus lassen sich dann wiederum Anforderungen an die Re-konstruktion der Benutzungsoberfläche im Rahmen einer Systemgestaltung ableiten.

2.3 Methodik der Prozessdimension I&E

Nachdem die Kontextualisierung der Prozessdimension I&E in den verschiedenen Inhaltsbereichen aufgezeigt wurde und sich daraus relevante informatische Kompetenzanforderungen ergeben, muss nun geklärt werden, wie zugehörige Lernprozesse zu einem für SchülerInnen motivierenden und erfolgreichen Kompetenzerwerb organisiert werden können.

Im Vordergrund steht das eigenständige explorative Handeln der SchülerInnen im Sinne der Prozessdimension I&E, das für sie zu informatischem Kenntniserwerb führen kann (vgl.[Be16]).

Dabei sollte im konstruktivistischen Sinn einerseits die natürliche Freude der Kinder am forschenden, selbst-entdeckenden Lernen gefördert und andererseits Hilfestellung durch die Lehrkraft bei der systematischen Gestaltung und Auswertung der Beobachtungen beim Umgang mit Informatiksystemen gegeben werden. Dies stellt die Lehrkraft vor anspruchsvolle Moderationsaufgaben bei der kognitiven Aktivierung der SchülerInnen. In der Grundschule und im Primarbereich können dabei ko-konstruktive Methoden, wie z.B. das "sustained shared thinking" [Ho12] zum Einsatz kommen, bei denen die SchülerInnen durch offene Fragen und Lernimpulse der Lehrkraft weitgehend autonom Aussagen und Vermutungen zu ihren Explorationserfahrungen äußern können. In einem weiteren Schritt können die Beobachtungen dann systematisch und formalisiert erfasst, analysiert und auf der Basis dieses Datenmaterials Hypothesen zum Zusammenhang von beobachtetem Systemverhalten und der inneren Struktur bzw. den inneren Zuständen des Informatiksystems formuliert werden. Daran könnte sich eine Phase zur (grafischen) Beschreibung von Systemzuständen, Datenstrukturen bzw. Ein- und Ausgabewörtern sowie ggf. von Algorithmen des erkundeten Informatiksystems anschließen. Je nach Intention und Zielsetzung kann dann ein Übergang zur (Neu-)gestaltung des Informatiksystems mit Modellierungs- und Implementierungsphase erfolgen.

Falls Lernprozesse zum Kompetenzerwerb in Bezug auf I&E in den verschiedenen Inhaltsbereichen nach diesen Prinzipien organisiert werden wird deutlich, dass die Methodik:

- über ein anfängliches unverbindliches Herumspielen und Ausprobieren nach dem "trial and error"- Prinzip hinausgeht und sinnerschließende Fragen und eine auswertbare Beobachtungsmethodik beinhalten sollte,

- gezielte, erkenntnisfördernde Interventionen der Lehrkraft, abhängig vom Kenntnisstand der Lernenden, den Lernprozess unterstützen können, ohne das entdeckende Lernen der SchülerInnen zu behindern,
- systematische empirische Beobachtungen von auszuwählenden relevanten Parametern beinhaltet, die im Hinblick auf Fragestellungen zum Informatiksystem ausgewertet werden können und zu einem Erkenntnisgewinn über dieses System beitragen.
- die Beobachtungen und Erkenntnisse so angelegt werden, dass von der von außen sichtbaren Funktionsebene (Nutzerperspektive) ausgehend anhand geeigneter Merkmale/Ereignisse/Beobachtungen auf die innere Bauweise (Architektur- und Konstruktionsperspektive) geschlossen werden kann, bzw. zumindest begründete Vermutungen über diese Bauweise angestellt werden können.

Besonders der vorletzte methodische Aspekt verdeutlicht, dass hier im Informatikunterricht auch Methoden eingesetzt werden können, die man aus dem naturwissenschaftlichen Unterricht beim Durchführen und Auswerten von Experimenten kennt. Das für das I&E ausgewählte Informatiksystem dient dabei nicht lediglich als Einstiegsbeispiel, sondern ist zentraler Unterrichtsgegenstand zum Kompetenzerwerb.

Eine differenzierte Darstellung, wie im Kontext der Prozessdimension P0 das Explorieren und Gestalten von Informatiksystemen als iterativer, wechselseitig verwobener Prozess erfolgen kann, ist in [Sc17] zum Explorations- und Gestaltungskreis in der informatischen Bildung sowie zum Informatikkreis als naturwissenschaftlich-technischer Lernmethode beschrieben [Fr19]. Beiden Konzepten liegt die Auffassung zugrunde, dass das Explorieren eines existierenden Informatiksystems, ganz im Sinne des informatischen Re-engineerings, nicht nur zum Systemverständnis beiträgt, sondern auch zu dessen Weiterentwicklung oder zur Neugestaltung eines Informatiksystems führen kann, das auf ähnlichen Prinzipien aufgebaut ist, wie das zuvor explorierte.

3 Fazit

Die Prozessdimension I&E verknüpft das Nutzen eines digitalen Artefakts (Interaktion) mit der fachlich orientierten Analyse und Erkundung der Architektur (dazu können zählen die innere Bauweise, Wirkprinzipien, Funktionsweisen, verwendete informatische Konzepte wie Datenstrukturen und Algorithmen, usw.). Die genaue Bezeichnung von Exploration, sowie Ziele und methodische Hinweise sind zusammengefasst in Abb. 1 dargestellt. Durch die im Papier beschriebenen fachwissenschaftlichen und fachdidaktischen Bezüge wird die Rolle von I&E in informatischen Kontexten verdeutlicht. Diese Rolle wurde anhand von Beispielen und Kompetenzen, auch im Kontext der GI-Standards, insbesondere der Primarstufe, weiter konkretisiert.

Zusammenfassend zeigt sich, dass I&E als Prozessbereich nicht nur auf die Methodik beschränkt ist. In gewisser Weise stellt er das ergänzende Gegenstück zum Prozessbereich

Modellieren und Implementieren dar. Dieser lässt sich ebenfalls nicht auf die Methodik beschränken. Für das Programmieren braucht man wie für das I&E nicht nur Prozesswissen, sondern auch Inhaltswissen; ebenso sind beide auf spezifische unterrichtliche Vorgehensweisen bezogen, die methodisch unterschiedlich ausgestaltet werden können und müssen. Der Prozessbereich I&E kann als vorgeschaltete Erkundungsphase für das 'Re-engineering' eines bestehenden Informatiksystems angesehen werden oder lediglich dem Wissenserwerb der Lernenden dienen, um ggf. in einem späteren Transferprozess das Wissen zur (Neu-)gestaltung eines Informatiksystems anzuwenden. In dieser Sichtweise ist der Prozessbereich Modellieren und Implementieren alleine nicht ausreichend für eine insgesamt gelingende informatische Bildung. Denn die konstruierende Perspektive des Modellierens und Implementierens kann und sollte durch die analytische und reflektierende Sichtweise der Exploration ergänzt werden. In diesem Sinne gehören auch der Explorations- und Gestaltungskreis zusammen[Sc17].

Exploration bezeichnet:
<ol style="list-style-type: none"> 1. ein Verfahren, um Erkenntnisse zu erlangen (methodische bzw. Prozesskomponente) 2. einen generellen Zugang zu Technologie im Alltag (Bildungspolitik- bzw. Kompetenzkomponente: Explorationsfähigkeit als Kompetenz, sich aus einer Nutzungsperspektive die innere Bauweise zu erarbeiten und aus diesen Erkenntnisse Rückschlüsse und Bewertungen der Technologie im Kontext ableiten können) 3. einen unterrichtlichen Zugang zu Technologie (Lernen aus Beispielen, nicht nur konstruierend, auch analysierend und reflektierend vorgehen)
Zielaspekte:
<ol style="list-style-type: none"> 1. Erkennen, dass es unterschiedliche Betrachtungsweisen gibt: Benutzersicht, Konstruktionssicht... 2. Zwischen den Betrachtungsweisen wechseln können 3. Notwendigkeit des Zusammenhangs der beiden Sichtweisen erkennen können (Duale Natur) 4. Duale Natur als fachliche Basis für Diskussion der Relevanz nutzen können- gesellschaftliche und technische Perspektive verknüpfen können.
Methodische Hinweise:
<ol style="list-style-type: none"> 1. Explorieren von außen nach innen. 2. Analytischer, empirischer, reflektierender, interpretierender Zugang (im Gegensatz zum konstruktiven Zugang). 3. Sollte wissens- und kompetenzbasiert sein. 4. Sollte hypothesengeleitet sein.

Abb. 1: Was ist Exploration?

Literaturverzeichnis

- [Be16] Benton, Laura; Hoyles, Celia; Kalas, Ivan; Nos, Richard: Building mathematical knowledge with programming: insights from the ScratchMaths project. In: Proceedings of the Constructionism 2016. S. 25–32, 2016.
- [Fr19] Franke-Wiekhorst, Antonia; Günther, Christine; Brünger, Karen; Magenheimer, Johannes; Romeike, Ralf: „Der Informatikkreis“: Kinder von drei bis zehn Jahren beim Forschen in Informatik begleiten – ein methodisch-didaktisches Material. GdSU-Journal, (9):10–21, 2019.
- [Ge19] Gesellschaft für Informatik (Hrsg.): Kompetenzen für informatische Bildung im Primarbereich. Empfehlungen der Gesellschaft für Informatik e.V. erarbeitet vom Arbeitskreis "Bildungsstandards Informatik im Primarbereich" unter Koordinierung von Ludger Humbert. Beilage zu LOG IN, 39. Jahrgang (2019), Heft, 2019.

- [HH15] Hellmig, Lutz; Hempel, Tino: Benutzen – Analysieren – Gestalten – Verankern als didaktische Schrittfolge im Informatikunterricht. In (Gallenbacher, Jens, Hrsg.): Informatik allgemeinbildend begreifen. Gesellschaft für Informatik e.V., Bonn, S. 145–154, 2015.
- [Ho12] Hopf, Michaela: Sustained Shared Thinking im frühen naturwissenschaftlich-technischen Lernen. Waxmann, 2012.
- [In] : Instructional Design of a Programming Course: A Learning Theoretic Approach.
- [Ku17] Kultusministerkonferenz, Hrsg. Strategie der Kultusministerkonferenz „Bildung in der digitalen Welt“ - Beschluss der Kultusministerkonferenz vom 08.12.2016 in der Fassung vom 07.12.2017. KMK, 2017.
- [Mo] Modellierung von Standardsoftware aus didaktischer Sicht.
- [MSS] Magenheim, Johannes; Schubert, Sigrid; Schaper, Niclas: Entwicklung von qualitativen und quantitativen Messverfahren zu Lehr-Lern-Prozessen für Modellierung und Systemverständnis in der Informatik (MoKoM). In (Bayrhuber, Horst; Harms, Ute; Muszynski, Bernhard; Ralle, Bernd; Rothgangel, Martin; Schön, Lutz-Helmut; Vollmer, Helmut J.; Weigand, Hans-Georg, Hrsg.): Formate fachdidaktischer Forschung, S. 109–128. Waxmann.
- [SB18] Schulte, Carsten; Budde, Lea: A Framework for Computing Education: Hybrid Interaction System: The Need for a Bigger Picture in Computing Education. In: Proceedings of the 18th Koli Calling International Conference on Computing Education Research. Koli Calling '18, 2018.
- [Sc17] Schulte, C.; Magenheim, J.; Müller, K.; Budde, L.: The design and exploration cycle as research and development framework in computing education. In: 2017 IEEE Global Engineering Education Conference (EDUCON). S. 867–876, 2017.
- [St17] Stiftung Haus der kleinen Forscher; Bergner, N.; Budrich, Barbara; Köster, H.; Magenheim, J.; Müller, K.; Romeike, R.; Schroeder, U.; Schulte, C.; Hubwieser, P.: Frühe informatische Bildung - Ziele und Gelingensbedingungen für den Elementar- und Primarbereich. Wissenschaftliche Untersuchungen zur Arbeit der Stiftung "Haus der kleinen Forscher". Verlag Barbara Budrich, 2017.
- [SW17] Sentance, Sue; Waite, Jane: PRIMM: Exploring pedagogical approaches for teaching text-based programming in school. In: Proceedings of the 12th Workshop on Primary and Secondary Computing Education. ACM, S. 113–114, 2017.
- [We01] Weinert, Franz: Concept of Competence: A conceptual clarification. In (Rychen, D. S.; Salganik, L., Hrsg.): Defining and Selecting Key Competencies, S. 45–65. Huber, 2001.

Entwicklung ökologischen Denkens im Informatikunterricht

Ilona Petrenko¹, Marco Thomas²

Abstract: Die hohe Relevanz von Ökologie und Informatiksystemen (IS) für unsere heutige Welt ist vermutlich unstrittig. In ökologischer Hinsicht beeinflusst der Zustand der Welt die Möglichkeiten der Menschheit, auf der Erde zu überleben. IS gehören zunehmend zum Werkzeug der Menschheit und haben sich in allen Bereichen des Lebens verbreitet. Weniger offensichtlich sind die Zusammenhänge zwischen Ökologie und IS, sowohl im positiven als auch im negativen Sinne. Um den rationalen Umgang mit beiden zu realisieren, kann man ökologisches Denken im Zusammenhang mit IS entwickeln. Die Grundidee des Projektes ÖKI ist es, ökologisches Denken der Schülerinnen und Schüler im Rahmen des Faches Informatik zu vertiefen. Dieser Beitrag dient zur Vorstellung von ersten Studien und Modulen zur unterrichtlichen Vermittlung ökologischer Aspekte im Kontext von Informatiksystemen.

Keywords: Informatik, Unterricht, Schule, Ökologie, Bildung, Informatiksystem, Umwelt

1 Einleitung

Die Idee des Projektes „Ökologie als Kontext im Informatikunterricht“ (ÖKI) entstand im Rahmen eines Praktikums unter Förderung der Deutschen Bundesstiftung Umwelt³ im Jahr 2017. Der Projekttitel mag ungewöhnlich klingen, aber sowohl Informatik mit den sich ergebenden digitalen Artefakten als auch Ökologie sind zwei zentrale Schlüsselthemen in unserer Welt geworden. Obwohl es nicht immer sichtbar wird, sind die Informatik und ihre Systeme in viele ökologische Zusammenhänge eingebunden und von großer Bedeutung – so beschränkt sich der Begriff »Ökologie« in seiner ursprünglichen Bedeutung nicht nur auf den schonenden Umgang mit natürlichen Ressourcen, sondern befasst sich allgemein mit den Wechselwirkungen von Organismen und ihrer unbelebten Umwelt. Das Denken in diesen Zusammenhängen kann als »ökologisches Denken« bezeichnet werden.

Als eine negative Folge der Einwirkung von Informatiksystemen (IS) auf die Umwelt kann man den hohen Energieverbrauch nennen oder auch die nicht einfache Wiederverwertung von wertvollen Materialien in der Elektronik. Es gibt aber auch positive Auswirkungen, wie eine mögliche Reduzierung des Papierverbrauchs in der Zukunft mit Hilfe des

¹ Westfälische Wilhelms-Universität Münster, Didaktik der Informatik, Corrensstraße 80, 48149 Münster, ilona.petrenko@uni-muenster.de

² Anschrift s.o., Marco.Thomas@uni-muenster.de

³ <https://www.dbu.de> (15.02.2019)

elektronischen Dokumentenverkehrs⁴, Videotelefonate zur Reduzierung von Reisen oder eine ökologisch nachhaltigere Landwirtschaft. Aus diesen wenigen Überlegungen kann eine mögliche allgemeinbildende Relevanz hinsichtlich einer Auseinandersetzung mit Informatiksystemen in ökologischen Kontexten abgeleitet werden, um ökologisches Denken zu fördern. Ökologisches Denken basiert auf dem vernetzten Denken und erfordert eine kontinuierliche Auseinandersetzung mit den Auswirkungen auf die Umwelt, die eine bestimmte Tätigkeit oder Entscheidung auslösen kann (vgl. [Wi19]). Die Entwicklung ökologischen, vernetzten Denkens ist ein wichtiges, fächerverbindendes Ziel, um mit unserer Umwelt in Einklang zu gelangen.

Ist es möglich und sinnvoll, ökologisches Denken gerade im Informatikunterricht zu entwickeln? Diese Frage führt zu ersten Forschungsfragen:

1. Welche Projekte und Literatur existieren zum Spannungsfeld Ökologie-Medien-Informatik, die Impulse für einen ökologischen Informatikunterricht geben?
2. Welche ökologischen Probleme bzw. Fragen können in den Informatikunterricht integriert werden?
3. Wie muss ein ökologischer Informatikunterricht konzipiert sein bzw. welche fachdidaktischen Ansätze sind geeignet?
4. Welche Informatiksysteme gibt es, die untersucht werden könnten?
5. Was umfasst ökologisches Denken und warum sollte es in den Informatikunterricht integriert werden?

Die Methodik ist in dem Projekt ÖKI noch nicht endgültig festgelegt, aber neben einer Literaturrecherche und dem Erfassen von Expertenmeinungen (s. u.) haben wir mit der Entwicklung von Modulen begonnen, die mittels Design-based-research [DRC03] evaluiert und optimiert werden sollen. Auch eine Anbindung an das Projekt „Informatik im Kontext“⁵ (IniK) erscheint uns geeignet und wurde mit Studierenden in einem Seminar diskutiert und bearbeitet.

2 Konzeptentwicklung

2.1 Zusammenhang Ökologie und Informatik

Die Auseinandersetzung mit und die Simulation von dynamischen Systemen – insbesondere in ökologischen und ökonomischen Zusammenhängen – wurde bereits vor 25 Jahren für den Informatikunterricht diskutiert ([Th93] u. a.) und ist auch heute noch in

⁴ Allerdings steigt derzeit der Papierverbrauch eher, insbesondere aufgrund des zunehmenden Paketversands. Immerhin haben ca. 30% der Unternehmen ihre Kommunikation weitestgehend digitalisiert. S. <https://www.verbraucherservice-bayern.de/themen/umwelt/papierverbrauch-steigt-trotz-digitalisierung> (15.02.2019).

⁵ Informatik im Kontext. <http://www.informatik-im-kontext.de> (15.02.2019)

einigen Curricula verankert⁶. Die Idee, eine »Dimension Ökologie« in den Informatikunterricht zu integrieren, wurde von J. Koubek im Zusammenhang mit der Integration von Informatik und Gesellschaft für das Projekt „Informatik im Kontext“ (IniK) eingebracht. Koubek hat auch Kriterien für die Auswahl von geeigneten Kontexten beschrieben [In19]:

- Mehrdimensionalität: rechtliche, ökonomische, ökologische, ethische, informatische Fragestellungen sind u.a. möglich.
- Breite: Der Kontext ist gesellschaftlich relevant.
- Tiefe: Der Kontext ist informatisch relevant.
- Lebenswelt: Bezug zur Lebenswelt der Schülerinnen und Schüler.
- Stabilität: Der Kontext ist über einen längeren Zeitraum relevant.

Auf den Folien zum Vorlesungs-Podcast „Dimensionen der Informatik - Ökologie“ nennt Koubek auch ethische Aspekte, wo er unter anderem das Prinzip der Nachhaltigkeit definiert: „*Nachhaltigkeit bedeutet, dass die gegenwärtige Generation ihre Bedürfnisse befriedigt, ohne die Fähigkeit der zukünftigen Generation zu gefährden, ihre eigenen Bedürfnisse befriedigen zu können*“ [KK08]. Ein nachhaltiges Handeln ist sicherlich eine Grundlage ökologischen Denkens und gehört zur Allgemeinbildung.

2.2 Existierende Projekte

In einem ersten Schritt auf der Suche nach passenden Themen haben wir verschiedene Quellen analysiert, um existierende Erfahrungen in dem Zusammenhang Schule-Informatik-Ökologie zu analysieren und aufzuarbeiten (vgl. 1. Forschungsfrage). Es ergab sich ein Modell, in das Projekte oder Themen in die Bereiche Informatik, Mediennutzung und Ökologie und ihre Schnittmengen eingeordnet werden können (Abb.1).

Die senseBox ist beispielsweise ein erfolgreiches Projekt der Geoinformatiker in Münster zum Messen von Umweltphänomenen und wird mittlerweile kommerziell vertrieben [Se19]. Allerdings liegen bisher kaum Erfahrungen zum Einsatz im Informatikunterricht⁷ vor, eher zum Einsatz im Fach Geographie. Das länderübergreifende MINT-Bildungsprogramm Experimento [Ex19] scheint vor allem für Kinder im Grundschulalter geeignet zu sein. Unsere Suche und Auswertung hierzu ist nicht abgeschlossen und offen für Neues.

⁶ Unterrichtsbeispiel zu dynamischen Systemen für den Hamburger Schulinformatik Lehrplan in der Oberstufe: <https://haukemorisse.de/blog/?p=2300> und <https://bildungsserver.hamburg.de/simulation/> (17.02.2019)

⁷ Ein Vorteil der senseBox liegt in dem abgestimmten und vorbereiteten Informatiksystem. Denkbar sind sicherlich auch der Einsatz von alternativen Informatiksystemen, die selbst zusammengebaut werden.

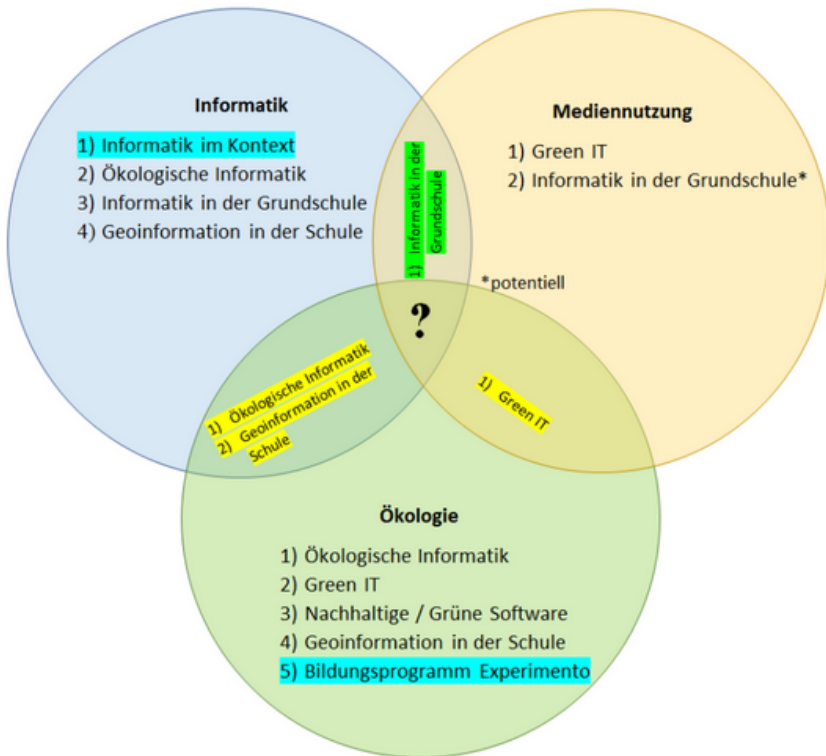


Abb. 1: Modell zur Gruppierung der Themen im Projekt ÖKI

2.3 Überlegungen für ein Konzept

Für die Konzeptentwicklung ist es erforderlich, praktische Aufgaben für den Schulunterricht zu entwickeln, die an einer Schule evaluiert bzw. für den Schulunterricht diskutiert werden können. Es ist zu überlegen, welche ökologischen Probleme in den Informatikunterricht integriert werden können (Forschungsfrage 2) und welche fachdidaktischen Ansätze geeignet sind (Forschungsfrage 3).

Zunächst wurde ein Modul skizziert, das die grundlegenden Ideen für den Informatikunterricht mit ökologischen Aspekten gesammelt hat [Pe18]. Das Modul „Umweltfreundlicher Umgang mit IT im Alltagsleben“ basiert auf der Idee eines rationalen Umgangs mit IT aus Sicht der Ökologie. Als Beispiele werden Probleme in Bezug auf das Recycling von IT, ihr Energieverbrauch sowie ihr allgemeiner Einfluss auf die Umwelt angeboten. Dabei wurde geplant, den Unterricht mit realen Beispielen aus dem Alltag zu unterstützen. Als Ziel sollten die Lernenden am Ende des Moduls einen umweltbewussteren Umgang mit IT-Mitteln verinnerlichen und die angeeigneten Kenntnisse im Alltag anwenden können. Konkrete Aufgabenbeispiele dienen der

Berechnung des Energieverbrauchs einer IT-Workstation pro Tag. Zudem sollte eine App Thema „Mülltrennung“ (Abb. 2) entworfen werden. Teilnehmer eines Workshops im Jahr 2018 hielten dieses für die Schüler der Sekundarstufe I für wenig motivierend.

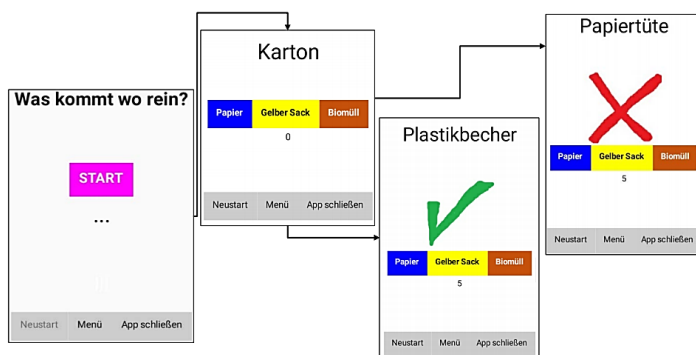


Abb. 2: Prototyp einer Mülltrennungs-App

3 Befragung von Lehrenden

Im Schuljahr 2017/18 wurde im Rahmen einer Umfrage unter Lehrenden ein Meinungsbild eingeholt. An der Umfrage haben 65 Lehrende aus NRW teilgenommen.

3.1 Methodik und Durchführung

Der Fragebogen wurde in EvaSys⁸ generiert und bestand aus 4 quantitativen und 3 qualitativen Fragen (Abb.3). Das Ziel der Umfrage für die weitere Konzept- und Modulentwicklung war, die Expertenmeinungen für die Beantwortung der zweiten Forschungsfrage einzuholen: welche ökologischen Probleme in den Informatikunterricht integriert werden können. Einladungen für die Teilnahme an der Umfrage wurden per Fax an Schulen der Sekundarstufe in NRW geschickt.

⁸ Eine in Münster verfügbare Evaluationssoftware für automatisierte Befragungen

7.1 Welche ökologischen Themen sollten aus Ihrer Sicht im Rahmen des Informatikunterrichtes behandelt werden? (Mehrfachauswahl möglich)

- Optimierte Ressourcennutzung bei der Konstruktion und Verschrottung von digitalen Endgeräten wie Computer, Handy etc.
- Reduzierung des Energieverbrauchs bei der Verwendung von Computern, Handys etc.
- Einsatz von Webformularen und Datenbanken zur Vogelzählung
- Regeln und Steuern von Heizungssystemen
- Andere

7.2 Fallen Ihnen noch weitere ökologische Themen ein, die im Informatikunterricht behandelt werden sollten?

7.3 Wie könnten ökologische Aspekte methodisch in den Informatikunterricht integriert werden? (Mehrfachauswahl möglich)

- Modellierung und Simulation von Umweltprozessen z.B. für Prognosen
- Ausstellung zu in digitalen Endgeräten verwendeten Materialien und deren Gewinnung
- Programmierung von sensor-gestützten Systemen zur Erfassung von Umweltdaten
- Andere

7.4 Fallen Ihnen weitere Methoden zur Integration ein?

7.5 Für wie sinnvoll halten Sie die Integration von ökologischen Aspekten im Informatikunterricht?

unnötig sehr sinnvoll

keine Angabe

7.6 Kennen Sie bereits ähnliche Projekte? Nein Ja

7.7 Sofern Sie bereits von solchen Projekten gehört haben, welche sind dies?

Abb. 3: Aufbau der Fragen zum Projekt ÖKI in dem Fragebogen

3.2 Auswertung und Interpretation

Quantitativ auszuwertende Daten wurden in EvaSys gesammelt und von uns mittels der Statistiksoftware SPSS ausgewertet. Qualitative Daten wurden von uns ohne Software ausgewertet, weil nicht viele Teilnehmer solche Fragen beantwortet haben.

Viele Teilnehmer haben die Idee des Projektes eher skeptisch gesehen: rund ein Drittel von den Befragten fand es nicht sehr sinnvoll, ökologische Aspekte in den Informatikunterricht zu integrieren, ein Drittel fand die Idee gut und ein Drittel hatte eine neutrale Haltung.

Die Lehrenden wurden auch gefragt, welche ökologischen Themen im Rahmen des Informatikunterrichtes behandelt werden sollten und wie ökologische Aspekte methodisch in den Informatikunterricht integriert werden könnten. Aus den angegebenen Themen wurden die folgenden häufiger ausgewählt (Mehrfachauswahl):

- Optimierte Ressourcennutzung bei der Konstruktion und Verschrottung von digitalen Endgeräten wie Computer, Handy etc. (59%)
- Reduzierung des Energieverbrauchs bei der Verwendung von Computern, Handys etc. (52%)
- Regelung und Steuerung von Heizungssystemen (43%)

In einem Freitextfeld haben die Lehrpersonen die angegebene Liste um die folgenden Themen ergänzt (jeweils eine Nennung):

- Papierflut durch Ausdrucken.
- Rechnerarchitektur in Bezug auf Energieverbrauch.
- Softwareoptimierung, um Endgeräte länger nutzen zu können (2 Nennungen).
- Fahrzeugsteuerung zur Verbrauchsreduktion
- Simulationen zur Folgenabschätzung menschlichen Handels
- Ressourcenhunger von Bitcoingenerierung und -transferierung
- Keine konkreten Themenfelder oder Inhalte vorgeben, sondern stattdessen jeweils Saison oder tagesaktuelle Themen ad hoc integrieren.

Alle drei Vorschläge zur Unterrichtsmethodik (s. Abb. 3) wurden von den Befragten ausgewählt:

- Modellierung und Simulation von Umweltprozessen z.B. für Prognosen (55%)
- Programmierung von sensorgestützten Systemen zur Erfassung von Umweltdaten. (55%)
- Ausstellung zu in digitalen Endgeräten verwendeten Materialien und deren Gewinnung. (37%)

Die Auswahl „Andere“ wurden nur von knapp 8% der Befragten angekreuzt, was dahingehend interpretiert werden könnte, dass nur wenige Lehrende eine Vorstellung zu dem Thema entwickelt haben. Diese Annahme wird gestützt durch die Aussage, dass 97% kein ähnliches Projekt kennen und nur 11% sich auch andere ökologische Themen als die vier von uns vorgeschlagenen für eine Behandlung im Informatikunterricht vorstellen können (vgl. [Pe18]).

4 Modulentwicklung

Aktuell arbeiten wir im Projekt ÖKI an drei Modulen. Die ersten Rückmeldungen von Experten wurden Ende März 2019 im Rahmen des Informatiktages NRW⁹ gesammelt. Dort wurden die Module zum ersten Mal präsentiert und beim Workshop durchgearbeitet.

⁹ <https://informatiktag-nrw.de/informatiktag/informatiktag2019> (15.02.2019)

Im folgenden Abschnitt skizzieren wir den Stand der Überlegungen zu den Modulen, wie sie im Workshop präsentiert wurden.

Jedes Modul enthält eine Problembeschreibung zu dem Kontext „Informatik und Ökologie“ und Vorschläge für praktische Aufgaben. Die Empfehlungen zu Bildungsstandards für das Fach Informatik ([GI08], [GI16]) und die IniK-Kriterien werden berücksichtigt.

4.1 Modul: Nachhaltige Nutzung von Smartphones

Die Lernenden bekommen in immer früherem Alter ein Smartphone. Die Eltern bringen ihnen die Nutzung allerdings höchstens oberflächlich bei. Hintergrundwissen zu umweltrelevanten Informationen wird dabei i. d. R. nicht vermittelt. Dies kann im Informatikunterricht als Themengebiet für den Aufbau ökologischer und informatischer Kompetenzen behandelt werden.

In diesem Modul müssen sich die Lernenden zuerst mit theoretischen Aspekten der nachhaltigen Nutzung von Smartphones auseinandersetzen und danach ein Quiz entwickeln, das Fragen zum Thema enthält. Die notwendigen theoretischen Kenntnisse für das Quiz werden im ersten, theoretischen Teil erworben. Denkbar ist auch eine App-Entwicklung o. ä., wenn die Lernenden zuvor entsprechende Fertigkeiten erworben haben (z. B. mit dem „MIT App Inventor“¹⁰), so dass Softwareentwicklung und Ökologie im Informatikunterricht miteinander verbunden werden. Die App könnte dann entsprechende Kenntnisse zur Nachhaltigkeit bei Handynutzung als Auskunft und das Quiz als Spiel (oder Umfrage) enthalten. Im Rahmen der Unterrichtseinheit werden u. a. die folgenden Fragen gestellt: Welche Gefahrstoffe sind in einem Handy versteckt? Wie viel Strom verbraucht ein Handy? Wie kann ich Energie bei der Handynutzung sparen? Was ist ein Fairphone?

4.2 Modul: Smart Farming

Die modernen Informations- und Kommunikationstechnologien werden immer häufiger in der Landwirtschaft zum Steuern, Regeln und Messen von Abläufen [Ba18] eingesetzt, auch verbunden mit ökologischen Zielen. Hierfür steht der Begriff Smart Farming oder Landwirtschaft 4.0. Die Lernenden können die (ökologischen) Auswirkungen der Automatisierung an diesem Beispiel in der Arbeitswelt kennenlernen und bewerten.

In einem Seminar zur Didaktik der Informatik haben wir im Wintersemester 2018/19 einen ersten Entwurf für das Modul erstellt und eine Integration in das Projekt IniK angedacht. Als Einstieg bekommen die Lernenden einen Überblick zum Thema und überlegen sich, was in der Landwirtschaft digitalisiert werden könnte. Es folgen drei weitestgehend

¹⁰ www.appinventor.mit.edu/explore

unabhängig einsetzbare Bausteine: Stall, Feld, Kommunikation. Ökologische Aspekte können sich beispielsweise auf die optimale Versorgung des Viehs unter Nutzung einer Vieh-Datenbank und eine ökologisch-ökonomische Düngung von Feldern beziehen. Im Rahmen der Unterrichtseinheit werden unter anderem folgende Fragen thematisiert: Wie kann man die Landwirtschaft sinnvoll digitalisieren bzw. was ist bereits digitalisierbar? Welche positiven und negativen Auswirkungen sind bei dem Einsatz von Informatiksystemen in der Landwirtschaft zu bedenken?

4.3 Modul: Reduktion des Energieverbrauchs mit Informatiksystemen

Vielen Menschen fehlt das Bewusstsein, dass die tägliche Nutzung von Elektrogeräten einen hohen und ökologisch schädlichen Energieverbrauch verursacht. Im Informatikunterricht kann darauf aufmerksam gemacht werden, wie man allgemein Elektrogeräte stromsparend nutzt (ggf. fächerverbindend mit dem Physikunterricht).

Im Rahmen dieses Moduls wird eine Simulation entwickelt, in der ein Smart-Homesystem die Heizung ausschalten bzw. niedriger stellen kann, wenn die Temperatur im Raum hoch genug ist bzw. wenn niemand daheim ist. Zudem soll das System die Beleuchtung ausschalten können. Das Hauptziel des Unterrichts ist, dass die Lernenden ein solches System modellieren und programmieren – auch im Sinne eines *physical computing*, indem man beispielsweise verschiedene Sensoren programmiert, die auf das Licht bzw. auf die Bewegung reagieren. Messen oder Simulieren des Strom- bzw. Gasverbrauchs können eine ökologische Dimension integrieren uvm. Im Rahmen der Unterrichtseinheit werden unter anderem die folgenden Fragen gestellt: Welche Informatiksysteme können für das Energiesparen verwendet werden? Welche „grünere“¹¹ Methoden zur Stromerzeugung kann es geben und wie können solche Methoden z. B. in den Smarthome-Systemen eingesetzt werden?

5 Fazit und Ausblick

Obwohl bei der ersten Umfrage ein Drittel der befragten Lehrer die Idee einer Integration von ökologischen Aspekten eher ablehnend bewertet haben, sehen wir ein Potenzial zur Bereicherung des Informatikunterrichts. Auch hat ein Workshop im Rahmen des NRW Informatiktags 2019 gezeigt, dass die Lehrpersonen unser Projekt interessant und sinnvoll finden: alle 14 Teilnehmer haben positive Rückmeldungen gegeben und 3 davon sind bereit, die fertigen Module in ihrem Unterricht auszuprobieren und sie danach zu evaluieren.

Ein weiterer Schritt ist nun, die Module weiter zu entwickeln und besonders die dritte und die vierte Forschungsfrage anzugehen, indem wir uns unter Berücksichtigung der GI-

¹¹ Mit „grünere“ Methoden werden alternative, umweltfreundlichere Methoden zur Energieerzeugung bezeichnet (z.B. Solarenergie): http://www.ess.co.at/URBANECOLOGY/Banzer_Froehlich.pdf (17.02.2019)

Empfehlungen zu Bildungsstandards überlegen werden, welche fachdidaktischen Ansätze für die Entwicklung eines Informatikunterrichtes mit diesen Modulen am besten geeignet sind. Parallel muss nach Informatiksystemen recherchiert werden, die als Gegenstände und Inhalte geeignet sind (4. Forschungsfrage).

Die letzte Forschungsfrage – was umfasst ökologisches Denken und warum sollte es in den Informatikunterricht integriert werden – wird das Projekt dauerhaft begleiten und führt immer wieder zu neuen Überlegungen, indem wir neue Literaturquellen durcharbeiten, die Module überarbeiten und auf Fachtagungen und Konferenzen diskutieren.

Literaturverzeichnis

- [Ba18] Bayer, M.: Was Sie über Landwirtschaft 4.0 wissen müssen. Computerwoche.de, veröffentlicht am 07.08.2018, <https://www.computerwoche.de/a/was-sie-ueber-landwirtschaft-4-0-wissen-muessen,3544215>, Stand: 11.02.2019
- [DRC03] The Design-Based Research Collective: Design-Based Research: An Emerging Paradigm for Educational Inquiry, Educational Researcher Vol.32 No.1, S.5-8, 2003
- [Ex19] Experimento - Naturwissenschaft und Technik lebensnah vermitteln, <https://www.siemens-stiftung.org/de/projekte/experimento>, Stand: 11.02.2019
- [GI08] Grundsätze und Standards für die Informatik in der Schule - Bildungsstandards Informatik für die Sekundarstufe I: GI, Gesellschaft für Informatik e.V., Beilage zu LOG IN, 28. Jg. (2008), Heft Nr. 150/151
- [GI16] Grundsätze und Standards für die Informatik in der Schule - Bildungsstandards Informatik für die Sekundarstufe II: GI, Gesellschaft für Informatik e.V., Beilage zu LOG IN, 36. Jg. (2008), Heft Nr. 183/184.
- [In19] Informatik im Kontext - Kriterien für die Auswahl von Kontexten, <https://medienwissenschaft.uni-bayreuth.de/inik/konzepte/kriterienkontexte/>, Stand: 14.02.2019
- [KK08] Koubek, J.; Klumpp S.: Dimensionen der Informatik. 25-Ökologie. Folien zur Vorlesung Informatik & Informationsgesellschaft II., 2008.
- [Pe18] Petrenko, I.: Bildung ökologischen Denkens im Informatikunterricht. In (Thomas, M.; Weigend, M. Hrsg.): Informatik und Medien. 8. Münsteraner Workshop zur Schulinformatik 18. Mai 2018. BoD- Books on Demand, Norderstedt, S. 81-88, 2018.
- [Se19] senseBox, www.sensebox.de, Stand: 14.02.2019
- [Th93] Thomas, M.: Computersimulation als Hilfsmittel zur Erkenntnisgewinnung über das Verhalten dynamischer Wirkungsgefüge, Schriftliche Hausarbeit im Rahmen der 1. Staatsprüfung an der Universität Paderborn 1993
- [Wi19] Willi, Josef: Ökologisch denken heißt Gesamtzusammenhänge erfassen, <http://www.landwirtschaftundleben.at/downloads/lehrbrief2.2.2.pdf>, Stand: 16.02.2019

Praxisberichte

Erfahrungsbericht zur Qualifizierungsmaßnahme Informatik als Erweiterungsfach (Lehramt Realschule) in Bayern

Marc Berges,¹ Matthias Ehmann,² Rainer Gall,³ André Greubel,³ Nicole Günzel-Weinkamm,² Verena Haller,⁴ Martin Hennecke,³ Ute Heuer,⁴ Julia Kronawitter,⁴ Annabel Lindner,¹ Nicolai Pöhner³

Abstract: Teil des Programms BAYERN DIGITAL II ist die berufsbegleitende Nachqualifizierung von rund 180 Lehrerinnen und Lehrern der Realschule. Über die Ermäßigung des Stundendeputats wird den Teilnehmenden ermöglicht, sich innerhalb von zwei Jahren für das Fach Informatik an bayerischen Realschulen nachzuqualifizieren. Ausbildungsziel ist das einem regulären Studium entsprechende Staatsexamen. An mehreren universitären Standorten wurden daher entsprechende Maßnahmen aufgelegt. Durch die besonderen Rahmenbedingungen kommen vor allem Lehr- und Lernformen des „blended learning“ Konzepts zum Tragen. Zweiwöchentliche Präsenztage sorgen dabei für eine kontinuierliche Flankierung, wobei versucht wird, im Sinne des „flipped classroom“ Prinzips die Präsenzzeit möglichst lernerzentriert und interaktiv zu gestalten. Untersuchungen zur Zufriedenheit und Motivation nach dem ersten Halbjahr zeigen den Erfolg der Maßnahme, aber auch weiteres Verbesserungspotential.

Keywords: Nachqualifizierung; Realschule; Blended Learning

1 Einleitung

Seit der Einführung des Pflichtfachs Informatik an bayerischen Gymnasien im Schuljahr 2003/04 setzt das Land neben der Einstellung von Absolventinnen und Absolventen der informatischen Lehramtsstudiengänge auch auf die Nachqualifizierung von im Dienst befindlichen Lehrkräften mit anderen Fächerkombinationen. Durch verschiedene Maßnahmen wie den Programmen NELLI bzw. SIGNAL (2001-06) und FLIEG (seit 2006) wurde so ein großer Teil der heutigen Lehrkräfte nachträglich für das Fach Informatik qualifiziert.

An der bayerischen Realschule wird Informatik gemeinsam mit Textverarbeitung und Technischem Zeichnen / Computer Aided Design im Rahmen des Pflichtfachs „Informatikstechnologie“ angeboten. Textverarbeitung und Technisches Zeichnen / Computer Aided

¹ Universität Erlangen-Nürnberg, Didaktik der Informatik, Martensstraße 3, 91058 Erlangen, marc.berges@fau.de

² Universität Bayreuth, Didaktik der Informatik, Universitätsstraße 30, 95447 Bayreuth, matthias.ehmann@uni-bayreuth.de

³ Universität Würzburg, Didaktik der Informatik, Emil-Fischer-Str. 30, 97074 Würzburg, martin.hennecke@uni-wuerzburg.de

⁴ Universität Passau, Didaktik der Informatik, Innstr. 33, 94032 Passau, ute.heuer@uni-passau.de

Design werden dabei oft von Fachlehrkräften unterrichtet. Entsprechend ist der Anteil der grundständig ausgebildeten Informatiklehrkräfte an der Realschule eher gering. Zwar gab es auch für die Realschule in der Vergangenheit diverse Fortbildungsangebote, Nachqualifizierungen, die wie SIGNAL oder FLIEG einen universitären Abschluss erreichen sollten, wurden jedoch bisher keine angeboten.

Das Programm BAYERN DIGITAL II ermöglicht seit 2018 nun auch die Nachqualifizierung von knapp 180 Lehrkräften bayerischer Realschulen. Diese erhalten für zwei Jahre eine Freistellung von einem Tag pro Woche. Ferner stehen Mittel für Reisekosten für Präsenzveranstaltungen an den durchführenden Universitäten zur Verfügung. Am Ende der Maßnahme sollen die Teilnehmerinnen und Teilnehmer die Erweiterungsprüfung für das Fach Informatik ablegen. In Bayern bedeutet dies die Teilnahme am Staatsexamen. Das Land verspricht sich von der Maßnahme neben der Verbesserung der Qualität des informatischen Unterrichts auch die Schaffung einer Grundlage für den weiteren Ausbau des Schulfachs an der Realschule. Im Rahmen des Beitrags soll der Aufbau und die Durchführung der Maßnahme kurz vorgestellt werden. Ferner wird über Erfahrungen aus dem ersten Halbjahr der Nachqualifizierung berichtet.

2 Maßnahme

2.1 Informatik an der bayerischen Realschule

Die Schulordnung für die bayerische Realschule von 2002 sah je nach gewählter Ausbildungsrichtung Informatik als Pflicht- bzw. Wahlpflichtfach vor. Seit 2007 wurde es in allen Ausbildungsrichtungen durch das Pflichtfach Informationstechnologie ersetzt. Inhaltlich umfasst das Fach die zuvor eigenständigen Bereiche Informatik, Textverarbeitung und Technisches Zeichen / Computer Aided Design. Dem Fach liegt ein modularisierter Lehrplan⁵ mit flexibilisierter Stundentafel zugrunde. Dieser legt für die Jahrgangsstufen 5 bis 7 (Anfangsunterricht) acht Module zu je 14 Unterrichtsstunden aus den Bereichen Tastschreiben / Textverarbeitung, Informationstechnische Grundlagen und Informatik fest. Ab der achten Jahrgangsstufe (Aufbauunterricht) sind Umfang und Inhalte je nach Ausbildungsrichtung unterschiedlich. Der Modulkanon umfasst derzeit die Bereiche Textverarbeitung, Tabellenkalkulation, Datenbanksysteme, Computergestützte Konstruktion, Datennetze, Programmierung - Algorithmen und Objekte, Logik und Robotik sowie Multimedia. Die Modulblöcke bestehen dabei aus zwei bis sechs Modulen à 14 Unterrichtsstunden.

Die Lehrerschaft ist im Fach Informationstechnologie sehr heterogen zusammengesetzt. Neben Lehrerinnen und Lehrern, die das Fach Informatik als Lehramtsfach an der Universität studiert haben, gibt es im Rahmen des Vorbereitungsdienstes nachqualifizierte Lehrkräfte. Hinzu kommen viele an Staatsinstituten ausgebildete Fachlehrerinnen und Fachlehrer. Da

⁵ <https://www.lehrplanplus.bayern.de/fachlehrplan/realschule/10/it>

die drei Qualifizierungssysteme inhaltlich und organisatorisch sehr unterschiedlich sind, ergeben sich unterschiedliche Schwerpunkte der Lehrkräfte in den drei fachlichen Bereichen.

Die Regelstudienzeit für Lehramt Informatik an Realschulen beträgt in Bayern sieben Semester. Rund ein Drittel des Studiums entfällt dabei auf Veranstaltungen aus dem Fach Informatik bzw. deren Didaktik. Am Ende des Studiums steht in Bayern nach wie vor das Staatsexamen als zentrale schriftliche Prüfung. Zwar haben die bayerischen Universitäten ihr Studium modularisiert und bieten meist neben dem Staatsexamen auch Bachelorabschlüsse an, für die Aufnahme in das Referendariat qualifiziert jedoch nur die staatliche Prüfung. Das Studium stellt somit vor allem eine Vorleistung für das Staatsexamen dar, geht aber auch in die Berechnung der Gesamtnote mit ein. Für das Staatsexamen Informatik schreibt die Lehramtsprüfungsordnung I (LPO I)⁶ drei schriftliche Prüfungen vor. Geprüft werden die Teilgebiete *Theoretische Informatik und Algorithmen und Datenstrukturen*, *Softwaretechnologie und Datenbanksysteme* sowie *Fachdidaktik Informatik*.

2.2 Herausforderungen

Die Durchführung der Maßnahme ist mit einer Reihe von Herausforderungen verbunden. Der enge Zeitplan von zwei Jahren, in dem die Maßnahme durchgeführt wird, stellt an die Studierenden hohe Anforderungen. Trotz der Reduktion der Unterrichtsverpflichtung wird von den Teilnehmerinnen und Teilnehmern gefordert, sich in relativ kurzer Zeit eine sehr große Menge komplexer informatischer Fachinhalte anzueignen. Dies ist nur mit Disziplin und durch kontinuierliches Arbeiten mit den Studieninhalten möglich. Die Lehrkräfte müssen eine Balance zwischen Schule, begleitendem Studium und Privatleben finden, die über einen Zeitraum von zwei Jahren tragbar ist und gleichzeitig ein möglichst rasches Vorankommen im Studium ermöglicht. Um dies zu unterstützen, wird den Teilnehmerinnen und Teilnehmern im Sinne von „blended learning“ [OT05] ein umfangreiches Studienmaterial zum Selbststudium zur Verfügung gestellt sowie an den Präsenztagen versucht, komplexe Sachverhalte möglichst zugänglich darzustellen.

Auf Seiten der Universitäten ist sowohl der straffe Zeitplan als auch die Heterogenität in Bezug auf die fachlichen Vorkenntnisse bei den Teilnehmerinnen und Teilnehmern eine große Herausforderung. So finden sich in der Gruppe verschiedenste Studienfächer im Erststudium, unterschiedliche Erfahrungen in Bezug auf IT / Informatik-Unterricht, sowie unterschiedlicher Umgang der Lehrkräfte mit der Doppelbelastung, z. B. in Bezug auf Frustrationstoleranz. Um den Studierenden hier möglichst gut zur Seite zu stehen, fungieren studentische Hilfskräfte an den Universitäten als unterstützende Tutorinnen und Tutoren.

⁶ http://www.gesetze-bayern.de/Content/Document/BayLPO_I-49

2.3 Aufbau der Maßnahme

Um innerhalb des engen Zeitrahmens von nur zwei Jahren eine fundierte Ausbildung der Teilnehmerinnen und Teilnehmer sowohl in fachlicher Hinsicht als auch in Hinblick auf die Unterrichtspraxis zu realisieren, folgt die Maßnahme einerseits einem relativ engen Curriculum und greift andererseits auf Blended Learning [OT05] und Flipped Classroom [BS12] Konzepte zurück. Dies ist notwendig und sinnvoll, da aufgrund des berufsbegleitenden Studiums nur begrenzt die Möglichkeit besteht, an Präsenzveranstaltungen an der Universität teilzunehmen. Da vorangehende Weiterbildungsprogramme jedoch gezeigt haben, dass die Teilnehmerinnen und Teilnehmer Präsenzzeiten für sehr wichtig halten und diese Auswirkungen auf den Erfolg der Maßnahme haben (vgl. [Sp09]), finden im Zweiwochenrhythmus Präsenztage an den beteiligten Universitäten statt, an denen die Inhalte aufgearbeitet und vertieft werden. Ziel ist hierbei kein Vorlesungsformat, vielmehr soll die Präsenzzeit möglichst interaktiv gestaltet werden und intensiv auf Fragen und Probleme der Studierenden eingegangen werden.

Der inhaltliche Aufbau der Maßnahme orientiert sich am Kerncurriculum für das bayerische Staatsexamen in Informatik⁷, das zentrale Fachkenntnisse aus den Bereichen Datenbanksysteme, Algorithmen und Datenstrukturen, Softwaretechnik und Theoretische Informatik sowie der Fachdidaktik Informatik als examensrelevant ausweist. Da die Fakultas in Informatik ausgewiesenes Ziel der Maßnahme ist, ist die Abdeckung dieser zentralen Themengebiete aus fachwissenschaftlicher Sicht erforderlich, zudem soll jedoch auch ein Wissensfundament aufgebaut werden, das die Lehrkräfte in der täglichen Unterrichtspraxis unterstützt. Vor diesem Hintergrund wurde ein Modell mit acht Modulen entwickelt (vgl. Tab. 1), bei dem zentrale Ideen wie die Algorithmik semesterübergreifend aufgegriffen werden und die Programmierung als wachsender Baustein vom ersten Semester bis zum Examen umfassend vermittelt und geübt wird. Zudem werden ein Softwarepraktikum, welches für die Studierenden im Erweiterungsfach die einzige Zulassungsvoraussetzung zum Staatsexamen darstellt, und ein Examens-Repetitorium in den Modulplan integriert. Für den Bereich Fachdidaktik ist kein eigenes Modul vorgesehen, vielmehr werden fachdidaktische Elemente und Werkzeuge in die fachpraktischen Module integriert. Am Ende der Module steht jeweils eine Klausur zur Überprüfung des Lernfortschrittes und als individuelle Rückmeldung für die Teilnehmerinnen und Teilnehmer. Diese Klausuren haben jedoch keinen Einfluss auf die Examenszulassung. Um einen Eindruck von der Gestaltung der Module zu geben, werden im Folgenden die Module des ersten Semesters detaillierter vorgestellt.

2.3.1 Modul Datenbanksysteme

Grundständige Studierende erwerben im Bereich Datenbanksysteme und Softwaretechnologie mindestens 15 ECTS (vgl. LPO I⁸). Dies wird im Rahmen der Maßnahme durch die

⁷ <https://www.verkuendung-bayern.de/amtsblatt/dokument/kwmb1-2009-2-34/>

⁸ http://www.gesetze-bayern.de/Content/Document/BayLPO_I-49

1. Semester	2. Semester	3. Semester	4. Semester
Datenbanksysteme	Algorithmen und Datenstrukturen	Softwaretechnologie	Theoretische Informatik
Fachdidaktik			
Einführung in die Programmierung	Programmierung	Softwarepraktikum	Repetitorium

Tab. 1: Modulplan der Maßnahme

Behandlung der im Kerncurriculum geforderten Inhalte abgebildet: „Datenmodellierung und Datenbankentwurf; relationales Modell (Grundlagen); Konvertierung eines ER-Entwurfs in einen relationalen Entwurf; Anfragesprachen in DBMS (SQL); Integrität (Strukturelle und Domänenspezifische Integritätsbedingungen); relationale Entwurfstheorie (Funktionale Abhängigkeiten, Normalformen); Transaktionsmanagement“. Diese Auflistung bildet den inhaltlichen Rahmen für das erste Modul, da die zeitlichen Restriktionen keine weitere Vertiefung zulassen.

Der grundsätzliche Aufbau des Moduls entspricht dem didaktischen Konzept des „Flipped Classroom“. Dabei werden den Lernenden entsprechende Lerninhalte in einer Selbststudiumsphase zur Verfügung gestellt. Als Literatur zum Eigenstudium wird in diesem Modul das Buch „Datenbanksysteme - Eine Einführung“ von Alfons Kemper und André Eickler [KE15] empfohlen. Zusätzlich wird auf zum Buch passende Begleitmaterialien, wie Präsentationen und Videoaufzeichnungen⁹ verwiesen. Zu den jeweiligen Kapiteln lösen die Teilnehmerinnen und Teilnehmer begleitende Aufgabenblätter auf Staatsexamensniveau. In der anschließenden Präsenzveranstaltungen werden Übungsblätter und im Selbststudium aufgekommene Fragen besprochen, um dann anhand weiterer Übungsaufgaben und Präsentationen zu den theoretischen Inhalten das aktuelle Thema zu vertiefen und im gemeinsamen Austausch weitergehende Fragen und Probleme zu klären. Abschließend wird ein kurzer Ausblick auf die Inhalte des nächsten Themenbereichs gegeben sowie mögliche Schwierigkeiten des Themas vorweggenommen. In der folgenden Selbststudiumsphase wird den Teilnehmenden noch einmal die Möglichkeit gegeben anhand von aufbauenden Übungsaufgaben die Inhalte zu vertiefen. Gleichzeitig beginnt der nächste Zyklus mit dem Einarbeiten in die theoretischen Inhalte und dem Lösen der ersten Übungsaufgaben.

Ergänzend zu den fachlichen Inhalten werden folgende beispielhafte fachdidaktische Inhalte im ersten Modul thematisiert: Allgemeinbildungsanspruch, Fundamentale Ideen, Lehrplan der Realschule Bayern: Module Datenbanksysteme I & II, Werkzeuge zur Modellierung und zum Erstellen von SQL-Abfragen und Aufgaben aus Schulbüchern und Aufgabensammlungen.

⁹ <https://db.in.tum.de/teaching/bookDBMSeinf/aufzeichnungen/>

2.3.2 Modul Einführung in die Programmierung

Während das erste Modul an allen Standorten weitestgehend gleich abgelaufen ist, wurde das Modul „Einführung in die Programmierung“ an den verschiedenen Standorten nach den lokalen Gegebenheiten unterschiedlich gehandhabt. Hier wird das Vorgehen der Universität Passau dargestellt.

Die Notwendigkeit, das Testen von Programmen systematisch in die Programmierveranstaltung des ersten Informatiksemesters zu integrieren, wird seit vielen Jahren gesehen (vgl. [BM16]). Im Bereich der Fort- und Weiterbildung von Lehrkräften wurden Überlegungen zum Unterrichten des Testens im Sinne einiger Ideen des „test-driven learnings“ jedoch bisher wenig beschrieben.

Weiter gibt es nur vereinzelt Berichte, die entsprechend auf das systematische Testen im Rahmen einer Einführung in die Programmierung in der Schule fokussieren (vgl. [SS13]), obwohl dies folgende positive Effekte haben kann: (1) Schülerinnen und Schüler erfahren, welche wichtige Rolle das Testen schon während und nicht erst am Ende der Entwicklung einer robusten und zuverlässigen Lösung spielt. (2) Sie erkennen, welche Vorteile eine Automatisierung in diesem Zusammenhang bietet. (3) Sie erhalten durch das Ausführen der Tests eine vergleichsweise schnelle und an ihr persönliches Arbeitstempo angepasste Rückmeldung über Aspekte ihres Arbeitsfortschritts. (4) Die lernbegleitende Prüfung einer Schülerlösung durch automatische Tests im Sinne eines inkrementellen „ein bisschen schreiben, testen, ein bisschen mehr schreiben, wieder testen usw.“ - Zugangs kann Lehrkräfte entlasten. So haben Lehrende mehr Zeit für die individuelle Betreuung und Unterstützung ihrer Schüler.

Entsprechend dieser Überlegungen wird ein Vorgehen gewählt, das einerseits konsequent und von Anfang an mit Aufgaben und zugehörigen vordefinierten Tests arbeitet, andererseits einige ausgewählte einfache vordefinierte Tests reflektiert. Dabei werden den Lehrkräften die vordefinierten Tests auch als „worked examples“ zur Verfügung gestellt.

So setzen sich die Lehrkräfte bereits in einer sehr frühen Phase ihres Lernprozesses mit dem regelmäßigen Testen ihres Codes auseinander. Dadurch entwickeln sie ein Bewusstsein für die Notwendigkeit der systematischen Programmentwicklung. Sie beschäftigen sich nicht nur mit ihren eigenen möglicherweise verschieden diffusen Erwartungen an den Code, sondern auch mit der durch die Tests sichergestellten Funktionalität. Die Studierenden arbeiten mit Java und entsprechenden JUnit-Tests, die in Gradle-Projekte¹⁰ integriert sind. Gradle wird als Projektmodell verwendet, um die Ausführung der Tests und den Projektimport in die Entwicklungsumgebungen möglichst einfach zu gestalten. Als Entwicklungsumgebung dient IntelliJ, wobei geplant ist BlueJ mit einzubeziehen.

Thematisch fokussiert das Modul die folgenden Konzepte: *Klasse, Objekt, Kontrollstrukturen, einfache Datenstrukturen, Datenkapselung, Exceptions, Polymorphie* und *Programmierkon-*

¹⁰ <https://gradle.org/>

ventionen. Der methodische Aufbau des Moduls deckt sich mit dem des Themenbereichs Datenbanken.

2.4 Teilnehmerinnen und Teilnehmer der Maßnahme

Die Gewinnung von Interessenten erfolgte seitens des Kultusministeriums über entsprechende Schreiben an die Schulleitungen. Die vom Ministerium ausgewählten ca. 80 Teilnehmerinnen und Teilnehmer der ersten Kohorte wurden im Anschluss gleichmäßig auf die Universitäten Bayreuth, Erlangen-Nürnberg, Passau und Würzburg verteilt und sind dort als reguläre Studierende in Erweiterungsstudiengängen für das Lehramt an Realschulen eingeschrieben. Die gleichmäßige Verteilung auf die Standorte führt bei einzelnen Teilnehmenden zu sehr langen Anfahrtswegen zu den Präsenzveranstaltungen.

Tabelle 2 zeigt relevante (berufs-)biographische Daten der Teilnehmerinnen und Teilnehmer. Die Hälfte von ihnen hat bereits erste Unterrichtserfahrung im Fach Informatik / Informationstechnologie gesammelt. Die unterrichtliche Tätigkeit erfolgte meist im Rahmen von fachfremdem Unterricht bzw. aufgrund einer Lehrerlaubnis.

Merkmal	Anzahl	Merkmal	Anzahl
Geschlecht ($n = 80$)		Unterrichtsfahrung ($n = 76$)	
weiblich	38	1-5 Jahre	15
männlich	42	6-10 Jahre	32
		mehr als 10 Jahre	29
Unterrichtsfächer ($n = 80$)		Unterrichtserfahrung Informatik/IT ($n = 76$)	
Mathematik oder Physik	42	keine	26
Biologie oder Chemie	8	1-5 Jahre	17
andere	30	6-10 Jahre	12
		mehr als 10 Jahre	11

Tab. 2: (Berufs-)biographische Merkmale der Teilnehmerinnen und Teilnehmer

3 Erste Evaluierungsergebnisse

Kurz vor Ende des ersten Halbjahres wurden die Teilnehmerinnen und Teilnehmer mit einer Online-Erhebung um ein erstes Feedback zu ihren bisherigen Erfahrungen mit der Maßnahme gebeten. Zudem wurden Selbstwirksamkeitsaspekte und Motivation untersucht.

Um die allgemeine Zufriedenheit besser einordnen zu können, haben wir zunächst nach dem empfundenen Kursniveau gefragt (Wie haben Sie die Gesamtschwierigkeit des ersten Kurshalbjahres empfunden?). Auf einer 5-stufigen Likert-Skala geben drei Teilnehmende an, das Kursniveau sei viel zu hoch, 29 empfinden es als zu hoch und 29 als angemessen. Die restlichen Teilnehmerinnen und Teilnehmern der Maßnahme haben kein Feedback gegeben.

Abb. 1 zeigt ein erstes Bild der Zufriedenheit der Teilnehmerinnen und Teilnehmer mit den einzelnen Komponenten des Kursdesigns, wie Lesestoff in Form des eingesetzten Fachbuchs, die Durchführung der Präsenzveranstaltungen, die für den Kurs erstellten Aufgabenblätter, die Auswahl der Themenbereiche und alle sonstigen Materialien wie z.B. Vorlesungsmitschnitte (Wie zufrieden waren Sie mit den folgenden Aspekten des Kurses?). Antworten konnten auf einer 4-stufigen Likert-Skala von gar nicht zufrieden (1) über nicht zufrieden (2) und zufrieden (3) bis zu sehr zufrieden (4) gegeben werden.

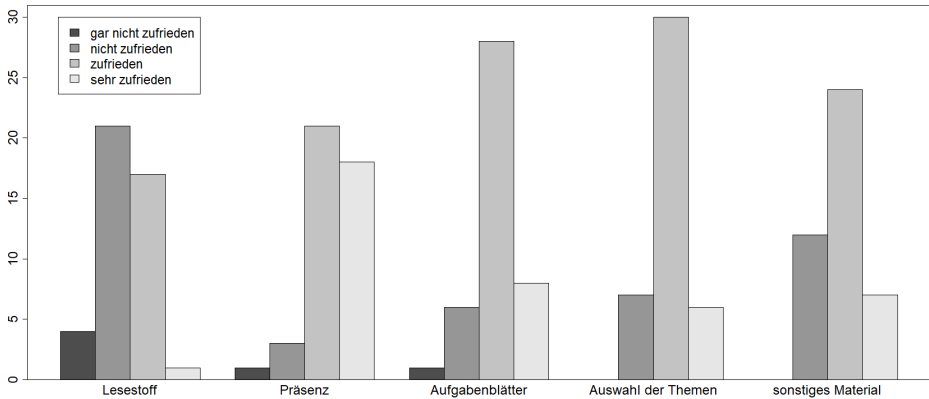


Abb. 1: Einschätzung der Zufriedenheit mit den einzelnen Komponenten des Kursdesigns

Neben der Zufriedenheit mit den einzelnen Komponenten des Kursdesigns haben wir auch nach der Zufriedenheit mit dem Lernfortschritt gefragt. Basierend auf dem Instrument von Michalis Giannakos [GHC13] haben wir folgende Aussagen auf einer 4-stufigen Likert-Skala bezüglich des Grades der Zustimmung bewerten lassen:

- Ich kann einen Überblick über die Konzepte des ersten Halbjahres geben.
- Ich habe die Konzepte des ersten Halbjahres verstanden.
- Ich habe im ersten Halbjahr viel gelernt.
- Ich nehme aus dem ersten Halbjahr ein erweitertes bzw. vertieftes Verständnis der Konzepte mit.

Die Mehrheit der Teilnehmerinnen und Teilnehmer ist mit dem Lernfortschritt zufrieden. Nur sechs der Teilnehmenden zeigen sich mit Ihrem Lernfortschritt unzufrieden.

Zur Erfassung der Motivation wurde auf die kurze Version des „Intrinsic Motivation Inventory“¹¹ von Deci & Ryan zurückgegriffen. Dieses Messinstrument basiert auf der Self-Regulation Theory [DR12]. Abb. 2 gibt einen Überblick über die Motivation bezüglich der vier Faktoren „Interesse/Vergnügen“, „wahrgenommene Kompetenz“, „wahrgenommene

¹¹ <https://selfdeterminationtheory.org/>

Wahlfreiheit“ und „Druck/Anspannung“ die mittels Faktorenanalyse auch in unseren Daten bestätigt werden konnten. Dabei spiegelt „Interesse/Vergnügen“ direkt die intrinsische Motivation wieder, die „wahrgenommene Kompetenz“ reflektiert das Kompetenzbedürfnis und die „wahrgenommene Wahlfreiheit“ das Autonomiebedürfnis. Sind die letzten beiden nicht erfüllt, stellt sich ein Gefühl von Druck und Anspannung ein, welches als negativer Prädiktor für die intrinsische Motivation gilt.

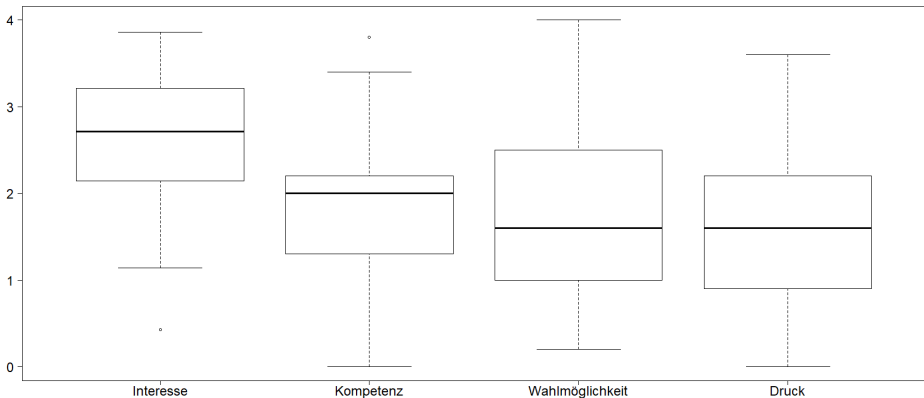


Abb. 2: Verteilung der Teilnehmenden bezüglich der Wahrnehmung (trifft überhaupt nicht zu (0), trifft eher nicht zu (1), teils teils (2), trifft eher zu (3), trifft voll und ganz zu (4)) der Motivationsfaktoren

4 Zusammenfassung und Ausblick

Insgesamt können wir für die Maßnahme nach dem ersten Halbjahr ein positives Fazit ziehen, haben aber im Rahmen der Evaluation einige Elemente gefunden, die in den nächsten Halbjahren noch verbessert werden müssen. So ist die Zufriedenheit mit dem Kursdesign insgesamt relativ hoch. Allerdings bezieht sich das vor allem auf die mit Präsenz assoziierten Elemente (Präsenzveranstaltung, Themenauswahl und Aufgabenblätter). Die Elemente des „flipped-classroom“ wie Lesestoff und sonstiges zum Selbststudium bereitgestelltes Material schneiden eher schlechter ab. Für die kommenden Kursmodule bedeutet das, dass weitere Anstrengungen in die Aufbereitung von Lehrmaterialien gelegt werden müssen. Die Präsenz lässt sich aufgrund der ohnehin als hoch bis sehr hoch empfundenen Arbeitsbelastung und der mitunter weiten Anfahrtswege nicht steigern. Trotz der hohen Arbeitsbelastung gibt es bisher von 80 Teilnehmenden nur vier AbbrecherInnen zu verzeichnen, was verglichen mit üblichen Abbruchraten nach dem ersten Semester eine ordentliche Quote darstellt.

Die Motivationsverteilung zeigt einige interessante Aspekte, die in die Planung und Durchführung der folgenden Module eingehen wird. So liegt das Interesse und damit die intrinsische Motivation weiterhin im „positiven“ Bereich. Allerdings ist die empfundene Kompetenz eher als „neutral“ zu betrachten mit einer leichten Tendenz zum „negativen“

Empfinden. Dies kann aber auch am Zeitpunkt der Erhebung vor der ersten Klausur liegen. Die fehlende Autonomie, die ein wenig in der Natur der Maßnahme liegt, ist ein Aspekt dem in den kommenden Semestern unbedingt begegnet werden muss, um ein weiteres Absinken der Motivation zu verhindern. Ebenso muss der empfundene Druck, der sicherlich mit dem Fortschreiten der Maßnahme eher zunehmen wird, im Auge behalten werden.

Neben den weiteren Semestern haben die gewonnenen Erkenntnisse auch unmittelbare Auswirkungen auf die zweite Kohorte der Realschullehrkräfte, die die Maßnahme im Schuljahr 2019/2020 beginnen werden. Außerdem soll der sich aus der Einführung von verpflichtendem Informatikunterricht in der 11. Jahrgangsstufe für alle Ausbildungsrichtungen der bayerischen Gymnasien ergebende massive Bedarf an neuen Informatiklehrkräften ebenfalls durch eine Nachqualifizierung gedeckt werden. Ziel ist es dabei, dass an jedem der rund 400 bayerischen Gymnasien eine zusätzliche Informatiklehrkraft zur Verfügung steht. Die Maßnahme wird zum Schuljahr 2019/2020 anlaufen und auf den Erkenntnissen der hier beschriebenen Maßnahme aufbauen.

Literaturverzeichnis

- [BM16] Braught, Grant; Midkiff, James: Tool Design and Student Testing Behavior in an Introductory Java Course. In: Proceedings of the 47th ACM Technical Symposium on Computing Science Education. SIGCSE '16, ACM, New York, NY, USA, S. 449–454, 2016.
- [BS12] Bergmann, Jonathan; Sams, Aaron: Flip your classroom: Reach every student in every class every day. ISTE and ASCD, Eugene, Or. and Alexandria, Va., 1. ed.. Auflage, 2012.
- [DR12] Deci, Edward L.; Ryan, Richard M.: Motivation, Personality, and Development Within Embedded Social Contexts: An Overview of Self-Determination Theory. In: The Oxford handbook of human motivation, S. 85–107. Oxford University Press, New York, 2012.
- [GHC13] Giannakos, Michail; Hubwieser, Peter; Chrisochoides, Nikos: How students estimate the effects of ICT and programming courses. In: Proceedings of the 44th ACM technical symposium on Computer science education. ACM Press, New York, S. 717, 2013.
- [KE15] Kemper, Alfons; Eickler, André: Datenbanksysteme: Eine Einführung. De Gruyter Oldenbourg, Berlin and Boston, 10. auflage. Auflage, 2015.
- [OT05] Oliver, Martin; Trigwell, Keith: Can 'Blended Learning' Be Redeemed? E-Learning and Digital Media, 2(1):17–26, 2005.
- [Sp09] Spohrer, Matthias: Konzeption und Analyse neuer Maßnahmen in der Fort- und Weiterbildung von Informatiklehrkräften. 2009.
- [SS13] Stejskal, R.; Siy, H.: Test-driven learning in high school computer science. In: 2013 26th International Conference on Software Engineering Education and Training (CSEE T). S. 289–293, May 2013.

„Bildung in der digital vernetzten Welt“ Ohne Informatik nicht denkbar!?

Folgerungen aus mehr als 50 Jahren 'Digitalisierung' der Bildung

Dieter Engbring¹

Abstract: In diesem Aufsatz wird diskutiert, ob bzw. in welchem Umfang „Bildung in der digital vernetzten Welt“ informatische Bildung beinhalten sollte. Dazu wird aufgezeigt, dass die Diskussion um eine solche Bildung nicht nur seit vielen Jahrzehnten geführt wird, sondern dass seit etwa 30 Jahren auch die entsprechenden Herausforderungen benannt sind, ohne dass diese gelöst werden konnten. Zur empirischen Erforschung dieser Herausforderungen wird ein Alternativvorschlag unterbreitet, der die verschiedenen Perspektiven miteinander verbindet.

Keywords: Digitalisierung, Allgemeinbildung, Medienbildung, Technikgenese

1 Einleitung

Der Prozess der 'Digitalisierung' wird nicht nur durch neue Hardware schulische Bildung verändern, auch Ziele und Inhalte werden sich wandeln müssen. Die Strategie der Kultusministerkonferenz für eine „Bildung in der digitalen Welt [KMK17] und der darauf Bezug nehmende Medienkompetenzrahmen NRW² sind Zeugen dafür. „Bildung in der digitalen vernetzten Welt (kurz: Digitale Bildung)“ ist die Bezeichnung für diesen Transformationsprozess, die in der »Dagstuhl-Erklärung« [GI16b] genutzt wird. In der Erklärung wird beschrieben, dass die Informatik daran Anteil haben sollte, ohne dass erneut die Forderung nach einem Pflichtfach Informatik erhoben wird.³ Mithin stellt sich die Frage, wie Informatik einen Beitrag zur Bildung in der digitalen vernetzten Welt“ leistet. Dieses zu untersuchen bietet die bildungspolitisch ausgerichtete Erklärung insofern einen Ansatzpunkt, da hierin (nicht zum ersten Mal) der Vorschlag unterbreitet wird, dass es drei Perspektiven auf eine solche Bildung gibt und dass diese aufeinander zu beziehen sind. Dieses ist jedoch nie wirklich gelungen. Möglichen Gründen dafür wird in diesem Aufsatz in Abschnitt 3 nachgegangen und zur weiteren Erforschung dieser Gründe in Abschnitt 4 ein Alternativvorschlag unterbreitet. Dieser Einleitung folgt zunächst ein Abschnitt, in dem die Ähnlichkeit zu früheren Diskussionen aufgezeigt wird. Mit einem 'Fazit mit Ausblick' werden die nun anstehenden Forschungsfragen benannt.

¹ Universität Bonn, Institut für Informatik, Endenicher Allee 19a, 53115 Bonn, dieter.engbring@uni-bonn.de

² <https://medienkompetenzrahmen.nrw.de/>

³ Dies ist mindestens ungewöhnlich, da diese Forderung nach einem Pflichtfach, ansonsten Bestandteil vieler solcher »Dagstuhl-Erklärungen« gewesen ist, zuletzt im Jahr 2015. <https://gi.de/meldung/informatikunterricht-gehört-in-die-schule/>. Dort als 3. »Dagstuhl-Erklärung« ausgewiesen.

2 Über 50 Jahre Digitalisierung des Bildungswesens

B. Koerber und J. Müller haben im Editorial der LOG IN 187/188 daran erinnert, dass schon sehr viele Jahre – wenn auch unter anderen Überschriften – von *Digitalisierung* der Bildung die Rede ist [KM18]. In den Debatten enthalten waren immer zwei Aspekte, die bildungspolitisch bis heute nicht immer getrennt werden. Einerseits geht/ging es um den Einsatz computergestützter Systeme zur Verbesserung (?) des Unterrichts. Andererseits sollen computergestützter Technologien im Unterricht thematisiert werden. Dieser Aufsatz kapriziert sich auf den zweiten Aspekt und zählt hierzu auch die Diskussionen um die Notwendigkeit informatischer Bildung.

Bereits vor etwa 50 Jahren gab es auch in Deutschland erste Pilotprojekte zu einem Informatikunterricht [KM15]. Bald nach ihrer Etablierung als Disziplin an den Universitäten, wurde Informatik 1972 zum Schulfach, da es in den Katalog der Wahlpflichtfächer der gymnasialen Oberstufe aufgenommen wurde. Seither stockt die Entwicklung jedoch. Zwar ist Informatik auch zu einem Wahlpflichtfach in der Sekundarstufe I geworden; Pflichtfach ist es jedoch nur punktuell.⁴ Anstelle der Einrichtung eines Pflichtfaches wurde seit Beginn der 1980er Jahre immer wieder versucht, Computer als Teil sog. fachübergreifenden Lernbereichen zu thematisieren. So wurde 1984 zunächst ein Rahmen- und dann 1987 ein Gesamtkonzept zur „informations- und kommunikationstechnischen Grundbildung“ (je nach Bundesland ITG oder IKG abgekürzt) vorgestellt. Dieses Gesamtkonzept enthält als bildungspolitischer Kompromiss neun Aufgaben [BLK87], die sich allerdings wenig kohärent auf Anwendungsschulung, kritische Reflexion der technologischen Entwicklung, ein wenig Hardwarekunde und Algorithmik beziehen. Die mit diesen Aufgaben verknüpften Inhalte und Ziele lassen sich – das ist letztlich das Ergebnis der nordrhein-westfälischen Interpretation – auch in einem Dreieck anordnen [ITG91]. Die Gegenüberstellung des »Dagstuhl-Dreieck« (links) mit dem Dreieck aus dem NRW-ITG-Lehrplan in der nachstehenden Abbildung zeigt die Ähnlichkeit.

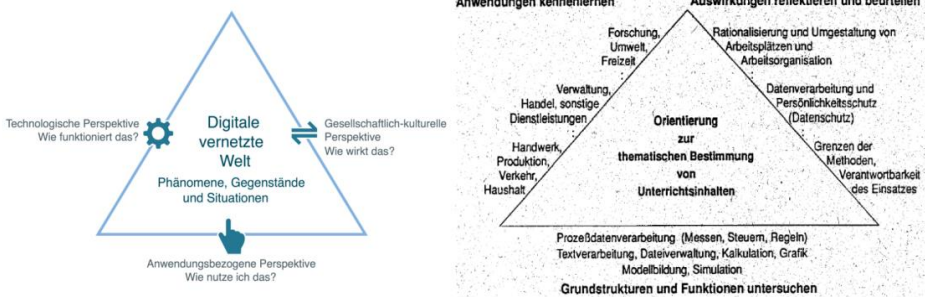


Abb. 1: Gegenüberstellung »Dagstuhl-Dreieck« und aus Auszug aus dem NRW-Lehrplan zur Umsetzung der Informationstechnischen Grundbildung

⁴ In einer Online Petition für ein Pflichtfach Informatik in Nordrhein-Westfalen (NRW) werden fünf Bundesländer aufgezählt, in denen es etabliert ist oder kurzfristig erfolgen soll. www.informatiknrw.de

Für jede der drei Perspektiven wird in dem älteren Dreieck eine Vielzahl von Beispielen genannt, an der sich die notwendige didaktische Analyse mit nachfolgender Aufbereitung Nahrung fand. Es sind damals eine Vielzahl zeitgemäßer Thematisierungen vorgelegt worden. Diese standen jedoch für sich allein, waren nicht exemplarisch, ohne echte Zusammenhänge, da mal dieses und mal jenes Fach zuständig war und nur auf den aktuellen Stand der Technik ausgerichtet. Ein auf die absehbare Zukunft gerichtetes Lernen war kaum möglich, da eine übergreifende (= informatische?) Perspektive fehlte.

Solche fachübergreifenden Lernbereiche erscheinen vielen bildungspolitischen Verantwortlichen auch heute noch ausreichend zu sein, den Themen 'Computer nutzen und einschätzen' gerecht zu werden. Dabei spielt einerseits das Wissen darum, dass es kaum ausgebildete Informatik-Lehrer gibt, sicher ebenso eine Rolle wie eine gewisse Ignoranz (in der ganzen Bandbreite von Unwissenheit bis Missachtung) in Bezug auf die Informatik, die vermeintlich nur ein Fach für *Freaks* und *Nerds* ist. Andererseits liegt die immer wieder aufkommende Forderung nach einem Pflichtfach zwar in der Logik eines Bildungssystems, das die Inhalte dessen, was als allgemeinbildend angesehen wird, weitestgehend über Pflichtfächer definiert. Dabei werden jedoch zugleich die Performanzprobleme übersehen, die Pflichtfächer haben. Vor allem Mathematik und Naturwissenschaften, an denen sich die Informatik immer wieder orientiert hat, haben solche Probleme, die so ähnlich auch auf die Informatik zutreffen werden (vgl. hierzu [En14]).

Eine Ursache könnte darin liegen, dass lebenspraktische und wissenschaftspropädeutische Zielsetzungen nicht ohne weiteres im selben Bildungsangebot erreicht werden können. Heymann hat vor über 20 Jahren für die Mathematik auf der Grundlage seiner Aufgaben allgemeiner Bildung einen Vorschlag unterbreitet, der ein drei Stufen umfassendes Kurssystem vorsieht. Auf der ersten Stufe sollen nur die lebensnotwendigen Inhalte vermittelt werden. Auf der zweiten Stufe, die dann nicht mehr verpflichtend ist, die darüber hinausgehenden und auf der dritten die speziell studienvorbereitenden. Für die Mathematik konkretisiert er: Wäre Lebensvorbereitung die einzige Aufgabe von Allgemeinbildung, würden die ersten sieben Jahre Mathematik ausreichen, in denen die Schüler das bürgerliche Rechnen inklusive der Prozent- und der Zinsrechnung und die elementare Geometrie erlernen. [He96; 151ff] Für die Schulinformatik hat man bisher nicht wirklich explizieren können, was solche basale Qualifikationen sein könnten. Auch die Kompetenzerwartungen der Sekundarstufe I [GI08] und für die Primarstufe [GI19] stehen in der Tradition des algorithmenorientierten Ansatzes, der – wie im folgenden Abschnitt gezeigt wird – vor allem fachwissenschaftliche Perspektiven fokussiert.

3 Ältere bis heute gültige Bestandaufnahmen

Ziel dieses Abschnitts ist es zu zeigen, dass nicht nur die fachübergreifenden, sondern auch die fachbezogenen Ansätze blinde Flecken in Bezug auf die drei verschiedenen zu berücksichtigenden Perspektiven aufweisen. Dazu wird zunächst auf Peschkes 1989 veröffentlichte Bestandaufnahme eingegangen, mit der er auf der eine „Krise des Informa-

tikunterrichts“ für die 1990er Jahre prognostizierte [Pe89]. Insbesondere reflektiert er darin die Beziehung von Informatikunterricht und ITG/IKG (3.1). Aus der kurz darauf veröffentlichten Bestandsaufnahme von Forneck [Fo92] zu den bis dahin vorliegenden fachdidaktischen Konzepten der Informatik geht zudem hervor, welche Herausforderungen die angestrebte Verbindung der drei Perspektiven mit sich bringen (3.2), die – das wird abschließend gezeigt – bis heute allenfalls im Ansatz gelöst sind. Hier geht es auch um die Frage, ob und wie lebenspraktische und darüber hinaus reichende (wissenschaftliche) Ziele überhaupt miteinander verzahnt werden können (3.3).

3.1 Zur „Krise des Informatikunterrichts“

Der Aufsatz zur prognostizierten *Krise des Informatikunterrichts* ist eigentlich eine Bestandsaufnahme zum Informatikunterricht nach Einführung der ITG/IKG. Peschke präsentiert fünf Schlussfolgerungen, die auf Erfahrungen basieren, aber nicht im strengen Sinne empirisch sind. Diese Schlussfolgerungen sind letztlich Hypothesen zum damaligen Stand der Informatik in der Schule: 1. Der Stellenwert und Qualität des Informatikunterrichts geben keinen Anlass zur Zufriedenheit; die Inhalte sind durch die partielle Einführung von Leistungskursen weiter diversifiziert worden. 2. Der 'Bildungskern' ist undeutlich geworden. Die Programmiersprache ist nicht mehr von zentraler Bedeutung. Dennoch konnten Algorithmen- und anwendungsorientierte Zugänge nicht auf einander bezogen werden. 3. Fachübergreifende inhaltliche Anforderungen wie die Diskussion um gesellschaftliche Aspekte lösen Unsicherheiten aus. Es fehlt diesbezüglich an entsprechenden Leitbegriffen und Grundkategorien. 4. Aufgrund der medialen Bindung an Hard- und Software kann man sich in der Informatik kaum auf Konstantes, Fundamentales und Exemplarisches besinnen. Im Informatikunterricht wird die Trennung zwischen Inhalt und Medium vielfach aufgehoben. 5. Die Grundbildung kann den Bedarf an informatischer Bildung nicht befriedigen. Der Bedarf wird wachsen, um ein Verständnis zu den neuen Technologien und deren Auswirkungen wirklich fördern zu können. [Pe89; 91ff] Erst durch die letzte Schlussfolgerung wird aus den Befunden in 1. bis 4. eine Krise. Wäre informatische Bildung prinzipiell verzichtbar, könnte Informatik als Wahlpflichtfach oder gar im AG-Bereich verbleiben. Die Diskussion um ihren Beitrag zur Bildung würde sich erübrigen. Rückblickend hat sich diese fünfte Hypothese bestätigt. Nicht nur die »Dagstuhl-Erklärung« [GI16b] weist darauf hin. Allerdings stellt sich auch die Frage, ob der Informatikunterricht – zumindest mit seinen derzeitigen Curricula und Umsetzungen – dazu geeignet ist. Die Zweifel ergeben sich u. a. aus der Reflexion der Schlussfolgerungen Peschkes.

Die *Diversifizierung* der Inhalte konnte trotz der Anstrengungen der Fachdidaktik Informatik nicht eingedämmt werden. Zwar haben die „fundamentalen Ideen der Informatik“ [Sc93] und danach auch die Bildungsstandards für die Sekundarstufen [GI08, GI16a] einen Beitrag zur Kanonisierung bzw. Konsolidierung geleistet. Diese kommen aber kaum zum Tragen, da sich insbesondere die *mediale Bindung* den Informatikunterricht immer wieder verändert hat: Dies betrifft auch die immer wieder neue Entwicklungsumgebungen (z. B. BlueJ) sowie Mini-Welten (z. B. Greenfoot, Scratch) und zudem immer neue

'Tangibles' (LEGO-Mindstorms, Raspberry Pi, Arduino, Calliope etc.).⁵ Zudem hat auch in Bezug auf die *fachübergreifenden Anforderungen* keine Ablösung von der technologischen Entwicklung stattgefunden. Dies betrifft vor allem die Thematisierung gesellschaftlicher oder kultureller Aus- bzw. Wechselwirkungen, die sich auch immer an gerade aktuellen oder neuartig erscheinenden Anwendungen der Informatik orientieren. Deren Thematisierung ist außerdem nur schwach mit den anderen Zielen verbunden, wie u. a. die Bestandaufnahme von Forneck im folgenden Unterabschnitt zeigt.

3.2 Revision der Ansätze informatischer Bildung

1992 legte Forneck eine Revision der bis dahin vorliegenden fachdidaktischen Ansätze vor, die empirisch fundiert ist.⁶ Er hat Unterrichtseinheiten gemäß dieser Ansätze gestalten, durchführen und analysieren lassen. Dabei wird deutlich, dass keiner der untersuchten Ansätze alle sich selbst gesetzten Ziele erreicht [Fo92]. In einem ersten Schritt wird auf Fornecks Erkenntnisse zum *algorithmienorientierten* Ansatz eingegangen, der sich in seinem Begründungszusammenhang auch auf die drei Perspektiven bezieht, [GI76] und im zweiten Schritt dann auf den *anwendungs-* und den *benutzungsorientierten* Ansatz, die dies auch tun.

Forneck beschreibt, dass der *algorithmienorientierte* Ansatz zwar die Ziele erreicht, die die Wissenschaftspropädeutik betreffen, die anderen Perspektiven jedoch weitestgehend unberücksichtigt lässt. Er schreibt: „[N]ormative und subjektive Fragen [fallen] aus dem Horizont der Problemstellung einer algorithmienorientierten Wirklichkeitskonstruktion ... Die additive Hinzufügung von Fragen der Bewertung gesellschaftlicher Auswirkungen der Informatik verdankt sich der transzendentalen Trias von Objektivität, Information und Subjektivität. Die ersten beiden Regionen der algorithmischen Weltkonstitution sind, da rationalisierbar, eigentlicher Gegenstand des Unterrichts. Die letzte Wirklichkeitsregion 'Subjektivität' bleibt dem algorithmischen Denken unfassbar. Deshalb können in algorithmienorientierten Konzeption im 'eigentlichen' Unterricht Fragen normativen Richtigkeit und der subjektiven Wahrhaftigkeit nicht integriert werden, weshalb sie additiv hinzugefügt werden müssen.“ [Fo92; 177]

Die anderen eher auf die Lebenspraxis ausgerichteten Ansätze (namentlich der *benutzungsorientierte* Ansatz, der maßgeblich von der ITG/IKG bestimmt ist sowie der *anwendungsorientierte*, der als expliziter Gegenentwurf zum *algorithmienorientierten* gestaltet wurde) erreichen die selbstgesteckten Ziele ebenso wenig. So schreibt Forneck abschließend zum anwendungsorientierten Ansatz: „[Es] wurde aufgewiesen, dass es in den untersuchten Unterrichtsreihen nicht gelingt, nach einer Algorithmisierung und Programmierung diese Tätigkeiten auf gesellschaftliche Fragestellungen zurückzubeziehen. Dies liegt auch an der Komplexität und Voraussetzungshaftigkeit der Algorithmisierung und Programmierung.“ Zudem führt Forneck aus, dass es diesen Ansätzen nicht

⁵ Tatsächlich ist dies nur eine Arbeitshypothese, die hier nicht ausführlich erörtert werden kann.

⁶ Auch der Autor dieses Aufsatzes hat Fragen zu Fornecks Methodik. Die Ergebnisse zu den Herausforderungen der Verbindung der drei in Frage stehenden Perspektiven scheinen hiervon jedoch unabhängig.

gelingen ist, den Begriff 'Anwendung' zu klären: Ist *Anwendung* im Sinne von 'Application' oder im Sinne der Anwendung informatischer Denkweisen gemeint? [ebd.; 229f]

Insgesamt resümiert er, dass man in allen ihm vorliegenden Ansätzen „den eigentümlichen Charakter technischer Objektivität.“ [ebd.; 272] unberücksichtigt lässt. „Dem *konstruktiven* Charakter der Technik muss ein konstruktiver Charakter (Informations)technischer Bildung entsprechen. Dieser nicht analytische, sondern konstruktive Charakter ist die eigentliche fachdidaktische Herausforderung, die mit der unterrichtlichen Behandlung von Technik verbunden ist.“ [ebd.; 273] Diese Herausforderung beschreibt er wie folgt: So „vermag ein Informatikunterricht die Leistung traditioneller Fächer insofern nicht erreichen, weil die Technisierung der Lebenswelt einen Entwicklungsstand angenommen hat, der es nicht mehr möglich erscheinen lässt, dass Individuen verstehende Rekonstruktionen der Wirklichkeit allein, sozusagen als Ausfluss eines wie auch immer gearteten Fächerkanons, zu leisten vermöchte.“ [ebd.] Allerdings leisten auch fächerübergreifende Ansätze dieses wohl nicht, weil sie die Informatik nicht miteinbeziehen. Diese Herausforderung wiederum ist allerdings aus der Didaktik der Informatik heraus allein nicht zu lösen, wohingegen die Integration der Sichten anderer Fächer in die Informatik von der Didaktik der Informatik geleistet werden kann. Die damit verbundenen Herausforderungen sind bis heute weiterhin ungelöst.

3.3 Zu den bis heute ungelösten Herausforderungen

In der Didaktik der Informatik hat man sich kaum mit Fornecks Aufforderungen befasst, die didaktische Analyse aus Sicht der Begriffe 'Anwendung' und 'Technik' zu intensivieren. Es wäre darum gegangen zu klären, auf was sich Anwendung bezieht (Denkweisen oder Informatiksysteme) und den (besonderen) technischen Charakter der Informatik zu berücksichtigen. Stattdessen hat man sich vornehmlich an der Mathematik – wie oben schon dargestellt – und deren Bildungsziele orientiert, insbesondere auch, was die über das Fach hinausreichenden Ziele betrifft. Diese werden in der Informatik vor allem unter der Überschrift »Computational Thinking« (CT) propagiert. 'Informatisches Denken' (als nicht direkte aber nahliegende Übersetzung) wird als durch die Informatik und auch in vielen anderen Bereichen anwendbares Ziel ausgegeben. CT setzt damit auf 'Anwendung der Denkweisen' statt auf 'Anwendungen der Informatik'.

Ursprünglich steht, wie Denning, Tedre und Youngpradit beschreiben, CT im direkten Zusammenhang mit der Entwicklung programmierbezogener Fähigkeiten. Diese werden dann als auf die Lebenswirklichkeit der Lernenden übertragbar dargestellt. Hierzu stellen Denning u. a. jedoch dar, dass viele Lernende den hier notwendigen Transfer offenbar nicht leisten. Sie sehen gar in diesen über die Informatik hinausreichenden Deutungen informatischer Denkweisen sogar einen „computational chauvinism“ [DTY17; 33].

Bezüglich des CT wird in den letzten Jahren ausdrücklich auf J. Wing Bezug genommen, die 2006 die Diskussion darum wiederbelebt hat. Weniger Beachtung findet, dass sie dabei Widersprüchliches fordert. Unter der Überschrift „What it is, what it isn't“ benennt sie u. a. zwei Ausschlüsse: „Ideas, not artifacts“ sowie „Conceptualizing, not

programming” während sie andererseits schreibt „Complements and combines mathematical and engineering thinking” [Wi06; 35]. Die Komplementarität nimmt auf den besonderen technischen Charakter der Informatik Bezug. Aber wie soll die Komplementarität verstanden werden, wenn man wirklich die Artefakte (Informatiksysteme) und das Programmieren außen vor lässt? Man würde tatsächlich den technischen Charakter der Informatik negieren. Außerdem wird nicht der Versuch unternommen, die anwendungsbezogene Perspektive für die Informatik zu klären. Dazu müsste es eigentlich 'Conceptualizing within programming' sowie 'Ideas by artifacts' heißen. Aus Letzterem würde sich ein rekonstruktives bzw. sogar experimentelles Vorgehen ergeben.

So bezieht CT in der Deutung von J. Wing zwar den (besonderen) technischen Charakter der Informatik mit ein, schlägt sich allerdings in der Deutung des Begriffes Anwendung einseitig auf die Seite der Anwendung der Denkweisen, da sie die Informatiksysteme ausblendet. Die durchaus vorhandenen Zusammenhänge von Anwendung der Denkweisen und Artefakten können so kaum verstanden werden. Diese besser zu verstehen, hat Peschke einen Ansatz vorgelegt [Pe90]. Dieser Vorschlag konnte jedoch in der Didaktik der Informatik nicht überzeugen. Zu nah positioniert er die Schulinformatik in der Nähe des Lernbereichs Arbeitslehre/Technik. Unter anderem definiert Peschke in diesem Aufsatz Rollen im Umgang mit – wie man heute sagen würde – Informatiksystemen, auch um Unterschiede zur ITG/IKG darzustellen. Beide Bildungsangebote adressieren diese vier Rollen, wenn auch in unterschiedlicher Ausprägung. Diese Rollen definiert er mit „Betroffener, Bediener, Benutzer und Gestalter“.⁷ Eine im Unterschied zu dieser normativen Setzung empirisch fundierte Kategorisierung des Umgangs mit Informatiksystemen hat viele Jahre später M. Knobelsdorf mit ihrer Analyse von Computernutzungsbiografien vorgelegt. Sie nennt auch vier Umgangsweisen: „ausprobieren“, „anwenden“, „verändern“ und „erzeugen“. Als zusätzliches Ergebnis kann sie zeigen, dass „Erzeugen“ nur von informatik-affinen Personen ausgefüllt wird [Kn08]. Damit stellt sich insbesondere die Frage, ob es klug ist, den Informatikunterricht beim 'erzeugen' (gestalten) zu starten. Im folgenden Abschnitt darzustellenden Zugang wird entsprechen beim „Ausprobieren“ bzw. „Anwenden“ angesetzt, um darüber zum „Verändern“ und ggf. zum „Erzeugen“ zu gelangen.

4 Didaktische Aufbereitung via „Dagstuhl in Progress“

Das »Dagstuhl-Dreieck« ist wie die dazugehörige Erklärung zunächst nur ein Kommunikationsmittel für bildungspolitische Auseinandersetzung. Es folgt letztlich der Marketing-Maxime 'keep it small and simple' und ist deswegen für die bildungspolitische Auseinandersetzung besonders gut geeignet. Aber auch für den wissenschaftlichen und fachdidaktischen Diskurs bieten die darin benannten Strukturmomente und Prinzipien die Möglichkeit den Prozess der Digitalisierung zu analysieren. Im Zuge einer Tagung mit

⁷ Er unterscheidet 'Bediener' und 'Benutzer' insofern, da *bedienen* mit der Unterordnung unter bzw. Anpassung an die Systemrationalität zusammenhängt, während *benutzen* deutlich selbstständiger und selbstbewusster ist [Pe90; 32]

dem programmatischen Titel „Dagstuhl in Progress“ haben neben Informatikern und Medienpädagogen auch Medienwissenschaftler das Dreieck weiterentwickelt und begrifflich präzisiert. Nach eineinhalb Tagen Diskussionen war man trotz anderer Überlegungen bei drei Perspektiven geblieben,⁸ hatte diese jedoch umbenannt. An die Stelle der letztlich zu kurz greifenden Fragen waren nun Inhalte getreten (vgl. hierzu die Darstellung in [En18] und Abb. 2 links). Dieses auf der Arbeitstagung weiterentwickelte Dreieck ist letztlich nur ein weiteres Zwischenresultat der notwendigen begrifflichen Diskussionen, die zu führen eine Daueraufgabe ist und nicht durch weitere bildungspolitische Verlautbarungen abgeschlossen werden kann. Es bildet im Folgenden den Ausgangspunkt einer didaktischen Aufbereitung, die die Ergebnisse der Bestandaufnahmen miteinbezieht. Es wird ein Zugang skizziert, der beim individuellen Umgang und der gesellschaftlichen Einbettung ansetzt und zugleich die Rolle von Technik und Techniken für den Menschen diskutiert, womit auch die Beziehung von Anwendungen der Informatik und Anwendung ihrer Denkweisen miteinbezieht ('ideas by artifacts'). Dieser Zugang stellt die Anwendungen der Informatik in einen technikgenetischen Zusammenhang, in dem die Bedeutung von digitalen Artefakten für Kultur, Arbeit und Freizeit herausgearbeitet wird.

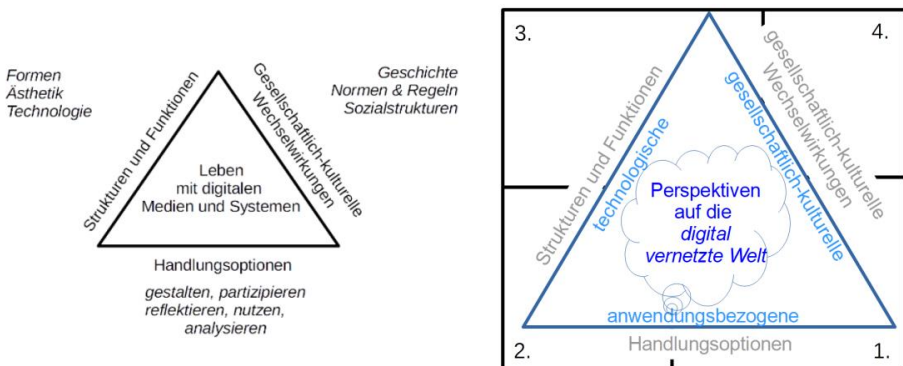


Abb. 2: Gegenüberstellung modifiziertes „Dagstuhl-Dreieck“ als (Zwischen-)Ergebnis der Arbeitstagung „Dagstuhl in Progress“ und Schaubild zur Verortung der vier Bausteine

Der weitere Aufbau ergibt sich aus dem Versuch einmal im Uhrzeigersinn um das Dreieck herum zu gehen (siehe Abbildung 2 rechts). Dabei wird in den ersten drei Bausteinen auf je zwei Perspektiven Bezug genommen. 2. und 3. beziehen sich dann auf die Strukturen und Funktionen solcher 'Medien' unterhalb der sichtbaren Oberfläche. In 2. geht es um die Vernetzung und Medienintegration am Beispiel von Dateiformaten, HTML und http. Diese werden auch in ihrer geschichtlichen Entwicklung ausgehend von der Diskussion um 'Multimedia' aus den 1990er Jahre miteinbezogen. In 3. wird auf das Besondere des technischen Artefakts 'Software' Bezug genommen. Dieses geschieht weniger konstruktiv denn rekonstruktiv. Hier ist Zeit und Raum das Zusammenspiel von

⁸ So hatte der Tulodziecki in einem auf der Tagung vorliegenden, jedoch unveröffentlichten Handout als Diskussionsbeitrag zur „Dagstuhl-Erklärung“ [Gl16b] angeregt, weitere Perspektiven zu berücksichtigen.

modellieren und implementieren ('conceptualizing within programming') zu thematisieren. 4. Zum Abschluss wird auf wichtige Normen und Gesetze Bezug genommen, die sich auch aus den Besonderheiten von Software ergeben.

5 Fazit und Ausblick

Dieser Rückblick auf frühere didaktische Diskussionen macht deutlich, dass eine *Bildung in der digital vernetzten Welt* wohl drei Perspektiven miteinander verbinden muss. Diese Perspektiven sind Teil der Begründungszusammenhänge für das Fach Informatik, ohne dass in der Praxis eine echte Verbindung dieser Perspektiven hergestellt wird. Allerdings leisten auch fachübergreifende Lernbereiche dies nicht. *Bildung in der digital vernetzten Welt* ist damit zwar ohne Informatik denk-, aber wohl nicht umsetzbar. Der Vorschlag aus Abschnitt 4 beinhaltet den Versuch durch eine technikgenetische Sicht auf die Anwendungen und die Denkweisen der Informatik, Lebensvorbereitung und Wissenschaftspropädeutik zu verbinden. Vielleicht ist diese Verbindung auch nicht möglich. Insofern ist Heymanns drei Stufen beinhaltendes Kurssystem eine andere mögliche Lösung, für die hier kein Vorschlag gemacht wird.⁹ Dieser Aufsatz ist damit die Aufforderung (auch des Autors an sich selbst) diesen Zugang und die Strukturierung in der Praxis zu evaluieren, der möglicherweise erst in der Sekundarstufe II umsetzbar ist. Auch dort könnte noch ein Beitrag zu einer *Bildung in der digitalen vernetzten Welt* geleistet werden und die Defizite in der allgemeinen Bildung aufgearbeitet werden.

Literaturverzeichnis

- [BLK87] Bund-Länder-Kommission für Bildungsplanung und Forschungsförderung (Hrsg.): Gesamtkonzept für informationstechnische Bildung. Materialien zur Bildungsplanung. Heft 16, Bonn, 1987
- [DTY17] Denning, P.J.; Tedre, M.; Youngpradit, P.: The Profession of IT. Misconception about Computer Science. *Communications of the ACM*. March 2017/Vol. 60, No. 3, S. 31 – 33. doi:10.1145/3041047
- [En14] Engbring, D.: Zum Verhältnis von Informatik und Naturwissenschaften. Ein Vorschlag zur MINT-Förderung. In: Thomas, M., Weigend, M. (Hrsg.): *Informatik und Natur*. 6. Münsteraner Workshop zur Schulinformatik - 9. Mai 2014. Books on Demand GmbH, Norderstedt. S. 9 – 18
- [En18] Engbring, D.: Überlegungen zu einem Beitrag zur Lehrerbildung in der digital vernetzten Welt. Ein auf Erfahrungen gestützter Bericht und Diskussionsbeitrag. In: Thomas, M., Weigend, M. (Hrsg.): *Informatik und Medien*. 8. Münsteraner Workshop zur Schulinformatik - 18. Mai 2018. Books on Demand GmbH, Norderstedt. S. 95 – 106

⁹ Hierzu können die laufenden Projekte im Primar- und unteren Sekundarbereich auch in dieser Hinsicht ausgewertet werden.

- [Fo92] Forneck, H. J.: Bildung im informationstechnischen Zeitalter. Untersuchung der fachdidaktischen Entwicklung der informationstechnischen Bildung. Sauerländer, Aarau Frankfurt a. M. Salzburg, 1992
- [GI76] Brauer, W. u. a.: Zielsetzungen des Informatikunterrichts. In: Zentralblatt für Didaktik der Mathematik 8 (1976) S. 35-43
- [GI08] Gesellschaft für Informatik. Grundsätze und Standards für die Informatik in der Schule Bildungsstandards für die Sekundarstufe I. https://www.informatikstandards.de/docs/bildungsstandards_2008.pdf (16.2.2019)
- [GI16a] Gesellschaft für Informatik. Grundsätze und Standards für die Informatik in der Schule Bildungsstandards für die Sekundarstufe II. https://www.informatikstandards.de/docs/Bildungsstandards_SII.pdf (16.2.2019)
- [GI16b] Gesellschaft für Informatik. Bildung in der digital vernetzten Welt. <https://gi.de/themen/detail/dagstuhl-erklaerung-bildung-in-der-digital-vernetzten-welt-1/> (16.2.2019)
- [GI19] Gesellschaft für Informatik. Kompetenzen für informatische Bildung im Primarbereich https://www.informatikstandards.de/docs/v142_empfehlungen_kompetenzen-primarbereich_2019-01-31.pdf (16.2.2019)
- [He96] Heymann, H. W.: Allgemeinbildung und Mathematik. Beltz, Weinheim Basel, 1996
- [ITG90] Der Kultusminister des Landes Nordrhein-Westfalen (Hrsg.): Vorläufige Richtlinien zur informations- und kommunikationstechnologischen Grundbildung in der Sekundarstufe I. Ritterbach. Frechen. 1990
- [Kn11] Knobelsdorf, M.: Biographische Lern- und Bildungsprozesse im Handlungskontext der Computernutzung. Dissertation FU Berlin. <http://ddi.cs.uni-potsdam.de/Examensarbeiten/Knobelsdorf2011.pdf> (16.2.2019)
- [KM15] Knobelsdorf, M.; Magenheimer, J. u. a.: Computer Science Education in North-Rhine-Westphalia, Germany. A Case Study. ACM Transactions on Computing Education (TOCE) - Special Issue II on Computer Science Education in K-12 Schools Volume 15 Issue 2, May 2015 Article No. 9 ACM New York, NY, USA doi>10.1145/271631
- [KM18] Koerber, B.; Müller, J.: Analoge Erfahrungen. Editorial der Log IN 187/188. S. 3
- [KMK17] Kultusministerkonferenz. Strategie der Kultusministerkonferenz. „Bildung in der digitalen Welt“. Beschluss der Kultusministerkonferenz vom 08.12.2016 in der Fassung vom 07.12.2017
- [Pe89] Peschke, R.: Die Krise des Informatikunterrichts in den neunziger Jahren. In: Brauer, W.; Stetter, F. (Hrsg.): Informatik und Schule 1989: Zukunftsperspektiven der Informatik für Schule und Ausbildung, Springer. S. 89-98
- [Pe90] Peschke, R.: Grundideen des Informatikunterrichts. Erfahrungen und Perspektiven aus den „alten“ Ländern der Bundesrepublik Deutschland. In: LOG IN 10 Heft 6, S. 25-33
- [Sc93] Schwill, A.: Fundamentale Ideen der Informatik. In: Zentralblatt für Didaktik der Mathematik 25 Heft 1 (1993) S. 20-31.
- [Wi06] Wing, J.: Computational Thinking. Communications of the ACM. March 2006/ Vol. 49, No. 3, 33-35

Aufgabe ist nicht gleich Aufgabe – Vielfältige Aufgabentypen bewusst in Scratch einsetzen

Katharina Geldreich¹, Mike Talbot¹ und Peter Hubwieser¹

Abstract: Aufgaben sind ein fester Bestandteil jeden Unterrichts und können je nach Einsatz verschiedene didaktische Funktionen übernehmen. Auch beim Programmierenlernen spielen Aufgaben eine zentrale Rolle und können bei entsprechender Auswahl und Anwendung den Lernprozess der Schülerinnen und Schüler entscheidend beeinflussen. In der internationalen Fachliteratur lassen sich verschiedene Frameworks und didaktische Ansätze finden, die sich damit auseinandersetzen, welche Entwicklung die Lernenden beim Programmierenlernen durchlaufen. Diese Ansätze und einhergehende mögliche Aufgabenstellungen wurden im Rahmen einer Fortbildungsmaßnahme für Grundschullehrkräfte zum Thema Algorithmik und Programmierung thematisiert. Damit die Teilnehmerinnen und Teilnehmern gleichzeitig eine große Vielfalt an Aufgabenstellungen kennenlernen, wurden bei dem Design der Aufgaben bereits bestehende Aufgabenklassifikationen für Programmieraufgaben herangezogen. Der Praxisbeitrag beschreibt die erwähnten theoretischen Ansätze und zeigt, wie sie bei der Aufgabengestaltung umgesetzt werden können. Unser Fokus liegt darauf, wann welcher Aufgabentyp im Lernprozess sinnvoll eingesetzt werden kann. Sämtliche Aufgaben beziehen sich dabei auf die blockbasierte Programmiersprache Scratch.

Keywords: Programmieren; Aufgabentypen; Scratch

1 Einleitung

Aufgaben können auf verschiedene Ziele ausgerichtet sein und dementsprechend unterschiedliche didaktische Funktionen übernehmen. Dabei unterscheidet man grundlegend zwischen Aufgaben, die Lernende beim Aufbau und der Festigung von Wissen und Fähigkeiten unterstützen, sowie prüfenden Aufgaben zur Ermittlung des Leistungsstands [Ri05]. Erstere stellen für Lehrkräfte ein zentrales Gestaltungselement für den Unterricht dar und können ihnen helfen, Lernprozesse gezielt zu steuern. Auch beim Erlernen einer Programmiersprache sind geeignete Aufgaben wichtig, um den Lernprozess der Schülerinnen und Schüler bei entsprechender Auswahl und Anwendung entscheidend zu beeinflussen. Wie der Prozess des Programmierenlernens gestaltet werden sollte, ist nicht abschließend geklärt und stellt Lehrende wie Lernende vor Herausforderungen [MR02]. Verschiedene Autoren beschreiben didaktische Ansätze, in denen die Lernenden zunächst mit bereits bestehenden Programmen arbeiten und diese immer weiter modifizieren, bis sie schließlich selbstständig programmieren. Diese Ansätze und damit zusammenhängende Aufgaben waren Gegenstand einer Fortbildung für Grundschullehrkräfte zum Thema *Algorithmen und*

¹ Technische Universität München, School of Education, Professur für Didaktik der Informatik, Arcisstr. 21, 80333 München, {katharina.geldreich,mike.talbot,peter.hubwieser}@tum.de

Programmierung. Um die Lehrkräfte gleichzeitig mit unterschiedlichen Aufgabentypen in Scratch vertraut zu machen, wurde bei der Aufgabengestaltung eine bestehende Taxonomie für Programmieraufgaben herangezogen. Dieser Beitrag gibt zunächst einen Einblick in die zuvor erwähnten didaktischen Ansätze und Aufgabentaxonomien und beschreibt, wie im Kontext der Fortbildung damit gearbeitet wurde. Um einen möglichst praxisnahen Einblick zu geben, werden verschiedene konkrete Beispielaufgaben beschrieben, die sich auf die visuelle Programmiersprache Scratch beziehen. Diese können sowohl im Fortbildungs- als auch im Unterrichtskontext eingesetzt werden.

2 Theoretischer Hintergrund

2.1 Scaffolding im Unterricht

Der Begriff *Scaffolding* (zu dt. *Gerüst*) bezieht sich im pädagogisch-psychologischen Kontext auf die Unterstützung eines Lernprozesses durch verschiedene Maßnahmen wie Anleitungen, Denkanstöße oder andere Hilfestellungen [Sc06]. Diese Maßnahmen sollen die Schülerinnen und Schüler auf dem Weg zum selbstständigen Handeln begleiten und werden mit zunehmendem Können zurückgenommen. Scaffolding kann auf verschiedenen Ebenen stattfinden und sich auf die Planung von Lernaufgaben und Unterrichtssequenzen (*Makro-Ebene*), die Strukturierung von Lernaufgaben in einzelne Arbeitsschritte (*Meso-Ebene*) oder die Interaktion zwischen Lehrkraft und Lernenden (*Mikro-Ebene*) beziehen [Va96]. Im Kontext der Informatikdidaktik untersuchen sowohl Lee et al. [Le11] als auch Sentance und Waite [SW17], welche Entwicklung die Schülerinnen und Schüler auf dem Weg zum selbstständigen Programmieren durchlaufen sollten und wie man sie im Sinne des Scaffolding dabei unterstützen kann. Lee et al. entwickelten das *Use-Modify-Create*-Lehrkonzept, in dem die Lernenden zunächst mit bereits bestehenden Programmen arbeiten und diese schrittweise modifizieren bis sie schließlich eigene Programme erstellen. Daraufhin folgt ein iterativer Prozess des Testens, Analysierens und der Weiterentwicklung. Sentance und Waite stellen den Lehransatz PRIMM vor, der unter anderem auf der Arbeit von Lee et al. basiert. PRIMM ist die Abkürzung für *predict, run, investigate, modify* und *make* (zu dt. *vorhersagen, ausführen, untersuchen, anpassen, machen*). In beiden Ansätzen arbeiten die Lernenden zunächst mit bestehenden Programmen, nehmen kleinere und größere Veränderungen daran vor und erstellen schließlich komplette Programme eigenständig (Abb. 1).

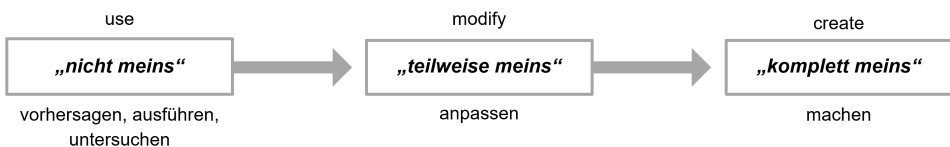


Abb. 1: Scaffolding beim Programmierenlernen

2.2 Aufgabentypen

Aufgaben sind ein zentraler Bestandteil des Informatikunterrichts und spielen sowohl für das Erarbeiten neuer Inhalte als auch für die Ermittlung des Leistungsstands eine wichtige Rolle. Besonders beim Programmierenlernen wird betont, dass sich Expertise nur durch praktische Aufgaben und intensives Üben einstellen kann [HM06]. Für die Entwicklung von Aufgaben lassen sich verschiedene Qualitätskriterien finden, an denen sich Lehrende orientieren können [BHP14]. Neben der Berücksichtigung von Faktoren wie Kompetenz- und Kontextorientierung wird empfohlen, die Aufgabentypen zu variieren. Welche Aufgabentypen im Kontext des Programmierens verbreitet sind, wurde mehrfach untersucht. Dabei entstanden verschiedene Aufgabentaxonomien, die z.B. deduktiv durch die Analyse verschiedener Curricula [Bo08] sowie induktiv aus der Analyse von Programmieraufgaben abgeleitet wurden [RBH13]. Ruf et al. [RBH15] unterscheiden zwölf verschiedene Aufgabentypen, die sich durch die zur Lösung erforderlichen Schülerhandlungen (z.B. Code testen, Code schreiben) und die Art der Darstellung der gegebenen Informationen (z.B. Text, Code, Bilder) unterscheiden.

3 Kontext

Um zu beurteilen, ob sich Aufgaben für die jeweilige Zielgruppe und deren Kenntnisstand eignen, ist es wichtig, dass Lehrkräfte entsprechende Kriterien kennenlernen [HB14]. Neben allgemeindidaktischen Kriterien wie z.B. Offenheit der Aufgabe, Lebensweltbezug [Ma10], sollten Aufgaben immer auch im Kontext der jeweiligen Fachdidaktik gesehen werden. Der Bereich *Aufgaben* ist daher auch in Modellen des fachdidaktischen Wissens (*pedagogical content knowledge*) für die Informatik zu finden [Hu13].

Zur Ausbildung von unterrichtsmethodischem Know-How wurden Grundschullehrkräfte im Rahmen einer mehrtägigen Fortbildungsmaßnahme mit verschiedensten Programmieraufgaben konfrontiert. Um die Programmierkenntnisse der Teilnehmenden weiter zu festigen, wurden sämtliche Programmieraufgaben zunächst eigenständig bearbeitet. Im Anschluss wurde der *Use-Modify-Create*-Ansatz, PRIMM sowie die Aufgabentaxonomie von Ruf et al. eingeführt (vgl. Punkt 2). Um die Theorie mit der Praxis zu verknüpfen, ordneten die Lehrkräfte die Aufgaben im Anschluss in die Scaffolding-Kategorien *use*, *modify* und *create* ein. Hier arbeiteten die Lehrkräfte zu zweit mit einem Satz Kärtchen, auf denen die Programmieraufgaben in Miniatur abgebildet waren und die entsprechend gruppiert werden mussten. Dieses Vorgehen förderte die Auseinandersetzung mit den verschiedenen Aufgabentypen in Bezug auf die Einordnung in die Scaffolding-Kategorien.

Die Fortbildung ist Teil des Projekts *AlgoKids – Algorithmen für Kinder*, das vom Bayerischen Staatsministerium für Unterricht und Kultus gefördert und über einen Zeitraum von zwei Jahren durchgeführt wird [GTH18]. Im Rahmen des Projekts werden insgesamt 40 Grundschullehrkräfte in den Bereichen *Algorithmik* und *Programmierung* fachwissenschaftlich und fachdidaktisch fortgebildet und bei eigenen Unterrichtsversuchen begleitet.

4 Aufgaben

Die nachfolgende Auswahl zeigt exemplarisch, welche Aufgabentypen im Laufe der Fortbildung bearbeitet wurden. Die teilnehmenden Lehrkräfte haben in einer ersten Fortbildung bereits intensiv mit Scratch gearbeitet, der Fokus lag hierbei jedoch vermehrt auf fachwissenschaftlichen Inhalten, insbesondere den algorithmischen Grundstrukturen.

4.1 Programme werden gelesen, ausgeführt, untersucht (*use*)

Programme nachbauen (Code→Code)

Ein komplett vorgegebenes Programm wird in Scratch nachprogrammiert und ausgeführt. Im Anschluss daran sollen Stellen markiert werden, bei denen man besonders sorgfältig sein musste (z.B. die Auswahl der richtigen Kostüme, Einfügen der *fühlen*-Blöcke in eine bedingte Anweisung). Der Fokus dieser Aufgabe liegt vorrangig auf der Bedienung von Scratch und der Handhabung der Programmierblöcke.

Blöcke erkunden (Code→Text)

Die Lernenden öffnen ein bereits bestehendes Scratch-Projekt und führen dieses aus. Auf einem Aufgabenblatt wird notiert, welche Funktionen die einzelnen Blöcke haben (Abb. 2). Aufgaben dieser Form eignen sich besonders für die Einführung von noch unbekanntenen Blöcken oder komplett neuen Scratch-Erweiterungen (z.B. die Übersetzungsblöcke in Scratch 3).

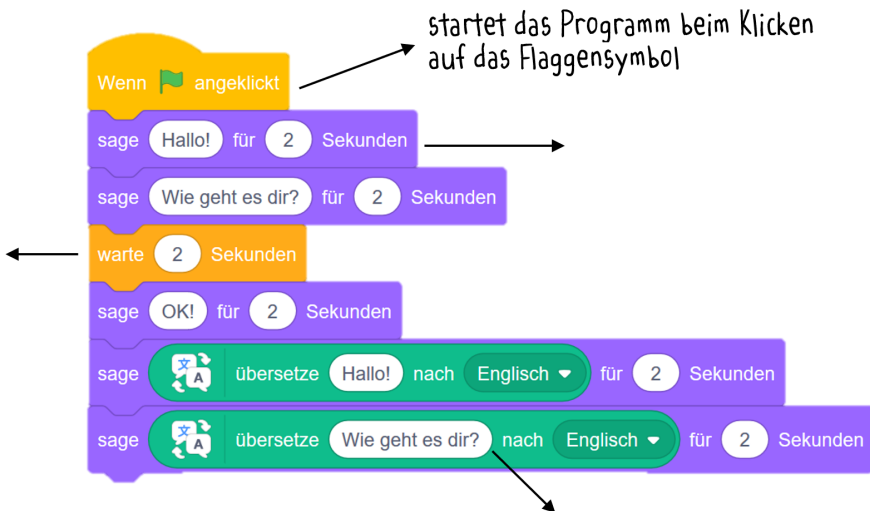


Abb. 2: Blöcke erkunden – Welche Funktion haben die einzelnen Blöcke?

Programme lesen (Code→Text)

Ein vorgegebenes Programm wird zunächst auf einem Aufgabenblatt gelesen (Abb. 3).

Danach soll beschrieben werden, was passiert, wenn der Mauszeiger unterschiedliche Positionen auf der Bühne einnimmt (z.B. Mauszeiger steht auf Position 2 → Schmetterling bewegt sich in Richtung des Igels). Die Lösungen werden im Anschluss durch Ausführen des Programms überprüft.

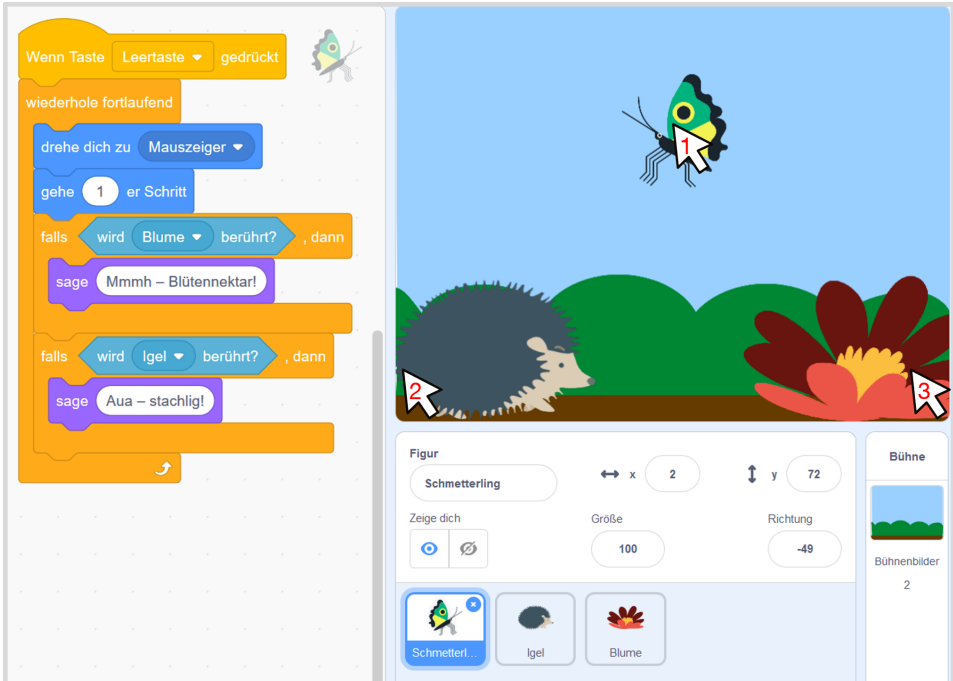


Abb. 3: Programme lesen – Was passiert nach dem Drücken der Leertaste, wenn der Mauszeiger auf den verschiedenen Positionen steht?

Programme lesen und Fehler finden (Code→Code)

Ein vorgegebenes Programm soll auf Fehler überprüft werden. Der Weg eines Marienkäfers durch ein Raster wird zunächst verfolgt (Abb. 4). Im Anschluss sollen Fehler im Programm gekennzeichnet werden, die dazu führen, dass der Käfer nicht bei seinem Ziel, der Blume, ankommt.

Programme testen (Code→Text)

Es soll getestet werden, ob ein bestehendes Programm das tut, was es soll. Hierzu wurde ein Programm erstellt, in dem eine Zahlenfee zwei durch den Nutzer eingegebene Variablen addiert und das Ergebnis ausgibt. Die Lernenden füllen eine Tabelle aus, in der sowohl die beiden Eingaben als auch deren Summe erfasst werden. Zusätzlich soll getestet werden was passiert, wenn Dezimalzahlen, besonders lange Ziffern oder Wörter eingegeben werden.

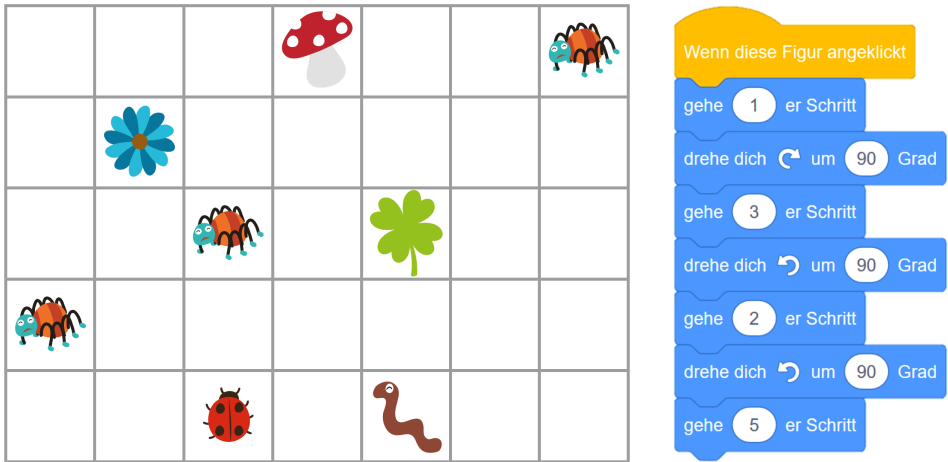


Abb. 4: Programme lesen und Fehler finden – Wie muss man das Programm verändern, dass der Marienkäfer bei der Blume ankommt?

4.2 Programme werden verändert, angepasst, erweitert (*modify*)

Programme debuggen (Code→Text+Code)

Ein Programm erfüllt nicht die gewünschte Funktion (Abb. 5). Es wird analysiert, wo sich der Fehler befindet und das Programm wird angepasst, um ihn zu beheben. Der Fokus liegt hier auf dem genauen Nachvollziehen des Programmablaufs.

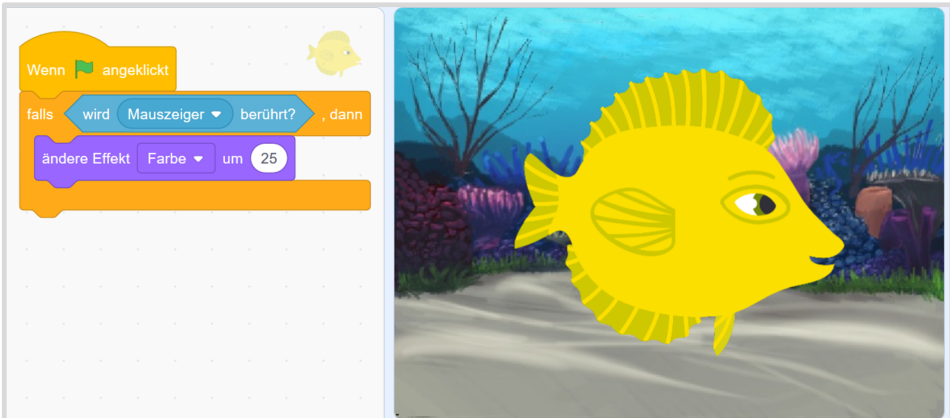


Abb. 5: Programme debuggen – Warum wechselt der Fisch beim Anklicken der grünen Flagge nicht die Farbe?

Programme erweitern (Code+Text→Code)

Ein Programm ist in Teilen vorgegeben und soll um weitere Funktionen erweitert werden. Die Bewegung einer Figur lässt sich bereits mit den Pfeiltasten steuern. Das Programm soll so erweitert werden, dass die Figur einen Dialog startet, wenn sie eine andere berührt.

Programme optimieren (Code→Code)

Ein fertiges Programm, welches die gewünschte Funktion scheinbar erfüllt, wird mittels verschiedener Fallunterscheidungen untersucht. Der Programmcode soll so optimiert werden, dass die gewünschte Funktion in jeder Situation erfüllt wird. In einem vorgegebenem Programm läuft eine Maus nacheinander zu verschiedenen Futterquellen. Da die Maus jedoch nur einzelne Koordinaten abläuft, kann sie das Futter nicht mehr finden, sobald man es auf der Bühne verschiebt. Die Lernenden sollen das Programm so modifizieren, dass die Maus immer zu ihrem Futter findet (Abb. 6).

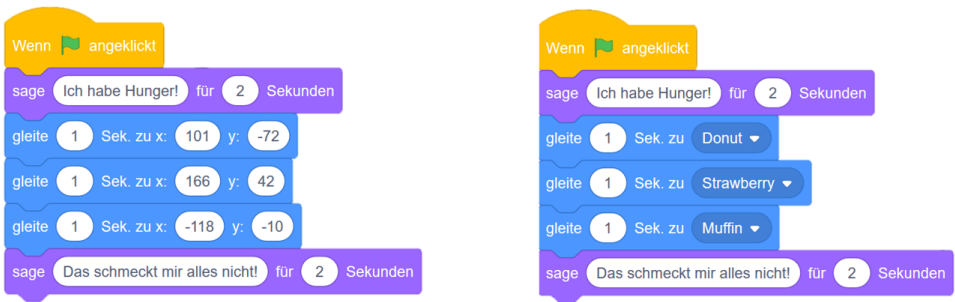


Abb. 6: Programme optimieren – Mit dem rechten Programm läuft die Maus auch zu ihrem Futter, wenn man die Figuren auf der Bühne verschiebt

Szenario in Scratch anpassen (Code+Text→Code)

Für ein vorgegebenes Programm soll das Bühnenbild so angepasst werden, dass die gewünschte Funktionalität erfüllt wird. Ein Geist möchte zu seinem Schlüssel (Abb. 7). Er kann bereits mit den Pfeiltasten gesteuert werden, ist aber nicht in der Lage, durch die Wand zu seinem Schlüssel zu fliegen. Das Bühnenbild soll so angepasst werden, dass der Geist zu seinem Schlüssel fliegen kann (z.B. durch Verändern der Randfarbe, Vergrößern des Rechtecks).

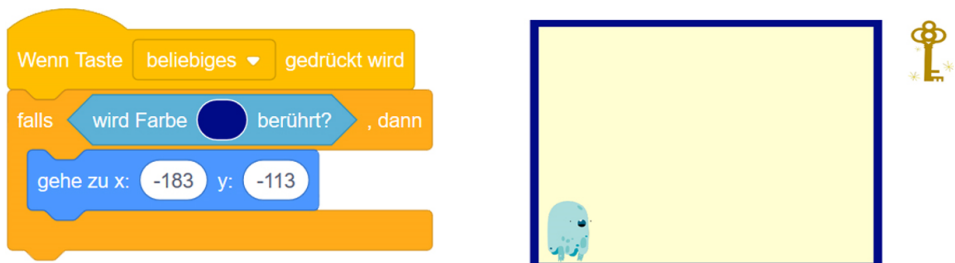


Abb. 7: Szenario in Scratch anpassen – Wie kann man das Bühnenbild verändern, dass der Geist zu seinem Schlüssel kommt?

4.3 Programme werden eigenständig erstellt (*create*)

Programmieren nach bildlichen Vorgaben (Bild→Code)

Eine vorgegebene Bildergeschichte soll in Scratch nachprogrammiert werden (Abb. 8). Dabei kann variiert werden, ob die Lernenden zusätzlich zu der Programmierung der Figur z.B. auch die Bühnenbilder selbst gestalten sollen.

In der Klassifikation von Ruf et al. [RBH15] gibt es den Aufgabentyp *Diagramm→Code*, in dem ein Programm zu einem Diagramm (z. B. Klassen-, Zustands-, Sequenzdiagramm) geschrieben wird. Da Diagramme dieser Form in der Grundschule nicht eingeführt werden, wurde stattdessen *bildliche Vorgaben* gewählt.



Abb. 8: Programmieren nach bildlichen Vorgaben – Die Bildergeschichte soll in Scratch implementiert werden

Programmieren nach textlichen Vorgaben (Text→Code)

Das Schreiben von Code nach textlichen Vorgaben gehört zu den häufigsten Aufgabentypen für Programmieranfänger [RBH15]. Es kann dabei variiert werden, wie detailliert man vorgibt, was programmiert werden soll.

Es soll eine Abwandlung des Spiels *Breakout* programmiert werden. Dazu bekommen die Lernenden eine kurze Beschreibung des Spiels:

Ein Erdball prallt vom Rand der Bühne und von einem Paddel ab. Er darf den Boden unter dem Paddel nicht berühren – sonst stoppt das Spiel. Mit dem Paddel, das man als Spieler von rechts nach links steuern kann, versucht man den Erdball vom Boden fernzuhalten.

Programmieren eigener Ideen nach textlichen Vorgaben (Text→Code)

Auch wenn man vorgibt, was programmiert werden soll, kann man die Aufgaben so gestalten, dass die Lernenden dennoch ihre eigenen Ideen umsetzen können. An einem Weihnachtsbaum hängen verschiedene Kugeln. Diese sollen so programmiert werden, dass jede Kugel beim Anklicken etwas anderes macht (z.B. Ton abspielen, Farbe ändern, sich bewegen).

5 Fazit

Im Beitrag wurde gezeigt, wie man durch variierende Aufgaben den Lernprozess des Programmierens gezielt steuern kann. Dabei handelt es sich keinesfalls um eine strenge Handlungsempfehlung, die Aufgaben sollen vielmehr dazu anregen, den eigenen Blick für das Potenzial unterschiedlicher Aufgabentypen zu schärfen.

Das Feedback der Grundschullehrkräfte, die an der beschriebenen Fortbildungsmaßnahme teilgenommen haben, war durchweg sehr positiv. Es wurde berichtet, dass in vielen Büchern, die sich mit dem Thema *Programmieren für Kinder* auseinandersetzen, nur Aufgaben bzw. Anleitungen zu finden sind, in denen Programme schrittweise nachgebaut werden müssen. Sich damit auseinanderzusetzen, wann dieses Kopieren von Code sinnvoll sein kann und wie man die Schülerinnen und Schüler darüber hinaus auf dem Weg zum selbstständigen Programmieren unterstützen kann, wurde von den Lehrkräften als sehr hilfreich eingeschätzt. Das Programmieren im Kontext der Grundschule ist für die Lehrkräfte eine sehr neue Thematik und sie können noch nicht auf einen vergleichbaren Aufgabenpool wie in anderen Fächern zurückgreifen. Wir halten es deshalb für umso wichtiger, sie von Beginn an mit den unterschiedlichsten Aufgabenstellungen vertraut zu machen und sie zu befähigen, Aufgaben gezielt und bewusst einzusetzen.

Literatur

- [BHP14] Brichzin, P.; Humbert, L.; Puhmann, H.: Aufgabenkultur im Schulfach Informatik. LOG IN 34/1, S. 27–32, 2014.
- [Bo08] Bower, M.: A Taxonomy of Task Types in Computing. In: Proceedings of the 45th annual Design Automation Conference. ACM Press, New York, S. 281–285, 2008.
- [GTH18] Geldreich, K.; Talbot, M.; Hubwieser, P.: Off to new shores: Preparing Primary School Teachers for Teaching Algorithmics and Programming. In (Mühling, A.; Cutts, Q., Hrsg.): Proceedings of the 13th Workshop in Primary and Secondary Computing Education. ACM Press, New York, USA, S. 1–6, 2018.

- [HB14] Humbert, L.; Brichzin, P.: Unterrichtskultur und Aufgaben im Informatikunterricht. *LOG IN* 34/1, S. 45–53, 2014.
- [HM06] Hassinen, M.; Mäyrä, H.: Learning Programming by Programming: a Case Study. In (Berglund, A.; Wigberg, M., Hrsg.): *Proceedings KolliCalling 2006*. S. 117–119, 2006.
- [Hu13] Hubwieser, P.; Berges, M.; Magenheim, J.; Schaper, N.; Bröker, K.; Margarithis, M.; Schubert, S.; Ohrndorf, L.: Pedagogical content knowledge for computer science in German teacher education curricula. In (Caspersen, M. E.; Knobelsdorf, M.; Romeike, R., Hrsg.): *Proceedings of the 8th Workshop on Primary and Secondary Computing Education*. ACM Press, New York, USA, S. 95–103, 2013.
- [Le11] Lee, I.; Martin, F.; Denner, J.; Coulter, B.; Allan, W.; Erickson, J.; Malyn-Smith, J.; Werner, L.: Computational thinking for youth in practice. *ACM Inroads* 2/1, S. 32, 2011.
- [Ma10] Maier, U.; Kleinknecht, M.; Metz, K.; Bohl, T.: Ein allgemeindidaktisches Kategoriensystem zur Analyse des kognitiven Potenzials von Aufgaben. *Beiträge zur Lehrerinnen- und Lehrerbildung* 28/1, S. 84–96, 2010.
- [MR02] Milne, I.; Rowe, G.: Difficulties in Learning and Teaching Programming — Views of Students and Tutors. *Education and Information Technologies* 7/1, S. 55–66, 2002.
- [RBH13] Ruf, A.; Berges, M.; Hubwieser, P.: Types of assignments for novice programmers. In (Caspersen, M. E.; Knobelsdorf, M.; Romeike, R., Hrsg.): *Proceedings of the 8th Workshop on Primary and Secondary Computing Education*. ACM Press, New York, USA, S. 43–44, 2013.
- [RBH15] Ruf, A.; Berges, M.; Hubwieser, P.: Classification of Programming Tasks According to Required Skills and Knowledge Representation. In (Brodnik, A.; Vahrenhold, J., Hrsg.): *Informatics in Schools*. Springer, Heidelberg, S. 57–68, 2015.
- [Ri05] Rieck, K.: *Gute Aufgaben*. IPN, Kiel, 2005.
- [Sc06] Schnotz, W.: *Pädagogische Psychologie: Workbook*. Beltz, PVU, Weinheim, Basel, 2006.
- [SW17] Sentance, S.; Waite, J.: PRIMM: Exploring pedagogical approaches for teaching text-based programming in school. In (Barendsen, E.; Hubwieser, P., Hrsg.): *Proceedings of the 12th Workshop on Primary and Secondary Computing Education*. ACM Press, New York, USA, 2017.
- [Va96] Van Lier, L.: *Interaction in the Language Curriculum - Awareness, Autonomy and Authenticity*. Longman, Harlow, 1996.

Informatik spannend präsentieren – Ein Einstieg in den Studiengang Lehramt Informatik

Lisa Göbel¹, Lutz Hellmig²

Abstract: Studiengänge für das Lehramt Informatik verzeichnen seit Jahren eine zu hohe Schwundquote. Als eine der Ursachen wird der mangelnde Praxisbezug besonders am Anfang des Studiums angesehen. Da Studienanfänger die erforderlichen Kompetenzen für die Erteilung von Unterricht überwiegend in späteren Semestern erwerben, müssen andere Formen didaktischer Praxis für die Studieneingangsphase gefunden werden. Im Artikel wird über die Entwicklung, Durchführung und Evaluation eines Pilotprojektes berichtet, in dem sich vor allem Studierende des ersten Semesters informatische Sachverhalte selbst erschließen und sie für die Präsentation in einem populärwissenschaftlichen Format didaktisch aufbereiten.

Keywords: Praxisbezug; Studieneinstieg; Lehramt Studium; Informatik; Studienabbruch

1 Motivation und Zielsetzung

An der Universität Rostock wurden Untersuchungen zum „Studienerfolg und -misserfolg im Lehramtsstudium“ durchgeführt. Die Ergebnisse der Studie zeigen, dass im Lehramtsstudium Informatik nur 5% aller Studierenden das Studium erfolgreich in Regelstudienzeit absolvieren. 10% der Studierenden studieren nach der Regelstudienzeit weiterhin Lehramt Informatik. 85% der Studierenden brechen das Studium ohne Abschluss ab. Die Schwundquote ist nach dem zweiten Fachsemester besonders hoch [Ra18].

Eine mögliche Ursache für die zu hohe Schwundquote könnte der geringe Praxisbezug in den ersten Studiensemestern sein. Die Studierenden belegen erst im vierten Semester die Grundvorlesung der Didaktik des Informatikunterrichts. Die Schulpraktische Übung findet im fünften oder sechsten Semester statt. Bis zum vierten Semester besteht das Studium ausschließlich aus Modulen der Fachwissenschaft. Diese Module entstammen fast ausnahmslos dem Bachelor-Studiengang Informatik.

Ramm et al. erkennen einen Zusammenhang zwischen dem Praxisbezug im Lehramtsstudium und dem erfolgreichen Abschluss. Insbesondere kann ein verbesserter Praxisbezug die zu hohe Schwundquote mindern. [Ra98]

¹ Universität Rostock, Institut für Informatik, Albert-Einstein Straße 22, 18051 Rostock, Deutschland lisa.gobel@uni-rostock.de

² Universität Rostock, Institut für Informatik, Albert-Einstein Straße 22, 18051 Rostock, Deutschland lutz.hellmig@uni-rostock.de

Die Schaffung eines frühen authentischen Praxisbezugs stellt aus mehreren Gründen eine gewisse Herausforderung dar. Die Studierenden haben zu Beginn des Studiums nur wenig fachliche sowie noch keine didaktischen Kompetenzen erworben. Damit kommt das Unterrichten in einer Klasse im ersten Semester nicht als Mittel des Praxisbezugs in Frage. Abgesehen von dem Recht der Schülerinnen und Schüler auf guten Unterricht, das eine solche Maßnahme per se ausschließt, spricht noch ein weiterer wichtiger Grund dagegen: Studierende des ersten Fachsemesters wären mit der komplexen Aufgabe der Planung und Durchführung von Fachunterricht überfordert. Die sich daraus ergebenden negativen Erfahrungen der Studierenden mit der Schulpraxis wirken sich nachhaltig auf die Motivation der Studierenden aus.

Mit der Entwicklung eines Pilotprojektes sollten ein früher Praxisbezug und ein soziales Miteinander unter Umgehung der Risiken hergestellt werden. In Zusammenarbeit mit dem Zentrum für Lehrerbildung und Bildungsforschung sowie mit Unterstützung des Prorektors für Studium und Lehre wurde das Projekt „Informatik spannend präsentieren“ entwickelt.

2 Entwicklung des Pilotprojektes

Das Durchlaufen der wichtigsten Schritte des Planungsprozesses für ein Bildungsangebot ist ein naheliegendes Szenario für die Herstellung eines Praxisbezugs. Die Studierenden arbeiten im Bereich des fachlichen und fachdidaktischen Professionswissens und damit in zentralen Handlungsfeldern von Lehrpersonen [BK11]. Zum pädagogischen Planungsprozess gehören die Bestimmung der zu entwickelnden Kompetenzen, die Analyse der Rahmenbedingungen, eine Sachanalyse, didaktische sowie methodische Überlegungen und die Bestimmung der Feinziele.

Eine Schauvorlesung mit populärwissenschaftlichem Charakter trägt diesen Anforderungen Rechnung. Sie ist weniger komplex als die Gestaltung von Unterricht, da eine Reihe von Anforderungen wie die Einbettung einer Stunde in die Stoffeinheit oder die Berücksichtigung spezieller Attribute der Lerngruppe keine Rolle spielt. Lernziele im engeren Sinne müssen ebenfalls nicht formuliert werden.

Die Vorgabe eines organisatorischen und inhaltlichen Rahmens dient der weiteren Reduktion der von den Studierenden zu klärenden Fragen.

Jede Gruppe bekommt den Auftrag, eine 15- bis 20-minütige Präsentation zu entwickeln und diese am Ende des Semesters im Rahmen einer öffentlichen Abschlussveranstaltung mit Wettbewerbscharakter zu präsentieren. Auf Grundlage der besten drei von fünf im Seminar entwickelten Ideen wird daraufhin eine insgesamt 30-minütige Schauvorlesung entwickelt und auf der Langen Nacht der Wissenschaften präsentiert. Das Publikum besteht aus Kindern im Alter ab 10 Jahren (und ihren Begleitern) mit wissenschaftlichen Interesse, aber ohne tiefere Vorkenntnisse.

Die thematische Schwerpunktsetzung wird durch die Betreuer im Vorfeld mit dem Anspruch vorgenommen, ein möglichst allgemeines informatisches Konzept in verschiedenen, dem Publikum vertrauten Anwendungsbereichen zu vermitteln. Als geeignetes Prüfkriterium erweisen sich in diesem Zusammenhang die Merkmale der fundamentalen Ideen [Br60, Sc93,

HNR06]. Die Strategie der *Minimalisierung* erfüllt mit der Anwendbarkeit auf Datenmenge und Speicherbedarf, den Zeitbedarf für Prozesse oder die Mächtigkeit/Reduktion von Systemen und die mit allen Aspekten verbundenen Kosten das Horizontalkriterium. Aspekte der Minimalisierung sind anschaulich (Repräsentationskriterium) auch Kindern vermittelbar (Vertikalkriterium). Seit jeher werden bei der Konstruktion von Informatiksystemen und der Entwicklung von Modellen – aus Kosten-, Platz- und Zeitgründen – Aspekte der Minimierung beachtet (Zeitkriterium). Die Erfüllung des Sinnkriteriums wird durch eine Reihe praktischer Bezüge augenfällig, wie sie im folgenden Abschnitt bei der Projektrealisierung beschrieben werden.

Für die sich der Grundidee unterordnenden Teilthemen werden Themenvorschläge und Literaturhinweise vorgegeben, eigene Vorschläge der Studierenden waren ebenfalls möglich. In der Regel besitzen die Studierenden nur eine vage inhaltliche Vorstellung über die Themen, so dass die selbstständige Aneignung der fachlichen Grundlagen ein maßgeblicher Bestandteil des Arbeitsprozesses ist.

Um einen motivationalen Anreiz für die Teilnahme am Projekt zu bieten, wird dieses als Wahlpflichtmodul mit 3 Leistungspunkten angeboten.

3 Realisierung des Pilotprojektes

Das Lehramtsstudium an der Universität Rostock beginnt jeweils in der ersten Oktoberwoche. Hier finden sämtliche Einführungsveranstaltungen statt. Anlässlich dieser Veranstaltung wurde das neue Seminar „Informatik spannend präsentieren“ beworben. Das Seminar stand unter dem Motto „Minimalisierung – Weniger ist mehr“. Anhand des Beispiels Entropie als Maß für den Informationsgehalt von Jens Gallenbacher [Ga17] wurde den Studierenden ein Einblick in die Thematik gegeben.

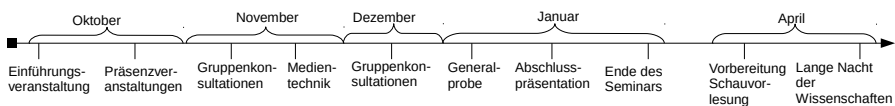


Abb. 1: Grobe Zeiteinteilung des Pilotprojektes

Es haben sich 14 Studierende – davon zwei Studierende des fünften Semesters und ein Student aus dem neunten Semester – für dieses Seminar entschieden. Von den elf Teilnehmern aus dem ersten Semester waren fünf Studierende aus dem gymnasialen und sechs aus dem regionalen Lehramt.

Im Rahmen des Seminars fanden sechs Präsenzveranstaltungen für alle Teilnehmer sowie vier Konsultationen für die einzelnen Gruppen statt (siehe Abb. 1).

In den Präsenzveranstaltungen wurden organisatorische Absprachen getroffen, ein inhaltlicher Zugang zum Thema hergestellt, gemeinsam getragene Bewertungskriterien erarbeitet sowie didaktisch-methodische und technische Anregungen zur Aufbereitung und Darstellung des Themas gegeben. In der letzten Präsenzveranstaltung fand eine Abschlussreflexion der

Präsentationen statt.

Einen ersten inhaltlichen Zugang zu der Idee der Minimalisierung haben die Studierenden über ein Placemat gefunden (siehe Abb. 2).

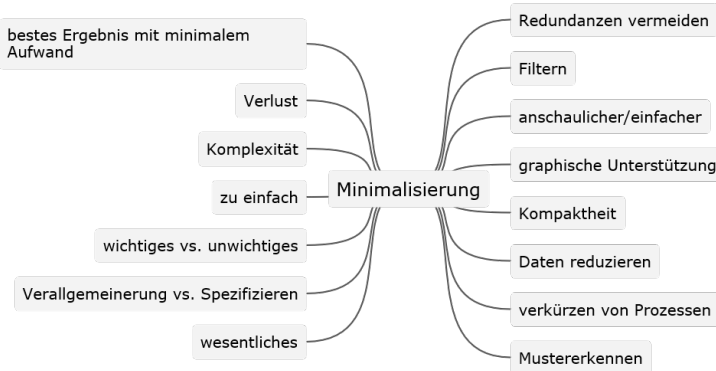


Abb. 2: Mindmap zum Thema Minimalisierung – Ergebnis des Placemats im Seminar

Zur Bearbeitung in Gruppen wurden sechs Themenbereiche vorgestellt. Auf der Vorschlagsliste standen die Minimalisierung von Automaten, verschiedene Sortier- und Suchverfahren, die Optimierung von Netzwerken, die Datenkomprimierung sowie die Minimalisierung von Datenbanken. Zu den Themen haben die Studierenden jeweils Literaturempfehlungen [BWF15, Ga17] erhalten. Das Thema „Optimierung von Netzwerken“ wurde nicht gewählt. Eigene Themenvorstellungen wurden nicht eingebracht. Bis auf eine Zweiergruppe wurden die Themen von Dreiergruppen bearbeitet. Dank der kleinen Gruppen war es den Studierenden trotz unterschiedlicher Zweitfächer möglich, Termine für Treffen außerhalb des Seminars zu vereinbaren. Bei zu großen Gruppen wäre es zu Terminfindungsproblemen gekommen.

Weiterhin haben die Teilnehmer Bewertungskriterien erarbeitet (siehe Abb. 3), die bei der Erstellung der Präsentationen beachtet werden mussten. Die Bewertung der Präsentationen auf der Abschlussveranstaltung wurde durch die Jury anhand dieser Kriterien vorgenommen.

Im Weiteren haben die Gruppen eine konkrete Sachanalyse zu den von ihnen gewählten Themen vorgenommen.

Auf Basis einer erstellten Sachanalyse in den Gruppen wurden die im Vorfeld entworfenen Ideen in Gruppenkonsultationen mit den Betreuern diskutiert, fokussiert und Möglichkeiten einer geeigneten Strukturierung und ansprechenden Darstellung besprochen. Mit dem Nachdenken über mögliche Darstellungsformen ergaben sich kontinuierliche Anlässe für eine wiederholte und vertiefte Sachanalyse. Die Aufgabe, Drehbücher zu erstellen, zwang die Studierenden ihre Präsentationen gründlich zu planen und detailliert auszuarbeiten.

Um die Präsentationen auf die örtlichen Gegebenheiten anzupassen, wurde die Medientechnik im Hörsaal sowie ihre Funktionsweise durch die Teilnehmer erkundet.

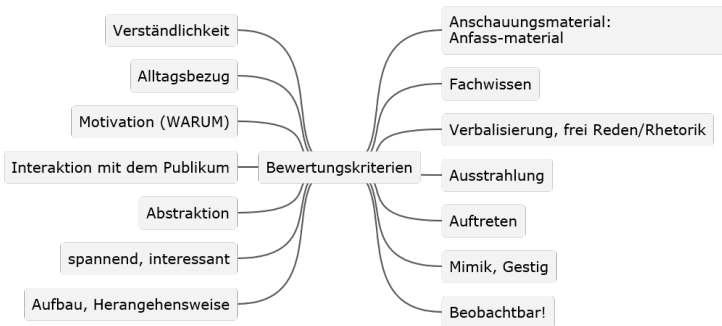


Abb. 3: Mindmap zu den Bewertungskriterien – erstellt im Seminar

Eine Generalprobe vor allen Kursteilnehmern gab den Studierenden die Möglichkeit ihre Präsentation zu üben. Wichtig war das Üben eines zügigen Auf- und Abbaus, sodass bei der Abschlusspräsentation ein schneller Gruppenwechsel gewährleistet wurde. Die Gruppen haben sich gegenseitig Tipps für die Vorstellung gegeben.

Auf der Abschlussveranstaltung vor interessierten Studierenden und Mitarbeitern hat eine zehnköpfige Jury aus Studierenden und Mitarbeitern die einzelnen Gruppen bewertet. Die Präsentationen der Gruppen wurden durch eine Moderation zu einer zusammenhängenden Vorstellung verbunden.

Der erste Beitrag wurde von der Gruppe mit dem Thema Datenbanken vorgestellt. Die Gruppe hat im Vorfeld eine Tabelle zu bekannten Youtubern erstellt (siehe Abb. 4). Die Tabelle beinhaltet die Kategorien Kanalname, Genre, Videoanzahl, Abonnenten, Aufrufe, Sprache, Beitrittsjahr, Vor- und Nachname. Es wurden verschiedene Kanäle vom simpleclub aufgelistet. Bei der Erstellung der Tabelle hat die Gruppe absichtlich den Namen der simpleclub-Youtuber falsch aufgelistet. Ein Kandidat aus dem Publikum sollte diesen Fehler beheben. Da das Auffinden, Löschen und Ersetzen sehr aufwendig ist, wurde diese Tabelle zu einer normalisierten Datenbank umgewandelt. Mit diesem Modell werden Redundanzen beseitigt, Inkonsistenzen vermieden und das Modifizieren von Daten vereinfacht.

Kanalname	Genre	Videoanzahl	Abonnenten	Aufrufe	Sprache	Beitrittsjahr	Vorname	Nachname
Biologie-simpleclub	Bildung	321	469 695	73 621 612	Deutsch	2014	Nick	Schock
Informatik-simpleclub	Bildung	106	117 282	5 126 745	Deutsch	2016	Nick	Schock
Geschichte-simpleclub	Bildung	118	179 737	11 831 077	Deutsch	2014	Nick	Schock
Geographie-simpleclub	Bildung	110	129 740	8 546 469	Deutsch	2014	Nick	Schock

Abb. 4: Ausschnitt der Tabelle zu bekannten Youtubern

Der folgende Beitrag wurde unter dem Motto „Maschinen sind einfach“ präsentiert. Die Gruppe hat eine Geschichte verfasst, in der ein Automat entwickelt wurde (Idee aus [BWF15]). Der Tolper-Troll Willi lebte in Fairytail Junction (siehe Abb. 5) und ist auf der Suche nach dem Zimtberg. Es gibt keinen direkten Weg, um von Willis Standpunkt

zum Zimtberg zu kommen. Daher muss er von Ort zu Ort wandern. Von jedem Ort aus gehen zwei Wege ab. Während die Geschichte vorgetragen wurde, entschied das Publikum, welchen Weg Willi als nächstes wählt. Die entstandene Landkarte wurde während der Geschichte mit Hilfe der Dokumentenkamera gezeigt. Nachdem die Geschichte beendet war, hat die Gruppe einen komplizierten Automaten gezeigt, welcher zu einem einfachen, aber gleichmächtigen Automaten reduziert wurde.

FAIRYTALE JUNCTION

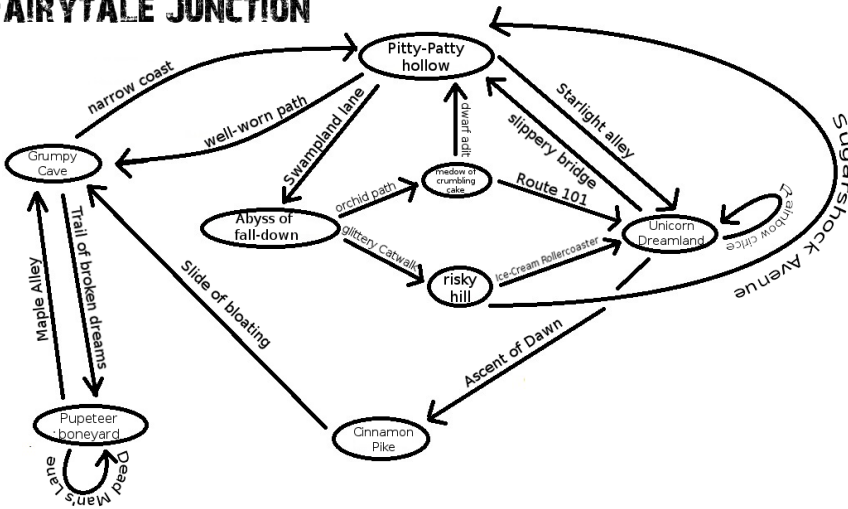


Abb. 5: Fairytail Junction

„Das Genie beherrscht das Chaos.“ Unter diesem Motto stand der dritte Beitrag. Zunächst sollten zwei Personen des Publikums ein unsortiertes Kartenspiel nach einer vorgegebenen Reihenfolge der Karten sortieren. Dabei wurde die schnellere Person nach ihrer Vorgehensweise befragt. Anschließend wurde ein Wettstreit zwischen zwei Gruppenmitgliedern präsentiert. Es wurde behauptet, dass Bubblesort ein schnelleres Sortierverfahren als Quicksort sei. Um diese Behauptung zu beweisen oder zu widerlegen, wurden jeweils sieben Personen des Publikums aufgefordert, die Algorithmen durchzuspielen. Es wurden verschiedene Getränke nach ihrem Alkoholgehalt aufsteigend sortiert. Die Anzahl der einzelnen Schritte wurden jeweils gezählt. Dadurch hat sich herausgestellt, dass Quicksort das schnellere Sortierverfahren ist.

Der folgende Beitrag knüpfte an das Sortieren an, denn in vorsortierten Listen kann schneller gesucht werden. Zunächst wurden zahlreiche Beispiele aus dem Alltag zum Thema Suchen präsentiert. Anschließend wurden die lineare sowie die binäre Suche anhand der Suche nach einem Filmtitel durchgeführt. Im Internet werden sehr viele Suchanfragen in kürzester Zeit ausgewertet. Doch wie funktioniert das? Dazu hatte die Gruppe ein Crawler-Spiel vorbereitet. Es wurde nach einem bestimmten Kuchenrezept gesucht. Die einzelnen Rezeptseiten waren als „Webseite auf Papier“ im Raum verteilt. Die Stichwörter

waren bereits farbig markiert. Am Ende der Seite gab es Abreißzettel mit der URL der Seite. Für jedes Stichwort stand ein Korb auf der Bühne zur Verfügung. In diesem Korb wurden die jeweiligen URL eingeworfen. Nach der Sortierung kann je nach Suchanfrage der entsprechende Korb ausgeschüttet werden. Dieses Crawler-Spiel wurde von sieben Personen des Publikums durchgeführt. Währenddessen wurde Musik eingespielt, in deren Text das Thema Suchen eine Rolle spielte.

Der Abschluss der Vorstellung wurde durch die Datenkomprimierung gebildet. Die Gruppe hatte ein Faxgerät als Anschauungsmaterial. Damit sollte eine Geburtstagseinladung versendet werden. Doch wie funktioniert dieses Gerät? Mit Hilfe der Lauflängenkodierung wurde am Beispiel eines Buchstabens (siehe Abb. 6) die Übertragung enaktiv präsentiert. Zwei Personen des Publikums haben die Übertragung durchgespielt. Eine Person musste die Zeichen kodieren und die andere Person war für die Übertragung zuständig. Zunächst durften nur die Worte schwarz und weiß übertragen werden. Diese Übertragung hat sehr lange gedauert. Aufgrund dessen wurde eine andere Übertragung festgelegt. Nun durften Zahlen übertragen werden. Es wurde vereinbart, dass jede neue Zeile zuerst die Anzahl der schwarzen Pixel überträgt (Idee von [BWF15]). Abschließend wurde besprochen, durch welche Möglichkeit die Buchstaben c und e schnell hintereinander übertragen werden können. Dabei wurde herauskristallisiert, dass lediglich die Veränderung übertragen wird.

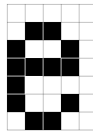


Abb. 6: Mit Lauflängenkodierung zu übertragender Buchstabe

Aus den besten drei Beiträgen – Automaten, Sortier- und Suchverfahren – wurde zu Beginn des neuen Semesters eine gemeinsame Präsentation als Schauvorlesung für die Lange Nacht der Wissenschaften erstellt. In fünf Konsultationen wurden die Beiträge auf insgesamt 30 Minuten reduziert. Bei der Modifikation der Präsentationen ist aufgefallen, dass einige Gruppen die Themen und die Idee der Minimalisierung nach wie vor nicht vollständig durchdrungen haben. Daher wurden die fachlichen Grundlagen nochmals grundlegend besprochen.

Die Schauvorlesung auf der Langen Nacht der Wissenschaften wurde vom Publikum sehr gut angenommen. Das Publikum bestand vorwiegend aus Kindern mit ihren Eltern sowie interessierte Schülerinnen und Schüler.

4 Evaluation

Das Projekt wurde mit Hilfe von zwei Fragebögen evaluiert. Der erste Fragebogen wurde im ersten Seminar von 13 Studierenden ausgefüllt und beinhaltete ausschließlich offene Fragen. Die Ergebnisse geben Aufschluss über die Motivation und die Ziele der Teilnehmer. Die Resultate zeigen, dass 7 von 13 Studierenden aufgrund der Werbeveranstaltung in der

Einführungswoche teilgenommen haben. Obwohl dieses Seminar zuvor noch nicht angeboten wurde, wurde den Erstsemestern die Teilnahme an so einer Veranstaltung von älteren Semestern empfohlen. Das Seminar wurde besucht, da der Fokus auf die Lehramtstätigkeit ausgerichtet ist. Das Knüpfen von sozialen Kontakten ist ein wichtiger Punkt für die Teilnehmer. Als weiteren Grund führen einige Teilnehmer den Erhalt von Leistungspunkten, das Sprechen vor Publikum und den Versuch, Informatik ansprechend zu vermitteln, an.

Die zweite Frage des Fragebogens zielte auf die Erwartungen und die Bedenken ab. Die meisten Befragten gaben an, keine Bedenken zu haben. Einige äußerten die Befürchtung möglicher Ideenlosigkeit, kreativer Überforderung oder zu hoher fachlicher Anforderungen. Die Studierenden erwarteten einen kreativen Austausch von Fachwissen, didaktischen und methodischen Aspekten. Die Teilnehmer erhofften ein lockeres Umfeld und eine positive Zusammenarbeit mit den Dozenten. Von den Dozenten wurde erwartet, dass sie selbst das Seminar spannend präsentieren.

Der zweite Fragebogen untersuchte sechs Kategorien. Die erste Frage war offen gestellt und lieferte Auskunft über die Erfüllung der Erwartungen. Die zweite und dritte Frage war siebenstufige Single-Choice-Fragen. Hier wurden die Komponenten der Lehrveranstaltung sowie die Seiteneffekte evaluiert. Die vierte Frage war offen formuliert und gibt Aufschluss über eine Weiterempfehlung des Seminars. Die folgende offene Frage erhebt Anregungen und einen Verbesserungshinweise für eine weitere Durchführung des Seminars. Im letzten Bereich hatten die Studierenden die Möglichkeit aufzuschreiben, was sie unbedingt noch sagen möchten.

Dieser Fragebogen gibt Aufschluss über die Entwicklung von professioneller Kompetenz [BK11]. Es werden die Aspekte motivationale Orientierung, Selbstregulation, Ziele sowie Professionswissen, vor allem Fachwissen und fachdidaktisches Wissen der Teilnehmer untersucht.

Dieser Fragebogen wurde von 11 Teilnehmern ausgefüllt. Die Studierenden waren während des Projektes sehr motiviert. Die Einstellung zur Berufsmotivation hat sich positiv entwickelt. Das selbstständige Arbeiten wurde von zehn Studierenden positiv bewertet, obwohl dieses einen zusätzlichen Zeitaufwand darstellte. Die Inhalte der Präsenzveranstaltung sowie der Konsultationen waren für zehn von elf Studierenden hilfreich. Sie haben dadurch fachliche Kompetenzen erworben. Der methodische Wechsel zwischen den Präsenzveranstaltungen und den Konsultationen wurde von allen Teilnehmern positiv bewertet. Die Studierenden haben durch das Seminar soziale Bindungen zu ihren Kommilitonen aufgebaut. Acht Teilnehmer lobten die lockere und freundliche Atmosphäre.

Zwei Studierende wünschten mehr Austauschmöglichkeiten mit allen Teilnehmern und vier Studierende eine zeitliche Vorverlegung der Präsenzveranstaltung. Ein Student sah Verbesserungsbedarf im Zeitplan vor Weihnachten. Zwei Studierende wollten mehr Proben im Hörsaal, wo die Abschlussveranstaltung stattfindet. Ein Teilnehmer würde sich über ein größeres Publikum freuen. Des Weiteren würden einige Teilnehmer dieses Seminar für Studierende jeden Semesters empfehlen. Ein Teilnehmer hätte sich Tipps für das freie Sprechen sowie einen engeren thematischen Rahmen gewünscht.

5 Schlussfolgerungen und Ausblick

Die Evaluationsergebnisse lassen erkennen, dass die Studierenden diese Form der didaktischen Praxis positiv beurteilen. Angesichts der überschaubaren Zahl der Teilnehmer des Moduls lassen sich aus den gewonnenen Daten selbstverständlich noch keine gesicherten Erkenntnisse ableiten. Die Erhebung liefert allenfalls Anhaltspunkte für die Formulierung von Hypothesen, die in weiteren Untersuchungen spezifisch untersucht werden sollten.

Die Studierenden waren im Seminar stets interessiert und engagiert. Aufgrund der hohen Zufriedenheit der Studierenden wird das Seminar im nächsten Wintersemester zum zweiten Mal angeboten. Weiterhin bleibt zu untersuchen, ob die teilnehmenden Studierende das Informatik Studium fortsetzen und erfolgreich beenden.

Beim nächsten Durchlauf des Projektes werden Modifikationen vorgenommen. Es wird eine Präsenzveranstaltung eingefügt, die Tipps zum freien Sprechen liefert. Die Präsenzveranstaltung wird in den ersten Wochen wöchentlich stattfinden. Dadurch können die Studierenden zeitiger konkret an ihren Themen arbeiten. Proben zur Präsentation werden früher geplant. Die Studierenden werden dadurch sicherer im Präsentieren und gewinnen Selbstbewusstheit im eigenen Handeln. Weiterhin wird das Seminar auch für Studierende höherer Semester geöffnet, wenngleich der Fokus auch weiterhin auf Studierende des ersten Semesters gelegt werden soll. Somit können erfahrene Studierende mit Studienanfängern zusammenarbeiten. Die Studierenden können auf diese Weise viel voneinander lernen und sich gegenseitig motivieren, gute Informatiklehrerinnen und -lehrer zu werden.

Dieses Pilotprojekt hat den Studierenden einen ersten Praxisbezug gegeben. Dabei hatten die Studierenden die Möglichkeit, ihre Kommilitonen kennenzulernen. Diese Methode des Praxisbezugs hat sich gut für die Informatik geeignet. Dank der Anwendung der fundamentalen Ideen der Informatik haben die Studierenden bemerkt, dass die Informatik ein großes Anwendungsfeld hat und trotz des ständigen Wandels in der Informatik die Grundideen beibehalten bleiben.

Diese Möglichkeit des Studieneinstieges bietet sich vor allem für Fächer mit einem sehr hohen Anteil an Fachwissenschaften in der Anfangsphase des Studiums an. Sofern die fundamentalen Ideen des jeweiligen Faches thematisiert werden, erhalten die Studierenden einen umfassenden Einblick in das Fach. Somit wäre eine Übertragung des Projektes auf andere Fächer möglich.

Literaturverzeichnis

- [BK11] Baumert, Jürgen; Kunter, Mareike, Hrsg. Das Kompetenzmodell von COACTIV. Professionelle Kompetenz von Lehrkräften: Ergebnisse des Forschungsprogramms COACTIV. 2011.
- [Br60] Bruner, J. S., Hrsg. The process of education. Havard University Press, Cambridge, 1960.
- [BWF15] Bell, Tim; Witten, Ian H.; Fellows, Mike, Hrsg. CS unplugged. An enrichment and extension programme for primary-aged students. 2015.

- [Ga17] Gallenbacher, Jens, Hrsg. Abenteuer Informatik. IT zum Anfassen für alle von 9 bis 99 – vom Navi bis Social Media. Springer, Heidelberg, 4. Auflage, 2017.
- [HNR06] Hartmann, Werner; Näf, Michael; Reichert, Raimond, Hrsg. Informatikunterricht planen und durchführen. eXamen.press, Springer-Verlag Berlin Heidelberg, 2006.
- [Ra98] Ramm, M.; Kolbert-Ramm, C.; Bargel, T.; Lind, G., Hrsg. Praxisbezug im Lehramtsstudium. Erfahrungen und Beurteilungen der Lehramtsstudierenden. Konstanz: Universität Konstanz, Arbeitsgruppe Hochschulforschung, 1998.
- [Ra18] Radisch, Falk; Driesner, Ivonne; Arndt, Mona; Güldener, Torben; Czapowski, Janek; Petry, Martin; Seeber, Anne-Marie, Hrsg. Abschlussbericht Studienerfolg und -misserfolg im Lehramtsstudium. Rostock: Universität Rostock, 2018.
- [Sc93] Schwill, A, Hrsg. Fundamentale Ideen der Informatik. Zentralblatt für Didaktik der Mathematik, 25(1), 1993.

Fit für den Lehrplan 21 – Wie Klassenlehrpersonen auf den Informatikunterricht vorbereitet werden (können)

Im Dialog lernen mit kompetenzorientierten Aufgabensets

Andrea Gumpert,¹ Pascal Zaugg²

Abstract: Informatik ist in der Schweizer Primarschule angekommen! Studierende werden an Pädagogischen Hochschulen darauf vorbereitet, Informatik zu unterrichten und in die Fächer zu integrieren. Keine leichte Aufgabe, denn häufig haben sie selbst keinen Informatikunterricht erlebt, was bedeutet, dass die Vorerfahrungen in dem Bereich sehr heterogen sind. Eine etablierte Fachdidaktik für das neue Modul Medien und Informatik gibt es darüber hinaus noch nicht. Um diese Herausforderungen zu meistern, werden Räume für individuelles Lernen und Vertiefen geschaffen. Anhand von kompetenzorientierten Aufgabensets lernen die Studierenden selbstgesteuert, aber begleitet und dialogisch, das, was sie an fachlichem und fachdidaktischem Handwerkszeug für den Medien-und-Informatik-Unterricht brauchen. Durch die individualisierbaren Aufgabensets, welche sich auf das Konzept des Dialogischen Lernens (vgl. u.a. [RG14a]) abstützen, wird das selbstbestimmte, freie Lernen im Präsenzunterricht an der Hochschule unterstützt. Kooperatives und selbstreflexives Lernen an schulpraxisorientierten Aufgaben ermöglicht es den Studierenden zum einen, fachwissenschaftliche Konzepte aufzubauen und zum anderen, fachdidaktische Erfahrungen für das Lehren im Fach Medien und Informatik zu nutzen (didaktischer Doppeldecker).

Keywords: Medien und Informatik; Lehrpersonenbildung; E-Portfolio; Dialogisches Lernen; Kompetenzorientierung

1 Einleitung und Rahmenbedingungen

Seit 2014 ist der Lehrplan 21³, welcher für alle 21 Deutschschweizer Kantone Gültigkeit hat, verabschiedet. Eine Neuerung ist das im Kanton Bern seit 2018 flächendeckend unterrichtete Modul „Medien und Informatik“ (im Folgenden MI). Medienbildung und informatische Bildung werden sowohl integriert in die Fächer (bereits ab der Kindergartenstufe) als auch in eigenen Lektionengefäßen vermittelt.

Im Studium an der Pädagogischen Hochschule Bern werden fachwissenschaftliche, fachdidaktische, erziehungswissenschaftliche und berufsbezogene Kompetenzen aufgebaut. MI

¹ Pädagogische Hochschule Bern, Institut Vorschulstufe und Primarstufe, Fabrikstrasse 8, 3012 Bern, Schweiz
andrea.gumpert@phbern.ch

² Pädagogische Hochschule Bern, Institut Vorschulstufe und Primarstufe, Fabrikstrasse 8, 3012 Bern, Schweiz
pascal.zaugg@phbern.ch

³ <https://be.lehrplan.ch/index.php>

ist für alle Studierenden der Vorschul- und Primarstufe obligatorisch und wird in den zwei Modulen „MI1“ und „MI2“ (gemeinsam 5 ECTS) unterrichtet. Die Studierenden erarbeiten in diesen Gefäßen Kompetenzen aus den Bereichen der Medienpädagogik, Mediendidaktik und Informatik sowie der Informatikdidaktik. Während im MI1 die Medienbildung im Vordergrund steht, fokussiert MI2 auf die Informatik.

Aufgrund des heterogenen Vorwissens der Studierenden sind neben fachdidaktischen Fragestellungen besonders auch Fachinhalte Gegenstand der Lehrveranstaltungen. Dies ist aufgrund des verhältnismäßig geringen Workloads eine große Herausforderung. Die im Folgenden beschriebenen kompetenzorientierten Aufgabensets wurden entwickelt, um Zeit und Raum für individuelles, exemplarisches Lernen und Vertiefen zu schaffen und flexibel auf das eingehen zu können, was die Studierenden an didaktischem und fachlichem Handwerkszeug brauchen. Insbesondere der Aufarbeitung, Überarbeitung und Reflexion von Präkonzepten und des eigenen Lernfortschritts wird ein großer Stellenwert eingeräumt. Ziel ist, dass die Studierenden Gestaltungsaspekte für ihr eigenes Lernen identifizieren und diese Erfahrungen für das Lehren nutzen (didaktischer Doppeldecker). Durch die Einbettung der Aufgabensets in offene Unterrichtssettings wird der Heterogenität der Studierenden, dem großen und dynamischen Stoffumfang und der Kürze der zur Verfügung stehenden Seminarzeit Rechnung getragen. Das zugrundeliegende didaktische Konzept des Dialogischen Lernens (vgl. [RG14a], [RG14b], [RKW08], [RG95]) bietet sich für flexibilisierte Arbeit an und bildet einen Rahmen für die bereits institutionalisierte E-Portfolioarbeit.

2 Theoretische Grundlagen zur Konzeption der kompetenzorientierten Aufgaben

In den konzipierten Aufgaben werden drei didaktische Wirkungslinien verknüpft: Lernen in offenen Kontexten (dialogisch), Kompetenzorientierung und Selbstreflexion im eigenen E-Portfolio. Ziel ist es, diesen drei Ansprüchen nicht nur gerecht zu werden, sondern sie auch so umzusetzen, dass sie sich gegenseitig positiv beeinflussen und ergänzen.

2.1 Dialogisches Lernen

Das Dialogische Lernen stellt ein praxisorientiertes Konzept zur Gestaltung einer neuen Lernkultur dar, welches den gesellschaftlichen Herausforderungen der Individualisierung und Flexibilisierung Rechnung tragen kann. Charakteristisch für diese sogenannte neue Lernkultur [Wi16] ist die Betonung des Lernens vor dem Lehren, die Fokussierung auf den Lernenden statt auf die Inhalte, die Eigenverantwortung sowie die Partizipation des Lernenden im Unterricht und die Verschiebung von der Produktorientierung hin zu einer Orientierung und lernfördernden Beurteilung des Lernprozesses.

Im dialogischen Unterricht steht die offene Auseinandersetzung mit den eigenen Präkonzepten und den Angeboten des Unterrichts im Vordergrund. Studierende bauen auf Vorerfahrungen, Kognitionen und Gefühlen auf und bekommen so einen individuellen, singulären Zugang zur Sache. Lernen wird als Prozess aus drei Elementen beschrieben: Das Bewusstmachen des eigenen Vorwissens („Ich mache das so.“), der Austausch mit den Peers („Wie machst Du es?“) und das Erfahren und die Erarbeitung des „Regulären“ im Dialog mit der Lehrperson und dem Material („Das machen wir ab.“) [RG95].

Zentrales Instrument des Dialogischen Lernens ist die Kernidee, welche das Thema bzw. die Inhalte beschreibt und das repräsentiert, was den „Witz der Sache“ ausmacht. Die Kernidee ist persönlich und individuell und transportiert die Vorstellungen der Lehrperson, die es ermöglichen sollen, den Lernenden Zugang zu einem Inhalt zu verschaffen. Die Kernidee wird von den Studierenden als Ergebnis der Auseinandersetzung mit Inhalten und Methoden formuliert und für den zukünftigen Unterricht an der Schule genutzt. Als solches soll die Kernidee als Brücke zwischen Lehrpersonenbildung und Schule fungieren.

Den Einstieg in eine Thematik bildet ein offener Auftrag, im Rahmen dessen Präkonzepte mit einem Thema schriftlich im E-Portfolio bzw. Lernjournal festhalten werden. Die Lehrperson wählt besonders gelungene, spannende oder interessante und diskutierenswerte Konzepte aus, kommentiert, gibt eine Rückmeldung und spielt diese sogenannten Autographen [RKW08] in den Unterricht zurück. Sie dienen gewissermaßen als Unterrichtsmaterial.

Die Rückmeldungen, welche die Lehrpersonen in regelmäßigen Abständen in den E-Portfolios hinterlässt, würdigen die Arbeiten der Lernenden und heben das Gelungene hervor. Fehlkonzepte werden identifiziert und für das Weiterlernen genutzt. Mit Hilfe der Kommentare oder in Gesprächen erarbeiten die Lernenden die fachlich regulären Konzepte („Das machen wir ab.“). Damit schließt sich der Kreislauf wieder zur Kernidee.

2.2 Kompetenzorientierung

Die Zieldimensionen von Unterricht – ob Schule oder Hochschule – werden als Kompetenzen formuliert. Nach Weinert sind Kompetenzen wie folgt definiert: „Kompetenzen sind die bei Individuen verfügbaren oder durch sie erlernbaren kognitiven Fähigkeiten und Fertigkeiten, um bestimmte Probleme zu lösen, sowie die damit verbundenen motivationalen, volitionalen (Anm: d.h. absichts- und willensbezogenen) und sozialen Bereitschaften und Fähigkeiten, um die Problemlösungen in variablen Situationen erfolgreich und verantwortungsvoll zu nutzen.“ [We01]. Doch wie lassen sich Kompetenzen fördern? Wie muss eine Aufgabe gestaltet werden, die nicht nur Wissen und Können fördert, sondern auch eine Lernumgebung schafft in der die Lernenden die Bereitschaft, ein Problem zu lösen, entwickeln können? Kalcsics und Wilhelm [KW17] konkretisieren die weinertsche Definition und bauen darauf ihr Prozessmodell für Kompetenzorientierte Aufgaben (Abb. 1) auf. Sie beschreiben eine kompetente Person als fähig, „in bestimmten Situationen einen Auftrag zu bearbeiten, eine Herausforderung zu meistern, eine Aufgabe zu lösen“ [KW17] und die Bereitschaft zu

haben, dies auch zu tun. Aufgaben sollen Motivationen und Handlungsbereitschaften – also die Persönlichkeit [Wi18] und die Lebenswelt eines Menschen – produktiv einbeziehen. Dies wird durch Lernumgebungen ermöglicht, „welche die Lernenden in eine intensive, aktive, selbst gesteuerte kooperative Auseinandersetzung mit dem Lerngegenstand bringen“ [Le11], das Vorwissen berücksichtigen, authentisch und individualisierbar sind. In einer produktiven, kooperativen und diskursiven Art und Weise werden Lernprodukte selbstständig erarbeitet [Le11] und das eigene Lernen reflektiert. Hierbei werden die zu entwickelnden Kompetenzen direkt genutzt, sind somit relevant und in einen Anwendungskontext eingebunden.

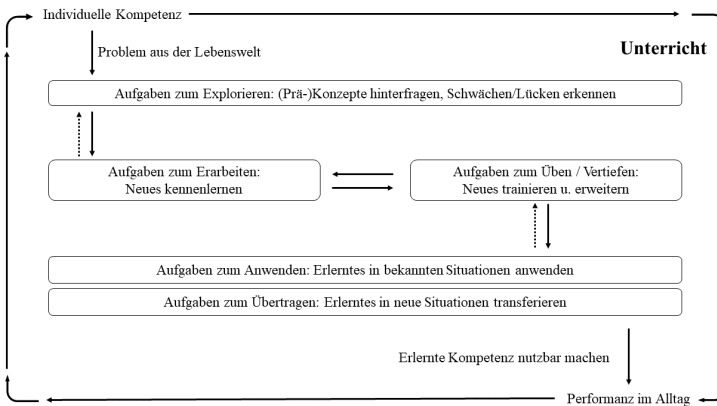


Abb. 1: Prozessmodell kompetenzfördernder Aufgabensets (in Anlehnung an [WLS16])

Iterativ werden Teilkompetenzen bzw. Kompetenzaspekte zu komplexeren Kompetenzen weiterentwickelt. Dabei sollen die Lernaufgaben die notwendigen Impulse geben, ohne den Lernprozess zu stark vorzustrukturieren [KW17].

2.3 E-Portfolioarbeit und reflexives Lernen

Beim Schreiben wird das Denken verlangsamt und Lernen wird bewusst wahrgenommen. Die positive Beeinflussung der Kompetenzentwicklung durch (selbst-) reflexives Lernen [Je08] wird sowohl im dialogischen Lernen als auch im Prozessmodell für kompetenzorientierte Aufgaben gezielt betont. Offermanns definiert Selbstreflexion als „das Auseinandersetzen mit den für einen selbst wichtigen eigenen subjektiven Deutungen (z. B. Gedanken oder Motive), den damit verbundenen Gefühlen sowie den daraus resultierenden Handlungen und deren Konsequenzen unter Berücksichtigung des Verhaltens und der subjektiven Deutungen anderer Personen sowie der bestehenden Strukturen (Regeln, Aufgaben, Umweltbedingungen), die einen umgeben.“ [Of04]. Will man die Potenziale der Selbstreflexion in seiner Komplexität lernfördernd ausnutzen, so muss das Anfertigen eines Lernjournals im E-Portfolio so begleitet und gerahmt werden, dass Schreiben nicht zur Alibiübung wird, welche die Studierenden nur zum Selbstzweck machen. Sowohl die Formulierung

der Aufträge als auch die didaktische Konzeption des dialogischen Unterrichts motivieren Lernende, sich mit ihren subjektiven Deutungen, den Gedanken und Gefühlen hinsichtlich ihrer eigenen Konzepte auseinanderzusetzen.

3 Konzeption der kompetenzorientierten Aufgaben und vorliegende Erfahrungen

3.1 Aufgabenkonzeption und Struktur

Zu sechs relevanten Themen der Informatik werden Aufgabensets konzipiert, welche aus fünf offenen Einzelaufträgen bestehen. Nach dem Prozessmodell für kompetenzfördernde Aufgabensets gliedern sich die Aufgabenabschnitte in Aufträge zum Explorieren (Präkonzepterhebung, Überfachliches), Erarbeiten (Wissensaneignung), Üben und Vertiefen (methodische Vertiefung), Anwenden (produktives Arbeiten) und zum Reflektieren. Abschließend erfolgt ein Peer-to-Peer-Feedback auf das Lerntagebuch des Partners oder der Partnerin.

Alle Aufträge sind gleich strukturiert, wie Abb. 2 zeigt. Klare Zielformulierungen („Was ist Ziel des Auftrages?“) und konkrete Kriterien („Worauf kommt es an?“) helfen den Studierenden, sich zu orientieren. Mit dem fachdidaktischen Hinweis wird erreicht, dass die Studierenden sich nicht nur als Lernende der Fachinhalte sehen, sondern auch die Schule und die Anwendung ihres Wissens im Unterricht vor Augen haben.

Explorieren: Wie der Ton in den Computer und wieder heraus gelangt

Was ist das Ziel des Auftrages?

Das Vorwissen soll aktiviert und die eigenen Vorstellungen explizit gemacht werden. Eigene Konzepte und Handlungsweisen sollen an einer konkreten Situation geprüft und hinterfragt werden.

Dabei soll ein Austausch stattfinden, um die Vorgänge bei der Digitalisierung unter verschiedenen Perspektiven sehen zu können und ggf. Fehlkonzepte aufzudecken. Diesen Auftrag kannst Du also gemeinsam mit einem Partner oder einer Partnerin erledigen.

Was ist zu tun?

Ihr müchtet eine Tonsequenz, also eine Melodie, etwas Gesprochenes oder ein Geräusch mit Hilfe des Computers bearbeiten und dann wieder abspielen.

Was denkt Ihr, was mit dem Ton passiert, wenn er in den Computer aufgenommen, verarbeitet und abgespielt wird?

Erstellt gemeinsam eine Übersicht, in der Ihr den Weg eines Tons in den Computer, die Bearbeitung und das Abspielen schematisch darstellt. Berücksichtigt dabei auch alles, was Ihr schon über die Darstellung von Ton und über Dateiformate wisst.

Falls Ihr Anregungen benötigt, welche Werkzeuge Ihr zur Erstellung von Übersichten brauchen könnt, schaut doch [hier](#) einmal nach. Diese Übersicht wird

Worauf kommt es an?

- Strukturierte, detaillierte und deutliche Darstellung durch adäquate Auswahl des Werkzeuges
- Nachvollziehbarkeit der Erarbeitung und der Denkwege (Darstellung des eigenen, in anderen Lehrveranstaltungen erarbeiteten, Vorwissens/Begriffe)
- Explikation von eigenen Fragestellungen und Unsicherheiten

Fachdidaktischer Hinweis

Damit Kinder von der Umwandlung von Ton überhaupt ein individuelles Konzept entwickeln können, welches sich verändern (conceptual change) oder erweitern

Abb. 2: Screenshot der Ansicht eines Auftrages (zur besseren Lesbarkeit leicht manipuliert)

Im SWITCHportfolio werden die Aufgabenkomplexe als Sammlung umgesetzt, wobei sich die Einzelaufträge auf Ansichten befinden. Diese Anordnung ermöglicht aber auch, die Reihenfolge der Bearbeitung zu ändern oder Arbeitsschritte zu überspringen, etwa wenn zu Themen oder Methoden bereits Kompetenzen in anderen Modulen erarbeitet wurden. Im Präsenzunterricht wird dies durch den Dialog mit der Lehrperson begleitet. Da die Aufgaben das benötigte Material enthalten und unabhängig vom Präsenzunterricht sind,

können sie auch im Selbststudium erarbeitet werden. Es ist allerdings im Sinne des Lernens als sozialen Prozess empfehlenswert und sinnvoll, kooperativ zu arbeiten.

3.2 Themenübersicht

Auf Basis des Lehrplans 21 wurden sechs inhaltliche Schwerpunkte definiert. Der Bereich der Algorithmen erhält dabei ein leichtes Übergewicht. Computational Thinking wird als fachdidaktisches Konzept häufig aufgegriffen und der allgemeinbildende Stellenwert der Informatik betont. Weiterhin wurde darauf Wert gelegt, dass die drei Schwerpunkte „Datenstrukturen“, „Algorithmen“ und „Informatiksysteme“, wie sie im Lehrplan 21 formuliert sind, nicht isoliert voneinander behandelt werden, sondern die Verbindungen zwischen den Kompetenzbereichen lebensweltnah aufgezeigt werden.

Es wurden folgende Aufgaben entwickelt:

Digitalisieren - Vom Analogen zum Digitalen und zurück⁴: Zuerst zeichnen und beschreiben die Studierenden wie sie sich das Digitalisieren von Ton vorstellen und diskutieren dies mit einer Partnerin bzw. einem Partner. Nachdem sie einen Lernfilm zum Thema rezipiert haben, überarbeiten sie ihre Zeichnung. Nun stellen sie ihren Namen im ASCII Code dar und beschäftigen sich mit der Umwandlung eines Bildes in Binärcode beim Scannen. Reflexion und Kernidee bildet den Abschluss der Aufgabe.

Suchen, Finden, Sortieren, Ordnen - So macht's der Computer⁵: Um an ihre eigenen Erfahrungen anzuknüpfen, sollen die Studierenden beschreiben, wann, wo und wie sie sortieren. Mittels Scratch wird der Bubble-Sort-Algorithmus visualisiert und die Studierenden sollen dies in ein Flussdiagramm übersetzen. Anschließend entwickeln sie eine Unterrichtseinheit zum Sortieren und reflektieren das Thema und ihren Lernprozess wiederum anhand von Alltagsphänomenen (Suchmaschinen / Onlinehandel).

Mensch und Maschine - Arbeitsteilung durch Automatisierung⁶: Einführend setzen sich die Studierenden mit den gesellschaftlichen Folgen der Automatisierung auseinander. Nach einigen anwendungskompetenzaufbauenden Übungen mit Scratch programmieren sie einen Rasenmäherroboter, optional nutzen sie dafür das LEIS⁷. Abschließend erfolgt wiederum ein Transfer in Form eines Gedankenexperiments.

Internet - Wo sind meine Daten?⁸: Zu Beginn wird das Vorwissen zum Internet erhoben und Materialien zur Recherche bereitgestellt. Es wird wieder auf das Präkonzept Bezug

⁴ <https://tinyurl.com/MI-KoA-digitalisierung>

⁵ <https://tinyurl.com/MI-KoA-sortieren>

⁶ <https://tinyurl.com/MI-KoA-menschmaschine>

⁷ Lego Education Innovation Center an der PH Bern

⁸ <https://tinyurl.com/MI-KoA-internet>

genommen und das Wissen im Bereich der Netzwerktopologien vertieft. Als Anwendungsaufgabe sollen die Studierenden eine Unterrichtseinheit zum Internet beurteilen und anschließend fachinhaltlich und fachdidaktisch reflektieren.

Mein Computer funktioniert nicht mehr! - Woran liegt's, was kann ich tun⁹: Zunächst notieren die Studierenden ihre Vorstellungen vom Weg des Buchstabens A auf den Bildschirm. Mit Material und einem aufgeschraubten PC vertiefen sie ihr Wissen rund um Informatiksysteme und entwerfen eine eigene Kurzgeschichte für Kinder. Mit der Software Calligraphr¹⁰ entwerfen sie ihre eigene Schriftart, vertauschen bewusst das A und das B und überdenken so die Funktionsweise und das Zusammenspiel von Betriebssystem und Anwendungssoftware bei der Übermittlung eines Buchstabens auf den Bildschirm. Weiterhin beschäftigen sie sich mit der Prozessorauslastung ihres eigenen PC und überlegen, wann und ob sich die Anschaffung eines neuen Gerätes lohnen würde und warum. Reflektierend fassen sie die Inhalte zusammen und stellen einen Schulbezug her.

xLOGO unterrichten¹¹: Ausgehend von der eigenen Vorstellung von gutem Unterricht erarbeiten die Studierenden didaktische Prinzipien zum Einsatz der Programmiersprache Logo im Programmierunterricht. Dazu bauen sie zunächst Erfahrungen mit der Sprache auf und setzen sich dann vertieft mit dem Konzept des Computational Thinking und den fachdidaktischen Wirkungslinien für alle Bereiche der Informatik auseinander.

3.3 Erfahrungen am Beispiel des Aufgabensets zum Thema Digitalisierung

Das Aufgabenset zum Thema „Digitalisieren - Vom Analogen zum Digitalen und zurück“ konnte bereits erprobt werden. Die Studierenden arbeiteten weitestgehend selbstgesteuert, während die Lehrperson die Zeit für individuelle Portfoliogespräche oder andere fachliche Abklärungen nutzen konnte.

Eine kurze Hinführung zum Thema, als Kernidee der Lehrenden zu verstehen, ist wie folgt formuliert:

In meiner Umwelt finde ich analoge und digitale Daten, die ich ineinander umwandeln kann. Der Computer hilft mir dabei. Als Mensch kann ich Lichtwellen und Schallwellen, also zeitlich kontinuierliche Signale, wahrnehmen.

Digitale Daten bestehen aus einer Abfolge von 0en und 1sen, sind also nicht kontinuierlich, sondern zeitlich diskret. Sie können durch einen Computer kopiert, verarbeitet und gespeichert werden. Dies kann Folgen für mich und meine Mitwelt haben.

⁹ <https://tinyurl.com/MI-KoA-informatiksysteme>

¹⁰ <https://www.calligraphr.com/>

¹¹ <https://tinyurl.com/MI-KoA-logo>

Die Studierenden zeichnen und beschreiben „den Weg“ des Tones in den Computer hinein und wieder heraus. Diese Zeichnung wird mit einem Partner diskutiert und mit Hilfe eines Lernfilms über Schallwellen, Abtastrate, Binärumwandlung, binäres Rechnen und Prozessorverarbeitung ergänzt (Abb. 3). Diese Übung, hier als didaktischer Selbstversuch durchgeführt, kann auch in der Schule so umgesetzt werden.

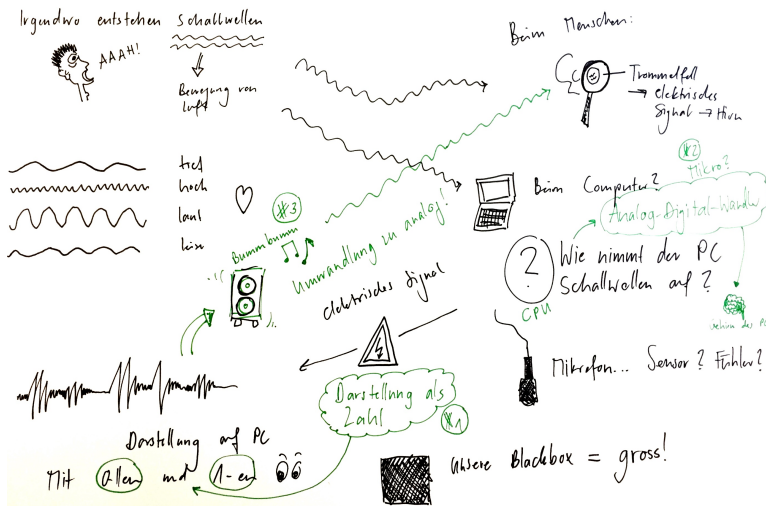


Abb. 3: Screenshot der Ansicht eines Auftrages (zur besseren Lesbarkeit leicht manipuliert)

Das Beispiel und die dazugehörige, hier nicht dargestellte Beschreibung der Zeichnung zeigt, dass Wellen und die Umwandlung in 0 und 1 durchaus bekannte Konzepte sind. Die Aufgabe der Dozentin bzw. des Materials ist es nun im Plenum und in Form einer Rückmeldung zu klären, was genau bei der Umwandlung von der Welle in ein elektrisches Signal passiert und Hinweise zu geben, wie sich Kinder der 5. oder 6. Klasse solch komplexe Vorgänge vorstellen können.

Es folgen Übungen zum ASCII-Code und zur Digitalisierung von Bildern, welche aber in den Reflexionen kaum erwähnt werden. Es ist daher davon auszugehen, dass die Digitalisierung des Tones ein anschauliches Phänomen darstellt.

Abschließend formulieren die Studierenden Kernideen zum Thema. Eine Studentin umschreibt sowohl ihre Entwicklung als auch ihre Vorhaben für den eigenen MI-Unterricht wie folgt: „Digitalisierung ist keine Zauberei, sondern wenn man weiß, wie es funktioniert, ist es erklärbar.“

Im Präsenzseminar wurde anschließend die Audiobearbeitung mit Audacity geübt und die Studierenden konnten so die erarbeiteten informatischen Konzepte sofort aktiv anwenden.

3.4 Fazit und Ausblick

Es konnte gezeigt werden, dass Dialogisches Lernen an der Hochschule dann besonders zielgerichtet umgesetzt werden kann, wenn offene, individuell bearbeitbare Aufträge Raum und Zeit schaffen, in einen Dialog mit den Studierenden zu treten. In Gesprächen und mit Hilfe von Kommentaren konnten die Facetten der Informatik, Verbindungen zur Medienbildung und fächerverbindende Potenziale diskutiert werden und die Arbeiten der Studierenden erhielten so einen Platz in der Lehrveranstaltung.

Die Aufgaben werden zunehmend in der Ausbildung von Primarlehrpersonen eingesetzt und werden zukünftig eine wachsende Bedeutung im Sinne einer Individualisierung und Flexibilisierung des Studiums erlangen.

Eine Entwicklungsaufgabe ist derzeit, die Aufgaben offen zur Verfügung zu stellen, sodass sie Lehrpersonen und Studierenden im gesamten deutschsprachigen (insb. Deutschschweizer) Raum zur Bearbeitung bereitstehen. Hier sind vorgängig Lizenzabklärungen notwendig, welche in Kürze abschlossen sein werden. Ebenso sind Kooperationen mit anderen, hochschulübergreifenden Projekten geplant und befinden sich derzeit in Abklärung.

Wünschenswert wäre eine Weiterentwicklung und Ergänzung der bestehenden Aufgabensets durch Studierende im Rahmen von Bachelorarbeiten. Entwicklungsarbeiten erfreuen sich immer größerer Beliebtheit und mit Hilfe der Studierenden lassen sich weitere spannende Themengebiete erschließen. Perspektivisch sind auch in der Schullandschaft solche offenen Aufgaben von Bedeutung, natürlich in Ergänzung bzw. unter Einbezug bestehender Lehrmittel.

Literatur

- [Je08] Jenert, T.: Ganzheitliche Reflexion auf dem Weg zu Selbstorganisiertem Lernen. In (Häcker, T.; Hilzensauer, W.; Reinmann, G., Hrsg.): Schwerpunkt "Reflexives Lernen". Bd. 2, 5, Bildungsforschung, S. 1–18, 2008.
- [KW17] Kalcsics, K.; Wilhelm, M.: Lernaufgaben gestalten. In (Kalcsics, K.; Wilhelm, M., Hrsg.): LERNWELTEN Natur - Mensch - Gesellschaft AUSBILDUNG. Schulverlag plus, Bern, Kap. 8, S. 81–89, 2017.
- [Le11] Leisen, J.: Kompetenzorientiert unterrichten. Unterricht Physik 123/124/100, S. 100–110, 2011.
- [Of04] Offermans, M.: Braucht Coaching einen Coach. Eine evaluative Pilotstudie. ibidem-Verlag, Stuttgart, 2004.
- [RG14a] Ruf, U.; Gallin, P.: Dialogisches Lernen in Sprache und Mathematik. Bd. 1: Austausch unter Ungleichen. Grundzüge einer interaktiven und fächerübergreifenden Didaktik. Klett, Seelze, 2014.

- [RG14b] Ruf, U.; Gallin, P.: Dialogisches Lernen in Sprache und Mathematik. Bd. 2: Spuren legen - Spuren lesen. Unterricht mit Kernideen und Reisetagebüchern. Klett, Seelze, 2014.
- [RG95] Ruf, U.; Gallin, P.: Ich mache das so! Wie machst du es? Das machen wir ab. Sprache und Mathematik, 1.-3. Schuljahr. Interkantonale Lehrmittelzentrale, Zürich, 1995.
- [RKW08] Ruf, U.; Keller, S.; Winter, F.: Besser lernen im Dialog. Dialogisches Lernen in der Unterrichtspraxis. Klett, Seelze-Velber, 2008.
- [We01] Weinert, F. E.: Vergleichende Leistungsmessung in Schulen- Eine umstrittene Selbstverständlichkeit. Leistungsmessungen in Schulen/, S. 17–31, 2001.
- [Wi16] Winter, F.: Leistungsbewertung. Schneider Verlag Hohengehren GmbH, Baltmannsweiler, 2016.
- [Wi18] Winter, F.: Lerndialog statt Noten. Neue Formen der Leistungsbeurteilung. Beltz, Weinheim und Basel, 2018.
- [WLS16] Wilhelm, M.; Luthiger, H.; Schweizer, G.: Prozessmodell kompetenzfördernder Aufgabensets. Ein Beispiel für den NMG-Unterricht aus biologischer Perspektive. In (Keller, S.; Reintjes, C., Hrsg.): Aufgaben als Schlüssel zur Kompetenz. Waxmann, Münster, S. 345–356, 2016.

FLACI – Eine Lernumgebung für theoretische Informatik

Michael Hielscher¹, Christian Wagenknecht²

Abstract: FLACI (www.flaci.com) steht für Formal Languages and Compilers and Interpreters und ist eine modular aufgebaute, webbasierte Lernumgebung für grundlegende Inhalte der theoretischen Informatik. Die Theorie formaler Sprachen und Automaten befasst sich mit potentiell unendlichen Mengen. Die damit verbundene Begrifflichkeit in FLACI adäquat abzubilden, ist die besondere Herausforderung. Die Lernumgebung bringt deklaratives, konzeptionelles und prozedurales Wissen mit, um Definitionen, Simulationen, Transformationen und Konvertierungen für formale Grammatiken, reguläre Ausdrücke und abstrakte Automaten vornehmen zu können. Die Webapplikation FLACI stellt eine konsequente Weiterentwicklung der Lernumgebung AtoCC aus dem Jahr 2006 dar. Eine lokale Installation entfällt, wodurch diverse softwaretechnische und betriebssystemspezifische Probleme vermieden werden können. Dadurch ist der Einsatz von FLACI, neben dem akademischen Bereich, auch für den Informatikunterricht in der Schule (vorwiegend Sekundarstufe II) attraktiv. Die Applikation ist sowohl im Unterricht als auch außerhalb zeitlich und örtlich unabhängig verfügbar, speichert alle erarbeiteten Inhalte auf Wunsch automatisch und regelmäßig in der FLACI-Cloud und bietet damit Potential zur kollaborativen Arbeit. Als offene Lernumgebung bietet FLACI einen Konstruktionsraum für einfache Compiler und Interpreter als praxisrelevante Motivation für die Anwendung von Theoriekenntnissen. Hierfür bietet die Umgebung spezielle kontextfreie Sprachklassen und Transformationen an, die die Herstellung von Compilern konzeptionell unterstützen. Die Verzahnung von Theorie und praktischer Anwendung wird seit vielen Jahren im Informatikstudium umgesetzt. Im Beitrag wird FLACI im Überblick vorgestellt und zu den GI-Bildungsstandards "Informatik für die Sekundarstufe II" in Beziehung gebracht.

Keywords: Theoretische Informatik, Formale Sprachen, Abstrakte Automaten, Lernumgebung

1 Einleitung

Die theoretische Informatik (TI) ist an Universitäten und Hochschulen ein fester Bestandteil des Informatikstudiums. Zunehmend werden ausgewählte Konzepte dieses Gebietes in den Schulunterricht integriert. Die GI-Bildungsstandards für Informatik in der Sekundarstufe II [RBD16] weisen Sprachen und Automaten als einen zentralen Inhaltsbereich aus und definieren konkrete Themen, an denen sich die Lehrpläne orientieren können. Im gymnasialen Informatikunterricht ist die TI je nach Bundesland und Lehrplan bereits ein obligatorischer oder wenigstens ein möglicher Wahlbereich, s. z. B. [FPA07]. Auf die besonderen Anforderungen einer stufengerechten fach- und mediendidaktischen Konzeption für diesen Inhaltsbereich wurde hingewiesen, s. z. B. [MHL16].

¹ Medien und Schule, Pädagogische Hochschule Schwyz, Forschung, Zaystrasse 42, Goldau, CH-6410, michael.hielscher@phsz.ch

² Fakultät Elektrotechnik und Informatik, Hochschule Zittau/Görlitz, FB Informatik, Brückenstr. 1, Görlitz, 02826, c.wagenknecht@hszg.de

In den vergangenen Jahrzehnten entstanden zahlreiche Lernumgebungen für TI: JFLAP [RF06], Kara [BHN03], Exorciser [Ts04], AtoCC [HW06], [HW07] und [WH14]. Im Folgenden beschränken wir unseren Betrachtungsrahmen auf die TI-Teilgebiete “Theorie der formalen Sprachen” und die “Automatentheorie”. Diese Lernumgebungen decken jeweils einen Teil der geforderten Kompetenzen (in unterschiedlicher Ausprägung) ab und sind mehrheitlich für das universitäre Studium konzipiert. Mit AtoCC wurde ein Weg gefunden, die begriffliche Arbeit durch Definition von Instanzen und Anwendung algorithmischer Zusammenhänge zu betonen [WH14]. Dabei wird die Aktivität des Lernenden nicht nur auf das seit über 50 Jahren ausgearbeitete Gebiet der Automaten Simulationen, s. [CSK11], reduziert.

Die mit AtoCC gewonnenen fachdidaktischen Erfahrungen (Studium und Sekundarstufe II) sind in die Entwicklung von FLACI eingeflossen, s. Kap. 2. Darüber hinaus wurden moderne mediendidaktische Aspekte beachtet: Der überwiegende Teil der im Einsatz befindlichen Lernumgebungen, einschl. AtoCC, basiert auf Technologien, die von mobilen Geräten (z. B. Tablets) nicht mehr unterstützt werden. Als Reaktion auf die zunehmende Verbreitung solcher Geräte in der Hand von Lernenden und Lehrpersonen wurde AtoCC grundlegend überarbeitet und FLACI als moderne Webapplikation entwickelt. Ein ähnliches Ziel wird auch in [PT18] verfolgt. Unter Berücksichtigung der Bildungsstandards ist ein didaktisches Werkzeug entstanden, das sowohl im Informatikstudium als auch im Informatikunterricht (vorwiegend Sekundarstufe II) eingesetzt werden kann. Darüber hinaus eignet sich FLACI als praxistaugliche Entwicklungsumgebung für Compiler und Interpreter, wie sie in modernen Webapplikationen Verwendung finden.

2 Struktur der Lernumgebung FLACI

Der Einsatz von FLACI ist darauf gerichtet, den Informatikunterricht durch Experimentiergelegenheiten zu ergänzen. Automatisiertes Feedback unterstützt die Lernenden etwa bei der Konstruktion von Automatenmodellen und formalen Grammatiken und weist auf allfällige Fehler oder Unvollständigheiten hin. Die Simulation von Ableitungen, von Arbeitsschritten abstrakter Automaten und von Compilationsprozessen ist dynamischer Natur und wird in FLACI mediendidaktisch passender repräsentiert als durch die Darstellung einzelner Visualisierungsschritte in gedruckten Unterrichtsmaterialien. FLACI stellt eine umfangreiche Palette an Transformations- und Konvertierungsverfahren bereit, die im Unterricht erarbeitet und von den Lernenden aufgabengeleitet und ergebnisorientiert angewandt werden. Außerdem bietet die Lernumgebung unmittelbare Hilfe zur Selbsteinschätzung vorgelegter Lösungen.

FLACI besteht aus fünf eigenständigen, aufeinander aufbauenden Modulen, die in der vorgegebenen Folge aber auch einzeln, verflochten oder unabhängig voneinander verwendet werden können. Im Folgenden werden die einzelnen Module kurz beschrieben.

2.1 Formale Sprachen

Das Theoriemodul bietet eine einfache, interaktive Umgebung, um zentrale Begriffe formaler Sprachen, wie Alphabet, Zeichen, Wort, Wortmenge und Sprache, zu explorieren. Es stehen vordefinierte Alphabete zur Auswahl. Zeichen, die aus mehreren Tastaturzeichen bestehen, bergen die große Gefahr einer typischen Fehlvorstellung vom Zeichenbegriff. Sie werden als atomare Blättchen dargestellt und verwendet. Die Lernenden können zudem eigene Alphabete definieren, s. Abbildung 1.

The image shows two side-by-side screenshots of the FLACI interface. Both have a blue header with 'Formale Sprachen' and a user profile 'MICHAEL DE'. The left screenshot has tabs for 'Alphabet und Zeichen', 'Wort', 'Wortmenge', and 'Sprache', with 'Wort' selected. It contains an information box defining a word w as a string of characters from an alphabet A . Below, it shows an alphabet A with icons (book, cloud, eye, person, heart, star) and a word w with the first three icons. A second information box defines the length of a word. At the bottom, it shows 'Wortlänge | w | = 3' and a 'WEITER ZUR WORTMENGE' button. The right screenshot has the same tabs but 'Alphabet und Zeichen' is selected. It defines an alphabet A as a finite, non-empty set of characters. Below, it asks to choose an example alphabet from six options: $A_1 = \{0, 1\}$, $A_2 = \{a, b, c, \dots, z\}$, $A_3 = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, $A_4 = \{\text{book}, \text{cloud}, \text{eye}, \text{person}, \text{heart}, \text{star}\}$ (selected), $A_5 = \{\text{begin}, \text{end}, \text{for}, \text{while}, \text{do}, \text{repeat}, \text{until}\}$, and A_6 is self-definable. A 'WEITER ZUM WORT' button is at the bottom.

Abbildung 1: Alphabet- und Zeichenbegriff (links) und Wortbegriff (rechts)

Die mögliche Eigenschaft von Wortmengen und Sprachen abzählbar unendliche Mengen zu sein, wird durch eine „aufklappbare“ Punktnotation, für beliebig viele weitere Elemente der betrachteten Mengen, erfahrbar. Die in FLACI eingebauten Definitionen der o. g. Begriffe dienen als verbindliche Implementierung der durch die Lehrperson eingeführten Begrifflichkeit, s. Abbildung 1.

2.2 Reguläre Ausdrücke

Sprachen durch manuelle Auswahl einzelner Wörter aus der jeweils zugrunde liegenden Wortmenge zu definieren erschöpft sich schnell und führt zur Notwendigkeit geeigneter Beschreibungsmittel für potenziell unendliche Mengen. Reguläre Ausdrücke sind eine kompakte und mächtige Definitionsform regulärer Sprachen. Die Syntax regulärer Ausdrücke wird in FLACI an konkreten Beispielen im Dialog illustriert. Zu jedem angegebenen regulären Ausdruck wird ein entsprechendes Syntaxdiagramm generiert. Die Betrachtung

tung von Kfz-Kennzeichen und E-Mail-Adressen werden zur abschließenden selbstständigen Übung angeboten. In einem Experimentierbereich lassen sich beliebige reguläre Ausdrücke und Suchtexte eingeben, Zufallswörter aus der beschriebenen Sprache erzeugen, äquivalente Grammatiken und endliche Automaten generieren, s. Abbildung 2.



Abbildung 2: Reguläre Ausdrücke mit interaktiven Erklärungen (links) und Beispielaufgaben (rechts)

Ergänzend kann die praktische Bedeutung regulärer Ausdrücke etwa bei Such- und Ersetzungsaufgaben in gängigen Texteditoren oder Werkzeuge wie `grep` auf der Kommandozeile thematisiert werden. Deshalb wurde in FLACI eine praxisrelevante Notation regulärer Ausdrücke gewählt und auf die formale Betrachtung regulärer Mengen und deren Überführung in zugehörige Sprachen verzichtet, was im Unterricht jedoch thematisiert werden muss, s. [WH14].

2.3 Kontextfreie Grammatiken

Schnell stößt man an die Grenzen regulärer Sprachen. Praktische Einsatzzwecke motivieren die Betrachtung formaler Sprachen, die nicht mit regulären Ausdrücken definiert werden können. Folglich stellt sich die Frage nach der Definition und passenden Beschreibungsmitteln für Sprachklassen, die über reguläre Sprachen hinausgehen.

Abstrakte Automaten und formale Grammatiken leisten das Gewünschte. Das Grammatik-Modul in FLACI erlaubt sowohl eine textuelle (BNF: Backus-Naur-Form) als auch visuelle Eingabe (Syntax-Diagramm) von Grammatiken, s. Abbildungen 3.

Für beliebige Eingabewörter, die zur Sprache gehören, können Ableitungen als Ableitungsbäume und tabellarische Satzformlisten dargestellt werden. Der Ableitungsprozess lässt sich schrittweise animieren und zwischen Links- und Rechtsableitung umschalten. Bei Mehrdeutigkeiten werden für ein betrachtetes Wort mehrere Ableitungsbäume zur

Auswahl angeboten. Das Modul stellt Algorithmen bereit, die Grammatiken z. B. zum Zwecke der Vereinfachung umformen oder beispielsweise überprüfen, ob die grammatischen Regeln syntaktisch denen einer regulären Grammatik entsprechen oder ob es sich um eine LL(1)-Sprache handelt. Außerdem lassen sich reguläre Grammatiken in reguläre Ausdrücke oder endliche Automaten konvertieren. Kontextfreie Grammatiken (kfG) können in Kellerautomaten (s. Abschnitt 2.4) konvertiert werden.

The screenshot shows the FLACI software interface. On the left, a list of grammar rules is displayed in a table format:

1	Stylesheet	→ Ruleset Ruleset Stylesheet
2	Ruleset	→ R1 R3
3	R1	→ Selector Selector , R1
4	R2	→ Declaration Declaration ; R2
5	R3	→ { R2 }
6	Selector	→ Identifier
7	Declaration	→ Property : Expression
8	Property	→ Identifier
9	Expression	→ Number Identifier Number %
10	Identifier	→ Buchstabe Buchstabe Identifier2
11	Identifier2	→ Buchstabe Ziffer - Buchstabe Identifier2
12	Number	→ Zahl Zahl . Zahl
13	Ziffer	→ 0 1 2 3 4 5 6 7 8 9

Below the rules, a section titled "So definiert man eine Grammatik in FLACI:" provides instructions:

- Man notiert nur die Produktionsregeln.
- Nach Vereinbarung ist das Nichtterminal auf der linken Seite der zuerst angegebenen Regel das Spitzensymbol der Grammatik
- Für ε in ε-Regeln ist EPSILON zu schreiben: ε → a | EPSILON
- Die Pünktchennotation hilft den Schreibaufwand zu reduzieren:

On the right, the "Ableitungsbaum von 1:" (Derivation tree) is shown. The root node is "Stylesheet", which derives "Ruleset". "Ruleset" derives "R1" and "R3". "R1" derives "Selector", which further derives "Identifier" (Buchstabe 'h' and Identifier2 't'). "R3" derives "Declaration", which derives "Property" (Identifier) and "Expression". "Property" derives "Identifier" (Buchstabe 'c' and Identifier2 'o'). "Expression" derives "Identifier" (Buchstabe 'b' and Identifier2 'i'). The final tree structure represents the derivation of the string "h t c o b i".

Abbildung 3: Formale Grammatiken in BNF (links) und Ableitungsbaum (rechts)

2.4 Abstrakte Automaten

Als äquivalente Beschreibungsmittel formaler Grammatiken können abstrakte Automaten mit Hilfe des entsprechenden Moduls in FLACI eingeführt werden. Das Modul umfasst deterministische und nichtdeterministische endliche Automaten (DEA, NEA), Kellerautomaten (DKA, NKA) und deterministische Turingmaschinen (DTM), s. Abbildung 4. Die Konstruktion erfolgt primär grafisch mittels Übergangsgraph des Automaten und einer vorherigen Definition der verwendeten Alphabete. Endliche Automaten lassen sich zusätzlich über eine editierbare Übergangstabelle erstellen und bearbeiten. Die Arbeitsweise der Automaten lässt sich für beliebige Eingabewörter simulieren und bietet die Möglichkeit einzelne Schritte animiert nachvollziehen. Nichtdeterminismus wird in Form geklonter Maschinen, deren Arbeitsweise einzeln beobachtet werden kann, erkennbar. Es stehen Verfahren bereit, mit denen endliche Automaten in äquivalente reguläre Ausdrücke oder reguläre Grammatiken, Kellerautomaten in kontextfreie Grammatiken konvertiert werden können. Zu jedem endlichen Automaten kann der äquivalente Minimal-Automat erzeugt werden. Auch dieses Modul bietet einen fließenden Übergang zu den Themen eines Informatikstudiums. Vielfältige Assistenzsysteme unterstützen bei der Konstruktion von Automaten (z. B. automatische Vervollständigung bei endlichen Automaten und farbliche Hervorhebungen bei fehlenden Übergängen).

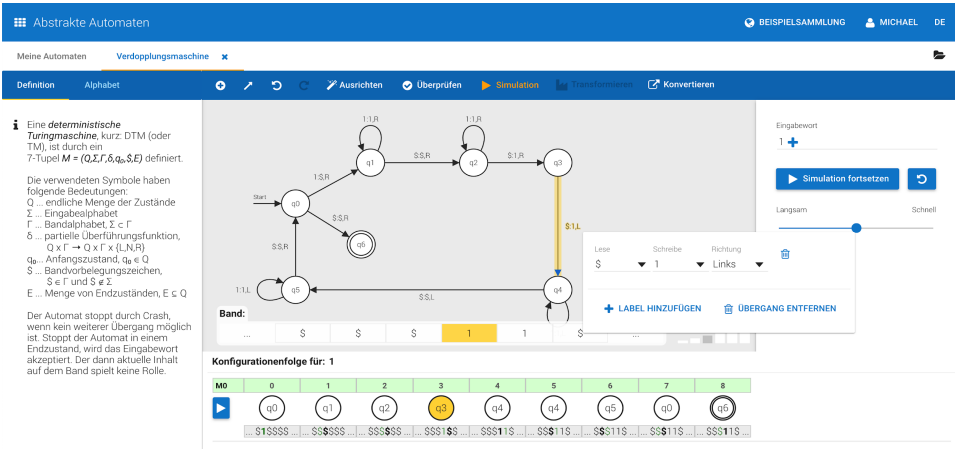


Abbildung 4: Simulation einer deterministischen Turingmaschine in FLACI

2.5 Compiler und Interpreter

Als Motivation für die Behandlung formaler Sprachen, abstrakter Automaten und zugehöriger Konzepte haben sich praktische Anwendungen im Compilerbau als tragfähig erwiesen. FLACI bietet in diesem Modul einen Editor zur Modellierung von Übersetzungsprozessen mit Hilfe von T-Diagrammen, s. Abbildung 5, sowie einen Editor zur Entwicklung von Compilern und Interpretern mit Hilfe einer formalen Beschreibungssprache (TDL: Translator Description Language). Compiler werden dabei nicht per Hand codiert, sondern mit Compilergeneratoren (Compiler Compiler) aus der formalen Definition automatisiert generiert. Als Ausgangspunkt kann modulübergreifend eine bereits erstellte formale Grammatik direkt verwendet werden, wobei automatisch passende Tokenklassen für den Scanner und Regeln für den Parser generiert werden. Für die Lernenden bleibt die Aufgabe, synthetische Attribute (S-Attribute) für jede Regel des Parsers als JavaScript-Code zu definieren, s. Abbildung 6. Der Scanner arbeitet mit regulären Ausdrücken, die zuvor erarbeitet werden müssen. Der Parser verwendet die gleiche Darstellung editierbarer Syntaxdiagramme wie das Grammatik-Modul von FLACI. Sowohl Scanner als auch Parser können simuliert und Zustandstabellen des generierten Compilers angezeigt werden. Enthält die Grammatik Mehrdeutigkeiten wird durch entsprechende Meldungen darauf hingewiesen. Alle erstellten Compiler und Interpreter lassen sich über eine URL als eigenständiges JavaScript-Programm in beliebige Web-Applikationen einbinden.

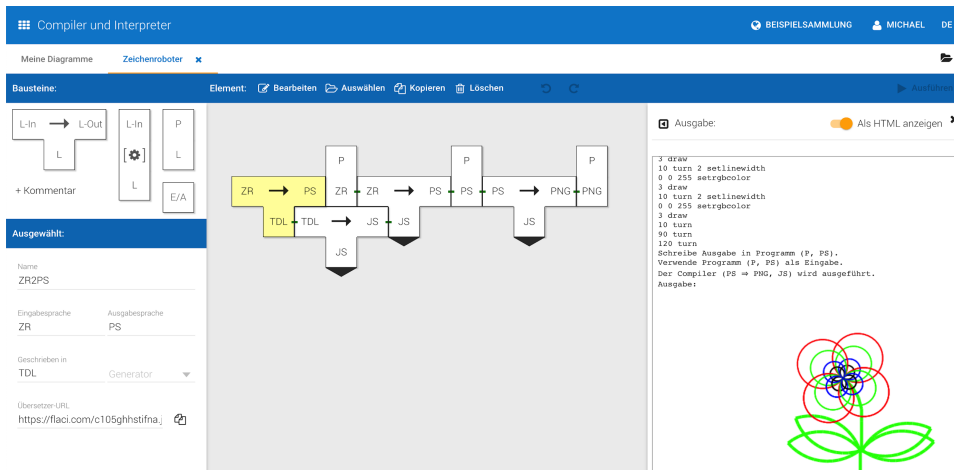


Abbildung 5: T-Diagramm für einen Übersetzungsprozess über mehrere Teilschritte und Ausführung (rechts)

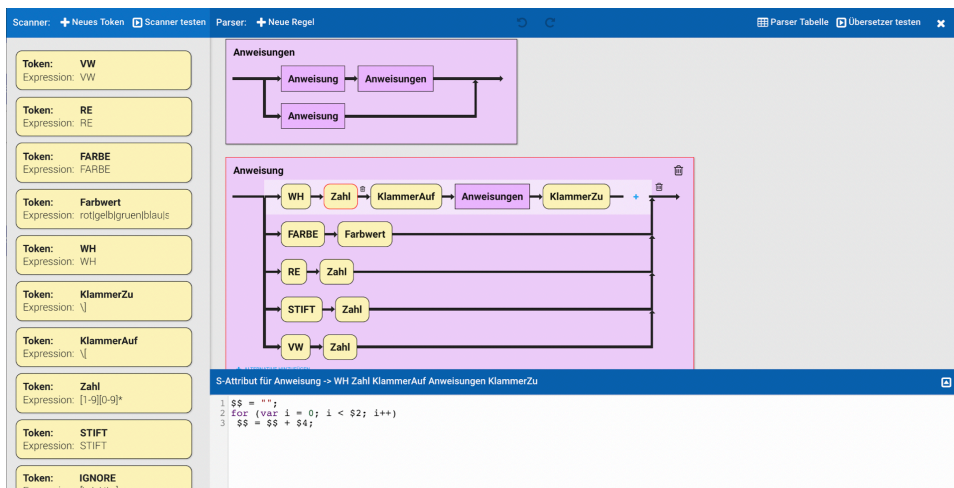


Abbildung 6: Compilerdefinition mit Scanner und Parser inklusive S-Attribute (unten)

2.6 Zusammenwirken der Module

Die erarbeiteten Grammatiken, Automaten und Compiler werden wahlweise mit Hilfe des lokalen Webbrowsers oder über den FLACI-Server (bei vorheriger Anmeldung) gespeichert. Die erstellten Produkte können auch als JSON-Dateien heruntergeladen, über Web-links auf dem FLACI-Server mit anderen Lernenden ausgetauscht oder von Lehrpersonen

für Lernende bereitgestellt werden. Auf diese Weise können auch Übungsmaterialien vorgegeben werden. Für viele Darstellungen innerhalb der Module (Automatengraphen, Ableitungsbäume usw.) lassen sich Grafiken für gedruckte Materialien in verschiedenen Formaten exportieren. Zwischen den einzelnen Modulen bestehen vielfältige Wechselwirkungen, die es ermöglichen, das Produkt des einen Moduls in einem anderen weiterzuverarbeiten. Abbildung 8 illustriert Konvertierungsmöglichkeiten zwischen den Modulen im Überblick.

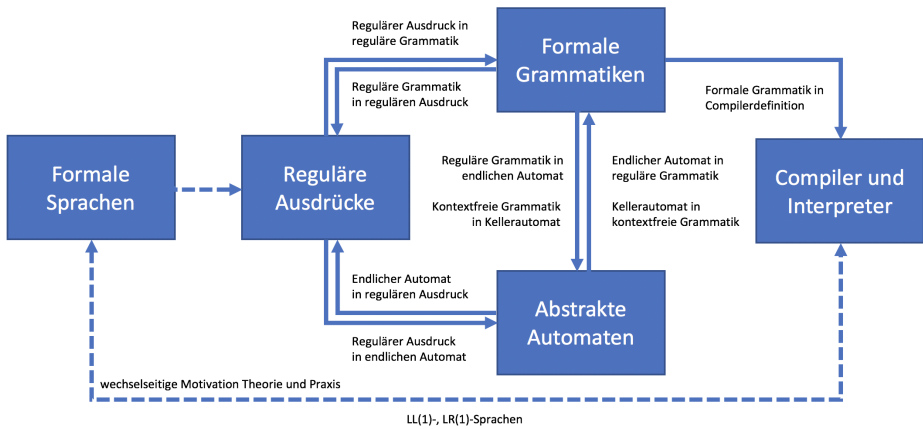


Abbildung 8: Die fünf Module von FLACI und Konvertierungsmöglichkeiten

3 Berücksichtigung der GI-Bildungsstandards

Die Lernumgebung FLACI deckt den Kompetenzbereich „Formale Sprachen und Automaten“ aus den GI-Bildungsstandards für die Sekundarstufe II, s. Tabelle 1, weitgehend ab und bietet darüber hinaus praktische Anwendungsgebiete im Compilerbau. Die in den Standards vorgeschlagene Niveaudifferenzierung lässt sich fließend mit FLACI abbilden: Anfangs werden nur reguläre Sprachen und endliche Automaten thematisiert und erst in einem zweiten Schritt kommen weitere Sprachklassen bedarfsgemäß hinzu. Bereits die FLACI-Module zu regulären Sprachen und Ausdrücken können die wichtigsten Teile der Grundanforderungen zur Unterscheidung formaler und natürlicher Sprachen sowie die Verwendung von Sprachdefinitionen zur Analyse, Beschreibung und Entwicklung formaler Sprachen abdecken. Die Ergebnisse der FLACI-Konvertierungen in Grammatiken bzw. Automaten in Verbindung mit Äquivalenzbetrachtungen werden auch für Konstruktions-, Kontroll- und Vergleichszwecke herangezogen.

Der Übergang von regulären zu kontextfreien Sprachen und Themen des Informatikstudiums ist ohne deren zwingende Thematisierung fließend möglich. Der modulare Aufbau der Lernumgebung ermöglicht einen selektiven Einsatz für einzelne Themenbereiche und unterstützt deren Struktur und Abfolge, wie sie in den Bildungsstandards angegeben sind.

Tabelle 1: Auszug aus Bildungsstandards für die Sek. II und passende Module aus FLACI

Grundlegendes und erhöhtes Anforderungsniveau	Modul in FLACI
<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> ● vergleichen formale mit natürlichen Sprachen, ● untersuchen den Zusammenhang zwischen einer Grammatik und ihrer Sprache, leiten Wörter einer Sprache ab und stellen Ableitungsbäume dar, ● verwenden Sprachdefinitionen (z. B. Grammatiken, Syntaxdiagramme) zur Analyse, Beschreibung und Entwicklung formaler Sprachen, ● überführen Grammatiken in endliche Automaten und umgekehrt. 	<p>“Formale Sprachen”</p> <p>“Formale Grammatiken”</p> <p>“Reguläre Ausdrücke” “Formale Grammatiken”</p> <p>“Formale Grammatiken” “Abstrakte Automaten”</p>
<p>Erhöhtes Anforderungsniveau</p> <p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> ● erläutern den Zusammenhang zwischen Grammatiken, Sprachen und Automaten, ● analysieren und implementieren Programme zu Problemstellungen auf Kellerautomaten, Turingmaschinen oder Registermaschinen, ● erläutern prinzipielle und praktische Grenzen der Berechenbarkeit. 	<p>“Formale Grammatiken” “Abstrakte Automaten” “Abstrakte Automaten” “Compiler und Interpreter”</p>

4 Fazit

Beim Einsatz der Lernumgebung AtoCC wurden in mehr als 10 Jahren im Informatikstudium und -unterricht (Sekundarstufe II) positive didaktische Erfahrungen gemacht. Mehr als 25'000 Downloads der AtoCC-Software von Lehrenden und Lernenden an Schulen und Hochschulen lassen ein beachtliches Interesse erkennen. Die Weiterentwicklung zu FLACI als Webapplikation stellt einen wichtigen Schritt zur Fortführung des Ansatzes dar: FLACI berücksichtigt zukunftsfähige mobile Geräte und reduziert den Installationsaufwand in Schulen sowie auf privaten Geräten der Lernenden und der Lehrpersonen. Nach den überaus positiven Erfahrungen mit FLACI im Informatikstudium ermöglicht die Passfähigkeit mit den Rahmenvorgaben für den Informatikunterricht den Einsatz dieser

Lernumgebung auch im Schulkontext. Zu einer wissenschaftlichen Evaluation wird aufgerufen. FLACI steht unter www.flaci.com zur kostenlosen Nutzung zur Verfügung. Der gesamte Quellcode des Projekts wurde unter einer MIT-Lizenz auf [github](https://github.com) veröffentlicht.

5 Literatur

- [BHN03] Brändle, M.; Hartmann, W.; Nievergelt, J.; Reichert, R.; Schlatter, T.: Kara: Ein theoriebasierter Ansatz für Lernumgebungen zu fundamentalen Konzepten der Informatik. In INFOS 2003 (S. 201-210).
- [CSK11] Chakraborty, P.; Saxena, P. C.; Katti, C. P.: Fifty Years of Automata Simulation: A Review. In: *acm inroads*, 2011, Vol. 2, No. 4.
- [FPA07] Friedrich, S., Puhlmann, H., Altdorf, L. G.: Bildungsstandards Informatik - von Wünschen zu Maßstäben für eine informatische Bildung. In: INFOS 2007, S. 21-32.
- [HW06] Hielscher, M., Wagenknecht, C.: AtoCC: learning environment for teaching theory of automata and formal languages. In *ITiCSE 2006*, Vol. 6, pp. 26-28.
- [HW07] Hielscher, M., Wagenknecht, C.: AtoCC-didaktischer Ort und erste Erfahrungen. In INFOS 2007, S. 159-170.
- [MHL16] Mane, D. T.; Howal, S. S.; Lokare, V. T.: Problem-based Learning using Simulation Tools for Automata Theory. In: *Journal of Engineering Education Transformations*, Special Issue, 2016.
- [PT18] Pereira, C. H., Terra, R.: A Mobile App For Teaching Formal Languages and Automata. In *Wiley Online Library*, Special Issue Article, 2018.
- [RBD16] Röhner, G.; Brinda, T.; Denke, V. u.a.: Bildungsstandards Informatik für die Sekundarstufe II. Beilage zu LOG IN, 36. Jg. 2016, Heft Nr. 183/184.
- [RF06] Rodger, S. H., Finley, T. W.: *JFLAP: An Interactive Formal Languages and Automata Package*. Jones & Bartlett Learning, 2006.
- [Ts04] Exorciser: Automatic Generation and Interactive Grading of Structured Exercises in the Theory of Computation. Doctoral dissertation, ETH Zurich, 2004.
- [WH14] Wagenknecht, C., Hielscher, M.: *Formale Sprachen, abstrakte Automaten und Compiler*. 2. Aufl., 2014, Springer Fachmedien Wiesbaden.

Innovative Gestaltung und Einsatz von Erklärvideos im Informatikunterricht

Tamara Hochthurn, Jens Gallenbacher

Abstract: Erklärvideos bieten eine Möglichkeit, zeit- und ortsungebunden individuell im eigenen Tempo Inhalte zu erlernen. Ein Nachteil ist die oft passive Konsumentenhaltung von Schülerinnen und Schülern, die das Erklärvideo nur oberflächlich betrachten. Dieser Beitrag stellt einen Unterrichtsverlauf mit selbstentwickelten Erklärvideos zum Thema Binärsystem sowie Nutzung verschiedener Bastelkärtchen als Artefakte vor. Die Idee ist, dass die Lernenden durch den haptischen Einsatz der Bastelkärtchen während des Erklärvideos in eine aktive Rolle schlüpfen. In einer Vorstudie mit wenigen Schulklassen wurde der Unterschied der Kompetenzentwicklung mit und ohne Artefakte untersucht. Dazu wurde die Methode „Flipped Classroom“ in zweierlei Form angewandt. Zum einen als klassische Inverted Classroom Variante, bei der die Schülerinnen und Schüler als vorbereitende Hausaufgabe das Erklärvideo bearbeiteten. Zum anderen als In-Class-Flip, bei dem das Erklärvideo im Unterricht betrachtet wird. Die Artefakte hatten in den vorliegenden Ergebnissen einen kurzfristigen leicht positiven Effekt. Die Methode des klassischen Flipped Classrooms hatte einen leicht positiven Einfluss auf Nachhaltigkeit.

Keywords: Erklärvideo, Artefakte, Flipped Classroom, Inverted Classroom, In-Class-Flip

1 Einleitung

„Informatik für alle“ ist das Motto der diesjährigen INFOS. In der Schulpraxis ist dieses „für alle“ jedoch gar nicht so leicht umzusetzen. Ständig finden Sport-, Musikevents oder Exkursionen statt, weshalb einige Schülerinnen und Schüler im Informatikunterricht nicht anwesend sind. Einige sind zudem schlichtweg krank, sodass sie den Unterricht versäumen. Jene Lernende verpassen somit die neuen Inhalte und Informatikunterricht ist an diesem Punkt nicht „für alle“. Um zu gewährleisten, dass abwesende Schülerinnen und Schüler neuen Unterrichtsstoff begreifen und zu einer ihnen beliebigen Zeit selbstständig nacharbeiten können, bieten sich Erklärvideos an.

Neben diesem mehr strukturellen Problem haben wir es im Informatikunterricht mit meistens sehr heterogenen Lerngruppen zu tun. Auch hier bietet sich an, einerseits durch selbstgesteuertes Lernen „im eigenen Tempo“, andererseits durch individuelle Förderung mit zusätzlichen Einheiten per Video eine zusätzliche Form der Differenzierung durchführen zu können.

In einer empirischen Untersuchung an einem Gymnasium wurden Erklärvideos zum Thema Binärsystem für den Informatikunterricht einer zehnten Klasse entwickelt. Als Nachteil von Erklärvideos ist die passive Rolle der Schülerinnen und Schüler zu nennen, welche in einer Art Konsumentenhaltung nicht unbedingt aufmerksam das Erklärvideo schauen und mitdenken. Deshalb wurde die innovative Idee umgesetzt, mit Artefakten zu

arbeiten. Diese Artefakte waren die Punktekärtchen aus der Abenteuer Informatik Ausstellung sowie an die binäre Uhr der Ausstellung angelehnte Lämpchenkarten [Ga17]. Eine wichtige Frage ist, ob dieser aktive Einsatz von Artefakten während des Schauens des Erklärvideos einen Effekt auf die Kompetenzentwicklung der Schülerinnen und Schüler hat. Des Weiteren wurde untersucht, ob jene erfolgreicher sind, wenn sie die Erklärvideos als klassischen Flipped Classroom zu Hause anschauen, oder wenn sie an einem In-Class-Flip teilnehmen.

Aufgrund der geringen Teilnehmerzahl handelt es sich nur um eine Vorstudie und es lassen sich aus der Untersuchung noch keine allgemeingültigen Aussagen schließen.

2 Theoretischer Hintergrund

2.1 Erklärvideos

Erklärvideos sind im Sinne von [Wo15] „eigenproduzierte Filme, in denen erläutert wird, wie man etwas macht oder wie etwas funktioniert bzw. in denen abstrakte Konzepte erklärt werden.“ Diese Definition ist sehr offen, sodass es nicht verwunderlich ist, dass eine Fülle und Vielzahl von unterschiedlichen Erklärvideos beispielsweise auf YouTube zu finden sind. Diese lassen sich in unterschiedliche Typen einteilen. Es gibt zum Beispiel Screencasts, in denen ein Bildschirm aufgenommen wird, oder Erklärvideos, in denen eine Sprecherin oder ein Sprecher einen Vortrag hält [Br17; Sc16; Sc13b]. Für diesen Unterrichtsentwurf wurden Videos nach Art der Legetechnik [Fi18; Sc16; Sc13b; Sc13a] produziert. Die Charakteristik eines Legevideos ist, dass auf einem neutralen Hintergrund Bilder und Textbausteine bewegt und real gefilmt werden. Alternativ beziehungsweise ergänzend kann auch mit einem Stift etwas geschrieben werden. Ein Merkmal sind die sichtbaren Hände, die die Elemente bewegen, während eine Stimme aus dem Off passend dazu den Inhalt erklärt. Sogenanntes Storytelling bettet die Erklärungen in eine fiktive Rahmenhandlung ein [Sc16; Sc13b].

Ein entscheidender Vorteil von Erklärvideos ist die Selbststeuerung des Lernprozesses durch die Schülerinnen und Schüler. Diese haben die Möglichkeit das Erklärvideo anzuschauen wo und wann sie möchten. Außerdem kann dabei jeder in seinem eigenen Tempo vorgehen und gegebenenfalls vor- oder zurückspulen, wodurch eine gewisse Differenzierung unterstützt wird [Ab18; Br17; Ha17; St10].

Aus Sicht der Lehrkraft ist ein Nachteil hingegen der erhöhte Arbeitsaufwand für die Produktion von Erklärvideos [Fr16; St10]. Dieser relativiert sich allerdings dadurch, dass mit der Zeit auf einen Fundus auf öffentlichen Plattformen zugegriffen werden kann. Des Weiteren besteht die Gefahr, dass Schülerinnen und Schüler Erklärvideos als Freizeitfilme ansehen und in eine passive Haltung verfallen [Bü18; Fi18; We12].

2.2 Flipped Classroom

Um Erklärvideos im Unterricht einzusetzen, bietet sich die Methode des Flipped Classrooms, oder auch Inverted Classrooms, an. Dabei werden die gewöhnlichen Phasen der Stoffvermittlung durch die Lehrkraft im Unterricht und die anschließende Übungsphase, welche in der Regel als Hausaufgabe zu Hause von den Schülerinnen und Schülern bearbeitet wird, „vertauscht“ [Be12; Bü18; Fä16; Ha12b; Sc12]. Konkret eignen sich die Lernenden in der ersten Phase - der sogenannten Inputphase beziehungsweise Selbststudienphase - selbstständig neues Wissen an. Dies geschieht zum Beispiel mit Hilfe eines Erklärvideos. Anschließend findet in der Präsenzphase im Unterricht ein Üben und Vertiefen der neuen Inhalte statt [Be12; Bü18; Fä16; Fi17; Ha18; Ha12a; Ha12b; Sa12].

Ein zentraler Punkt im Flipped Classroom ist die Schülerzentrierung. In der Inputphase sind die Schülerinnen und Schüler selbst verantwortlich, sich den neuen Stoff anzueignen. Die Schülerzentrierung in der Präsenzphase variiert jedoch in ihrer Ausprägung in der Praxis stark. So sprechen einige Autorinnen und Autoren lediglich von einer eigenständigen Auswahl aus mehreren Aufgaben. Andere wiederum deuten die Schülerzentrierung im strengeren Sinne, sodass die Schülerinnen und Schüler den Unterricht in der Präsenzphase auch inhaltlich selbstständig gestalten, nachdem sie das Erklärvideo in der Inputphase betrachtet haben [Ab18; Be12; Fä16; Fi17; Go14; Sa12; Sc12].

Ein weiterer Vorteil der Flipped Classroom Methode ist ein effektiver Zeitgewinn im Unterricht, der sinnvolles Vertiefen, Üben, Anwenden oder individuelles Differenzieren ermöglicht [Bü18; Fä16; Fi17; Ha18; Ha12a; Sa12; Sc12; We12].

Problematisch ist, wenn Schülerinnen und Schüler in der Inputphase das Erklärvideo nicht angeschaut haben und somit in der Präsenzphase ohne nötiges Vorwissen auftauchen. [Fi17; Go14; Ha12b; Sa12].

2.3 In-Class-Flip

Um diesen entscheidenden Nachteil zu umgehen, gibt es die Variante des In-Class-Flips. Bei diesem findet die erste Phase des Inverted Classrooms nicht zu Hause bei den Lernenden statt, sondern im Unterricht. Die Erarbeitung bleibt dennoch individuell in eigenem Tempo und nicht als von der Lehrkraft vorgegebener Inputvortrag [Bü18; Go14].

Ein Vorteil dieser Variante ist, dass die Schülerinnen und Schüler bei schwerwiegenden Problemen die Lehrkraft direkt fragen können. Zudem hat die Lehrkraft die Sicherheit, dass die Schülerinnen und Schüler das Erklärvideo wirklich anschauen und bearbeiten [Go14]. Andererseits verliert man den Vorteil des Zeitgewinns des klassischen Inverted Classrooms [Go14].

3 Unterrichtsablauf und Material

3.1 Kompetenzen

Die durch die Unterrichtseinheit zu erwerbenden Kompetenzen wurden aus dem Lehrplan abgeleitet, in dem die grundlegende Kompetenz „Binäre Darstellung von Daten erläutern“ gefordert wird [Rh19]. Zur Überprüfung und Messung dieser Kompetenz wurden die folgenden spezifischen Kompetenzen gewählt: Die Schülerinnen und Schüler können ...

- ... eine mindestens sechsstellige Binärzahl in eine Dezimalzahl umwandeln.
- ... eine Dezimalzahl (mindestens bis 100) in eine Binärzahl umwandeln.
- ... ohne Hilfsmittel erklären, was eine Binärzahl ist und wie diese aufgebaut ist.

3.2 Erklärvideoproduktion

Um diese Kompetenzen zu erreichen, wurden zunächst zwei Erklärvideos produziert. Dazu galt es zuerst die Artefakte auszuwählen. Die Wahl fiel auf die Punktekarten der Abenteuer Informatik Ausstellung sowie auf Lämpchenkarten, welche an die binäre Uhr der Ausstellung angelehnt sind [Ga17]. Zudem wurden Stift und Papier als Artefakte angesehen. Aufgrund des gewünschten Einsatzes der Artefakte bot sich die Erklärvideo Art der Legetechnik an, da so die Artefakte wirklich eingesetzt und verwendet werden konnten. Zuerst wurde dazu ein Storyboard geschrieben, welches für jede Szene den Sprechtext und die dazugehörigen Bilder und Aktionen beschreibt. Für das charakteristische Merkmal des Storytellings von Erklärvideos wurde eine fiktive Rahmengeschichte erfunden. Diese handelt von zwei Freunden, Anna und Ben, welche mit den Artefakten die binäre Uhr der Abenteuer Informatik Ausstellung erforschen.

Der Grundablauf ist bei beiden Erklärvideos identisch. Dadurch liegt der Unterschied lediglich im Einsatz der Artefakte. Dies bedeutet, dass in dem Erklärvideo mit Einsatz von Artefakten an einigen Stellen konkrete Handlungsanweisungen für die Schülerinnen und Schüler gegeben werden. Diese sind beispielsweise die benötigten Karten auszuschneiden, mit diesen Aufgaben zu lösen, sowie mit Stift und Papier die Stellenwerttabelle zu erstellen und sich zu überlegen, wie diese gebildet wird. Im Erklärvideo ohne Artefakte fallen diese Szenen weg, sodass die Schülerinnen und Schüler nicht aktiv am Erklärvideo teilnehmen. Damit jedoch die Lernenden ohne Artefakte nicht per se weniger Übungsaufgaben bekommen, rechnen sie im Anschluss an das Erklärvideo die gleichen Aufgaben als „Check-Up Aufgaben“.

Die beiden Erklärvideos findet man unter http://www.abenteuerinformatik.de/video/binaer_mit.mp4 und http://www.abenteuerinformatik.de/video/binaer_ohne.mp4.

3.3 Unterrichtsablauf

Im Unterricht werden die Videos mit Flipped Classroom eingesetzt. In der Selbststudienphase betrachten dabei die Schülerinnen und Schüler das jeweilige Erklärvideo entweder mit oder ohne Artefakte. In der anschließenden Präsenzphase findet dann eine identische Stationenarbeit statt. Es sei angemerkt, dass dabei inhaltlich wenig Schülerzentrierung vorhanden ist, da die Stationen vorgegeben sind. Es gibt sieben Stationen, bei denen zwei Pflicht sind, um die grundlegenden Umrechenarten zu üben. Danach haben die Lernenden die Möglichkeit frei aus den Stationen auszuwählen. Es wird darauf geachtet, dass die Stationen möglichst unterschiedliche Lerntypen ansprechen. So ist die dritte Station spielerisch angelegt, da es sich um zwei verschieden schwere Memorys aus Binärzahl-Dezimalzahl Paaren handelt. Die Schülerinnen und Schüler können also ihrem geschätzten Leistungsstand entsprechend auswählen, welche Variante sie zum Üben verwenden. Für die vierte Station wurde ein weiteres kurzes Erklärvideo zum „Finger Binärsystem“ produziert, welches auf http://www.abenteuerinformatik.de/video/finger_binaer.mp4 zu finden ist. Jeder Finger stellt bei diesem eine andere Zweierpotenz dar, was in dem Erklärvideo veranschaulicht wurde. Die zwei Zustände Eins und Null werden durch einen ausgestreckten oder nicht ausgestreckten Finger dargestellt. Hierbei werden haptisch lernende Schülerinnen und Schüler angesprochen, da sie mit den eigenen Fingern binär zählen lernen. Dieses Zählen wird in Station fünf noch näher beleuchtet. Dazu wurde eine Stellenwerttabelle als Arbeitsblatt erstellt, in die die Schülerinnen und Schüler die Binärzahlen zu den Dezimalzahlen eins bis 20 schreiben und Regelmäßigkeiten erforschen sollen. Wieder eher spielerisch ist die sechste Station. Dabei geht es um einen Zaubertrick, der sich das Binärsystem zu Nutze macht. Die Lernenden erhalten dazu ebenfalls aus der Abenteuer Informatik Ausstellung die passenden Zauberkarten und erfahren das, was eine Freiwillige oder ein Freiwilliger beim Zaubertrick auch gesagt bekommt [Gal7]. Anschließend sollen sie knobeln und gemeinsam ausprobieren, wie der Trick funktioniert. In der letzten Station schließlich geht es um die Addition von Binärzahlen, die die Schülerinnen und Schüler herleiten sollen.

Während der Stationenarbeit liegen die Lösungen am Pult, sodass die Schülerinnen und Schüler selbstverantwortlich ihre eigenen Ergebnisse kontrollieren. Ebenso gibt es bei einigen Stationen gestaffelte Tippkarten, die schrittweise helfen. Diese Elemente dienen der Selbstverantwortlichkeit und Eigenständigkeit der Schülerinnen und Schüler.

Gegen Ende der Präsenzphase sind zehn Minuten zur Besprechung im Plenum eingeplant. An zwei Beispielen werden die grundlegenden Umrechenarten gemeinsam zusammengefasst, sowie allgemein beschrieben, was eine Binärzahl ist. Dazu wird parallel ein Tafelbild mit den beiden Musterlösungen erstellt.

Neben der klassischen Inverted Classroom Methode wird ebenfalls ein In-Class-Flip verwendet. Dazu bekommen alle Lernenden einen Computer und Kopfhörer, damit ein individuelles Lernen möglich ist. Die Präsenzphase findet direkt im Anschluss an die Selbststudienphase statt. Die beiden Durchführungen unterscheiden sich lediglich in der Inputphase, die anschließende Stationenarbeit in der Präsenzphase ist identisch. Es wurde

darauf geachtet, dass beide Gruppen in der Präsenzphase eine identische Zeitspanne für die Stationen zur Verfügung hatten, um sonst entstehende Vorteile einer längeren Übungszeit auszuschließen.

Alle Materialien der Stationenarbeit und die genauen Unterrichtsabläufe sind unter http://www.abenteuer-informatik.de/download/flippedclassroom_binaersystem.zip zur Nutzung im eigenen Unterricht zu finden.

3.4 Empirische Untersuchung

Untersucht wurde, ob sich der Einsatz von Artefakten im Vergleich ohne Artefakte positiv auf die Kompetenzentwicklung der Lernenden auswirkt. Außerdem war die Frage, ob das Schauen des Erklärvideos als Hausaufgabe beim klassischen Flipped Classroom erfolgsversprechender ist oder das Schauen im Unterricht beim In-Class-Flip.

Dazu wurden vier Parallelkursen eines städtischen Gymnasiums die unterschiedlichen Varianten zugewiesen. Es gab also jeweils einen Kurs mit dem klassischen Inverted Classroom mit Einsatz von Artefakten und ohne Artefakte, sowie jeweils einen In-Class-Flip Kurs mit und ohne Artefakte.

Um zu überprüfen, ob die Schülerinnen und Schüler bereits das Binärsystem kennen wurde ein Vortest durchgeführt. Neun Wochen später fand dann die eigentliche Doppelstunde mit der Stationenarbeit statt und in der Stunde darauf ein angekündigter erster Nachttest. Dieser war von den Aufgabentypen identisch mit denen des Vortests. Die erste Aufgabe bestand aus jeweils zwei Umrechenaufgaben einer Binärzahl in eine Dezimalzahl, während die zweite Aufgabe die Umrechnung zweier Dezimalzahlen in Binärzahlen war. Die dritte Aufgabe fragte nach einer möglichst genauen Beschreibung was eine Binärzahl ist und wie diese aufgebaut ist. Zwei Wochen nach dem ersten Nachttest fand die Kursarbeit der Schülerinnen und Schüler statt. Dabei wurde jeweils eine Umrechenart gestellt sowie die Transferaufgabe ein neues Stellenwertsystem zu einer anderen Basis zu beschreiben und eine Dezimalzahl darin darzustellen. Fünf Wochen nach dieser Kursarbeit, also insgesamt sieben Wochen nach dem ersten Nachttest, wurde ein zweiter unangekündigter Nachttest durchgeführt.

4 Ergebnisse

Trotz der geringen Teilnehmerzahl von 53 Schülerinnen und Schülern mit dementsprechend kleiner statistischer Relevanz, lassen sich folgende Tendenzen ablesen: Das Fazit des Vortests war, dass keine Schülerin und kein Schüler die Aufgaben absolut richtig lösen konnte. Es gab insgesamt fünf Schülerinnen und Schüler in allen Kursen, die jedoch ein gewisses Vorwissen zeigten. Dadurch ergab sich für den Kurs mit dem

klassischen Flipped Classroom ohne Artefakte¹ von zwölf möglichen Punkten eine durchschnittliche Gesamtpunktzahl von 0,83. Beim In-Class-Flip ohne Artefakte² wurden im Durchschnitt 0,85 Punkte erzielt, beim In-Class-Flip mit Artefakten³ 0,53 sowie beim klassischen Flipped Classroom mit Artefakten⁴ 0 Punkte.

Im folgenden Diagramm ist die anschließende Entwicklung der Schülerinnen und Schüler zwischen dem ersten und dem zweiten Nachtest dargestellt.

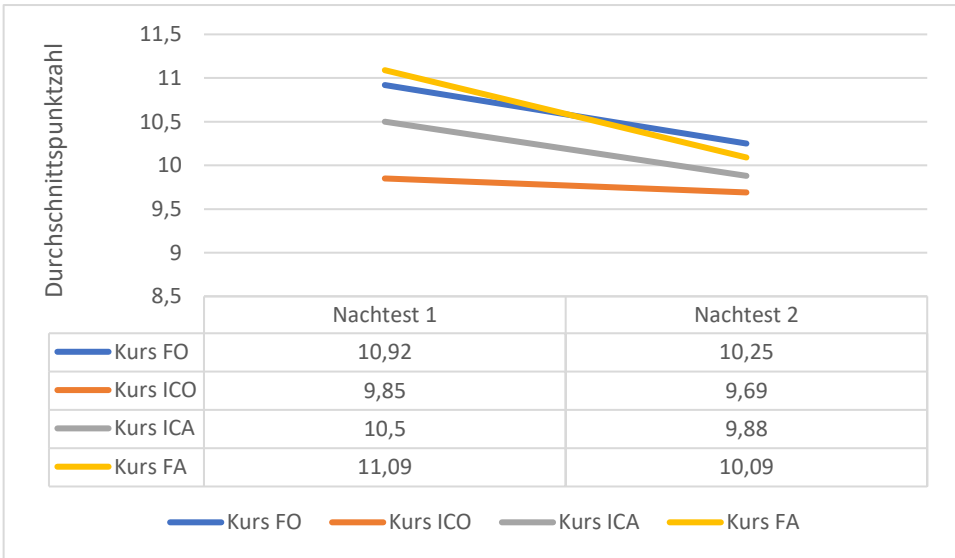


Abb. 1: Entwicklung der durchschnittlichen Punktzahl zwischen dem ersten und zweiten Nachtest

Bereits beim ersten Nachtest erzielten die Lernenden mit dem klassischen Flipped Classroom ein leicht besseres Ergebnis als jene mit dem In-Class-Flip. Außerdem gab es die Tendenz, dass innerhalb einer Flipped Classroom Variante der Kurs mit Artefakten eine leicht höhere Punktzahl erreichte. Jedoch gilt nicht pauschal die Aussage, dass Lernende mit Artefakten ein besseres Ergebnis erzielten als Lernende ohne Artefakte.

Im Vergleich dazu schnitten alle Kurse im zweiten Nachtest im Durchschnitt etwas schlechter ab als im Ersten. Die Ergebnisse befinden sich aber auf etwa einem Niveau zwischen 9,6 und 10,2. Dennoch waren die Kurse mit dem klassischen Inverted Classroom auch beim zweiten Nachtest noch leicht besser als die In-Class-Flip Kurse. Allerdings gilt nicht mehr pauschal die Tendenz, dass innerhalb der einzelnen Flips die Kurse mit Artefakten besser waren.

¹ Dieser Kurs wird im Folgenden Kurs FO benannt.

² Dieser Kurs wird im Folgenden Kurs ICO benannt.

³ Dieser Kurs wird im Folgenden Kurs ICA benannt.

⁴ Dieser Kurs wird im Folgenden Kurs FA benannt.

Die Ergebnisse der Kursarbeit unterstützen das Ergebnis des ersten Nachttests und weisen darauf hin, dass die beim ersten Nachttest erzielten Ergebnisse kein Zufall waren. So sind auch dort die identischen Tendenzen erkennbar wie beim ersten Nachttest. Der Vorsprung der Kurse mit Artefakten im Vergleich zu der direkten Flip Variante ohne Artefakte ist jedoch nicht bei allen Teilaufgaben so deutlich wie beim ersten Nachttest. Bei der Transferaufgabe ist diese Tendenz am meisten ausgeprägt. Besonders hervorzuheben ist, dass die Transferaufgabe von dem Kurs FA mit 2,18 von 4 Punkten am besten gelöst wurde, während danach der Kurs FO mit 1,79 auf gleicher Höhe mit dem Kurs ICA mit 1,71 Punkten folgt. Der Kurs ICO schließlich erzielte lediglich 1 von 4 Punkten.

5 Interpretation der Ergebnisse

Zuerst lässt sich festhalten, dass alle Schülerinnen und Schüler im Vergleich zum Vortest eine deutliche Kompetenzentwicklung gezeigt haben. Sowohl beim ersten Nachttest als auch beim Zweiten wurde im Durchschnitt eine erfreulich hohe Punktzahl erreicht. Dennoch gab es wohl den Effekt des Vergessens, da der zweite Nachttest in allen Kursen etwas schlechter war als der erste Nachttest.

Interessant ist die Abstufung der Ergebnisse im ersten Nachttest. Diese sind ein Indiz dafür, dass sich die Schülerinnen und Schüler mit dem klassischen Flipped Classroom zu Hause intensiver mit den Erklärvideos befasst haben als jene im In-Class-Flip, wodurch sie ein besseres Ergebnis erzielten. Diese Tendenz ist in allen drei Überprüfungen vorhanden.

Die Hypothese, dass die Nutzung von Artefakten eine bessere Kompetenzentwicklung begünstigen, lässt sich so jedoch in den vorliegenden Ergebnissen nicht bestätigen. Allerdings war beim ersten Nachttest die Tendenz vorhanden, dass immerhin im direkten Vergleich der Kurs ICA besser war als ICO und der Kurs FA besser war als der Kurs FO.

Dass der Kurs FA bei der Kursarbeit das beste Ergebnis erzielte, deutet zudem darauf hin, dass es möglich ist, dass die Verbindung von klassischem Flipped Classroom und Einsatz von Artefakten ein tieferes Verständnis bei den Schülerinnen und Schülern fördert.

6 Fazit

Auch wenn in den vorliegenden Ergebnissen der empirischen Untersuchung keine bahnbrechenden Effekte beim Einsatz mit Artefakten feststellbar waren, hat das vorgestellte Konzept dennoch seine Berechtigung. Die Methode des Flipped Classrooms bzw. In-Class-Flips in Verbindung mit den Erklärvideos und Artefakten ist eine gelungene Methode, die auf überwiegend große Begeisterung der Lernenden gestoßen ist.

Konkret die Nutzung der Artefakte spricht besonders die haptisch, enaktiv lernenden Schülerinnen und Schüler an, welche oft im alltäglichen Unterricht eher zu kurz kommen. Denkbar ist es, dass man den Lernenden beide Erklärvideos als Optionen zur Verfügung

stellt, sodass noch mehr der Forderung nach Eigenständigkeit nachgegangen wird. Dann haben die Schülerinnen und Schüler die Möglichkeit sich bewusst auszusuchen, ob sie lieber aktiv oder lieber passiv ein Erklärvideo betrachten.

Zusammenfassend lässt sich sagen, dass es für ein signifikant relevantes Ergebnis wünschenswert wäre, wenn an mehreren Schulen die Unterrichtssequenzen in der beschriebenen Form eingesetzt werden, um eine größere Stichprobe zu erhalten. Zudem wäre es sehr spannend zu sehen, welche Kompetenzen die Lernenden nach einem wesentlich längeren Zeitraum noch vorzeigen können und ob dann die Artefakte oder die Flip-Varianten vielleicht deutlichere Effekte erzielen. Es besteht also noch Forschungsbedarf. Die Pilotstudie hat aber dennoch ergeben, dass es sich um eine innovative, schüleraktivierende Methode handelt, die sinnvoll im Unterricht eingesetzt werden kann.

Literaturverzeichnis

- [Ab18] Abila, C.; Schallert, S.: SchülerInnenaktivierende Methoden in der Präsenzphase. In (Buchner, J. et al. Hrsg.): *Inverted Classroom. Vielfältiges Lernen Begleitband zur 7. Konferenz Inverted Classroom and Beyond 2018* FH St. Pölten, 20. & 21. Februar 2018, 2018; 15-19.
- [Be12] Bergmann, J.; Sams, A.: *Flip your classroom. Reach every student in every class every day.* International Society for Technology in Education, Eugene, Or, Alexandria, Va, 2012.
- [Br17] Brehmer, J.; Becker, S.: "Erklärvideos". ... als eine andere und/oder unterstützende Form der Lehre. Georg-August-Universität Göttingen, 2017.
- [Bü18] Bülls, O.; Freisleben-Teutscher, C.; Buchner, J.: Potentiale interaktiver Videos für das Inverted Classroom Model. In (Buchner, J. et al. Hrsg.): *Inverted Classroom. Vielfältiges Lernen Begleitband zur 7. Konferenz Inverted Classroom and Beyond 2018* FH St. Pölten, 20. & 21. Februar 2018, 2018.
- [Fä16] Fährnich; Thein: *Differenzieren mit Erklärvideos. Mehr Zeit im Unterricht*, 2016.
- [Fi18] Finger, A.; Wiegmann, J.: Biodiversität vor der Haustür: Einsatz von Simpleshows im Biologieunterricht. In (Meßinger-Koppelt, J.; Maxton-Küchenmeister, J. Hrsg.): *Naturwissenschaften digital. Toolbox für den Unterricht.* Joachim Herz Stiftung Verlag, Hamburg, 2018; S. 60–63.
- [Fi17] Finkenberg, F.; Trefzger, T.: *Flipped Classroom im Physikunterricht in der Oberstufe*, Dresden, 2017.
- [Fr16] Frank, C.: Handlungsorientierung im besten Sinne. Erklärvideos im Unterricht. In *Pädagogik*, 2016; S. 23–25.
- [Ga17] Gallenbacher, J.: *Abenteuer Informatik. IT zum Anfassen für alle von 9 bis 99 - vom Navi bis Social Media.* Springer, Berlin, 2017.

- [Go14] Gonzalez, J.: Modifying the Flipped Classroom: The "In-Class" Version. <https://www.edutopia.org/blog/flipped-classroom-in-class-version-jennifer-gonzalez>, 2014. 13.10.2018.
- [Ha12a] Handke, J.: ICM-Effekte in der Hochschullehre. In (Handke, J.; Sperl, A. Hrsg.): Das Inverted Classroom Model. Begleitband zur ersten deutschen ICM-Konferenz. De Gruyter, Berlin/Boston, 2012; S. 139–148.
- [Ha12b] Handke, J.: Voraussetzungen für das ICM. In (Handke, J.; Sperl, A. Hrsg.): Das Inverted Classroom Model. Begleitband zur ersten deutschen ICM-Konferenz. De Gruyter, Berlin/Boston, 2012; 39-52.
- [Ha17] Handke, J.: Handbuch Hochschullehre Digital. Leitfaden für eine moderne und mediengerechte Lehre. Tectum Verlag, Baden-Baden, 2017.
- [Ha18] Hahn, H.; Puschner, N.: Von der Präsenzvorlesung zu "Mini-Lectures" - Umsetzung eines Inverted-Classroom-Szenarios. In (Kortenkamp, U.; Kuzle, A. Hrsg.): Beiträge zum Mathematikunterricht 2017. Vorträge auf der 51. Tagung für Didaktik der Mathematik vom 27.02.2017 bis 02.03.2017 in Potsdam, 2018; S. 369–372.
- [Rh19] Rheinland-Pfalz Ministerium für Bildung, Wissenschaft, Jugend und Kultur. Lehrplan Informatik Grund- und Leistungsfach, ohne Jahresangabe. Zugriff über: <http://lehrplaene.bildung-rp.de/gehezu/startseite.html?keyword=informatik>, Stand 10.05.2019.
- [Sa12] Sams, A.: Der "Flipped" Classroom. In (Handke, J.; Sperl, A. Hrsg.): Das Inverted Classroom Model. Begleitband zur ersten deutschen ICM-Konferenz. De Gruyter, Berlin/Boston, 2012; S. 13–23
- [Sc12] Schäfer, A. M.: Das Inverted Classroom Model. In (Handke, J.; Sperl, A. Hrsg.): Das Inverted Classroom Model. Begleitband zur ersten deutschen ICM-Konferenz. De Gruyter, Berlin/Boston, 2012; S. 3–11.
- [Sc13a] Schön, S.; Ebner, M.: Gute Lernvideos. ... so gelingen Web-Videos zum Lernen! Books on Demand, Norderstedt, 2013.
- [Sc13b] Schön, S.: Klappe zu! Film ab! - Gute Lernvideos kinderleicht erstellen. In (Pauschenwein, J. Hrsg.): Lernen mit Videos und Spielen. Tagungsband zum 12. E-Learning-Tag der FH Joanneum am 18.9.2013. Graz, 2013; S. 3–9.
- [Sc16] Schlegel, F. Schlegel, F.: Erklärvideos im Unterricht. Einstieg in die Filmbildung mit YouTube-Formaten. - Workshop für Lehrkräfte und MedienberaterInnen -. Film + Schule NRW, Münster, 2016.
- [St10] Stelmes, C.; Linckels, S.; Meinel, C.: Digitale Videos im Unterricht. Die "YouTube-Generation" lernt Judo mit E-Videos. In LOG IN, 2010; S. 22–28.
- [We12] Weidmann, D.: Das ICM als Chance für die individuelle Förderung von Schülern? In (Handke, J.; Sperl, A. Hrsg.): Das Inverted Classroom Model. Begleitband zur ersten deutschen ICM-Konferenz. De Gruyter, Berlin/Boston, 2012; 53-70.
- [Wo15] Wolf, K. D.: Video-Tutorials und Erklärvideos als Gegenstand, Methode und Ziel der Medien- und Filmbildung. In (Hartung, A. et al. Hrsg.): Filmbildung im Wandel. new academic press, Wien, 2015; S. 121–131.

Zur Entwicklung des Masterstudiengangs «Medien und Informatik»

Beat Döbeli Honegger¹, Martin Hermida², Regina Schmid³

Abstract: Seit 2017 wird nach und nach in allen 21 deutschsprachigen Kantonen der Schweiz der Lehrplan 21 eingeführt. Mit dem darin enthaltenen Modul «Medien und Informatik» betritt die Schweiz im deutschsprachigen Raum Neuland. Doch fehlen derzeit nicht nur Lehrpersonen mit den notwendigen Kompetenzen, sondern auch entsprechend ausgebildete Expertinnen und Experten zur Aus- und Weiterbildung dieser Lehrpersonen sowie zum Aufbau einer Fachdidaktik «Medien und Informatik». Mit dem «Joint Degree Masterstudiengang in Fachdidaktik Medien und Informatik» versuchen die Pädagogische Hochschule Schwyz, die Universität Zürich, die Fachhochschule Luzern und die Pädagogische Hochschule Luzern diesem Mangel zu begegnen und die Grundlagen einer entsprechenden Fachdidaktik aufzubauen. Der vorliegende Beitrag beschreibt die Konzeption des Masterstudiengangs und die Herausforderungen anlässlich der ersten Durchführung

Keywords: Medien und Informatik, Fachdidaktik, Lehrplan 21, LehrerInnen-Bildung

1 Der Themenbereich «Medien und Informatik» im Lehrplan 21

Am 31.10.2014 hat die Deutschschweizer Erziehungsdirektorenkonferenz (D-EDK) den Lehrplan 21, der vom Kindergarten bis zum Ende der obligatorischen Schulzeit alle Fächer umfasst, zuhänden der Kantone verabschiedet [DEDK15a]. Unter anderem enthält der Lehrplan 21 auch das Modul «Medien und Informatik» [DEDK15b], das sich aus verschiedenen Perspektiven mit digitalen Phänomenen beschäftigt. Ähnlich wie die Dagstuhl-Erklärung [Br16], die eine technologische, eine gesellschaftlich-kulturelle und eine Anwendungsperspektive vorsieht, sind es im Lehrplan 21 die drei Bereiche Medien, Informatik und Anwendung. Der Lehrplan sieht dabei vor, dass die Bereiche Medien und Informatik in einem eigenen Gefäss vermittelt werden, Anwendungskompetenzen hingegen vorwiegend in anderen Fächern. Obwohl der Lehrplan 21 ausschliesslich eine Empfehlung an die Kantone ist, haben alle 21 mindestens teilweise deutschsprachigen Kantone beschlossen, den Lehrplan 21 einzuführen. Gewisse Kantone sehen dabei für «Medien und Informatik» bereits ab der 5. Klasse ein eigenes Zeitgefäss (meist im Umfang einer Wochenlektion) vor. Andere Kantone integrieren die Themen in der 5. und 6. Klasse in bestehende Fächer und beginnen erst ab der 7. Klasse mit einem Zeitgefäss für Medien und Informatik (meist 2 Jahreslektionen verteilt auf die Klassenstufen 7-9).

¹ Pädagogische Hochschule Schwyz, IMS, Zaystrasse 42, CH-6410 Goldau, beat.doebeli@phsz.ch

² Pädagogische Hochschule Schwyz, IMS, Zaystrasse 42, CH-6410 Goldau, martin.hermida@phsz.ch

³ Pädagogische Hochschule Schwyz, IMS, Zaystrasse 42, CH-6410 Goldau, regina.schmid@phsz.ch

2 Aus- und Weiterbildungsbedarf an Pädagogischen Hochschulen

Unabhängig davon, ob die Kompetenzen von «Medien und Informatik» in einem eigenen Zeitgefäss oder integriert in bisherigen Fächern vermittelt werden sollen, benötigen Lehrerinnen und Lehrer in allen drei Bereichen Kompetenzen, über die sie jedoch häufig noch nicht verfügen. Im Gegensatz zu den letzten Einführungen von neuen Fächern in der Schweizer Volksschule (Frühfranzösisch und Frühenglisch) fehlen den Lehrpersonen meist nicht nur die entsprechenden didaktischen Kompetenzen, sondern auch die zugrunde liegenden fachlichen Kompetenzen. Ferner fehlen ihnen Unterrichtserfahrungen aus der eigenen Schulzeit [Dö15]. Abbildung 1 zeigt die erforderlichen Kompetenzbereiche, wobei die ebenfalls notwendigen mediendidaktischen Kompetenzen in allen Fächern und nicht nur bei «Medien und Informatik» eine Rolle spielen.

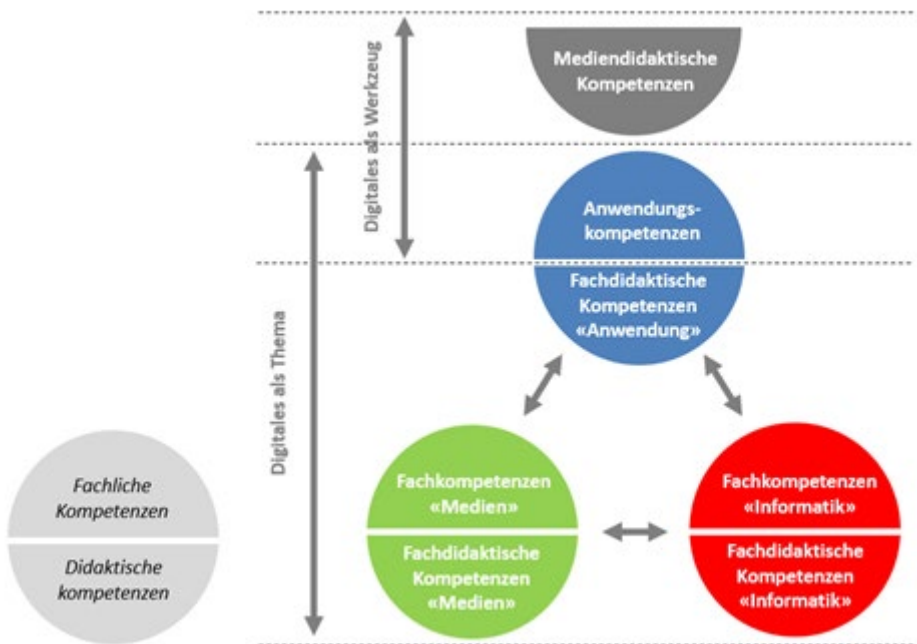


Fig. 1: Notwendige digitale Kompetenzen von Lehrpersonen [Dö15]

Die meisten Kantone haben unterdessen entschieden, wie und ab wann der Lehrplan 21 und damit auch «Medien und Informatik» eingeführt werden soll [Hä18]. Die Pädagogischen Hochschulen der deutschsprachigen Schweiz wiederum haben mit der Ausbildung künftiger und der Weiterbildung bestehender Lehrpersonen begonnen (erste Berichte siehe [DH17] und [LR18]).

Pädagogische Hochschulen in der Deutschschweiz sind derzeit stark gefordert und können insbesondere die Weiterbildungsnachfrage nicht immer vollständig befriedigen. Zudem zeigt sich, dass auch viele im Bereich «Medien und Informatik» tätige Mitarbeitende Kompetenzdefizite aufweisen. Darüber hinaus handelt es sich bei Medien und Informatik um bisher eher getrennt agierende Bereiche, deren Zusammenlegung zwar einerseits Synergien, andererseits aber auch Reibungsflächen bietet. Eine gemeinsame Fachdidaktik mit einem geteilten Fachverständnis fehlt bisher weitgehend [DM15].

3 Der «Master of Arts in Fachdidaktik Medien und Informatik»

Ausgehend vom geschilderten Bedarf nach einer Qualifizierung von Mitarbeitenden pädagogischer Hochschulen im Bereich «Medien und Informatik» hat die Pädagogische Hochschule Schwyz als Leading House zusammen mit der Universität Zürich, der Fachhochschule Luzern und der Pädagogischen Hochschule Luzern im Rahmen des Bundesprogramms zum «Aufbau der wissenschaftlichen Kompetenzen in den Fachdidaktiken 2017-2020» [Sw17] einen entsprechenden Masterstudiengang konzipiert.

3.1 Studienumfang und Berufsmöglichkeiten

Der «Master of Arts in Fachdidaktik Medien und Informatik» umfasst mindestens 90 ECTS-Punkte und dauert in der Regel drei Jahre. Studierende mit einer Vorbildung in Medien- und Kommunikationswissenschaft, Informatik, Erziehungswissenschaften oder einem Lehrdiplom auf Primar- und Sekundarschulstufe I können sich für den Masterstudiengang bewerben. Weisen Sie nicht alle erforderlichen Qualifikationen zur Aufnahme eines Masterstudiengangs auf, so müssen sie zusätzliche vorbereitende oder ergänzende Leistungen erbringen. Der Masterstudiengang qualifiziert zur Dozierentätigkeit in Aus- und Weiterbildung an Pädagogischen Hochschulen, zur Beratung von Schulleitungen und Lehrpersonen, zur Entwicklung von Lehrmitteln und zur Forschungstätigkeit. Der Studiengang ist jedoch kein Lehramtsstudium, d.h. er ist weder auf die Tätigkeit als Lehrperson ausgerichtet, noch ist mit ihm eine formale Unterrichtsberechtigung auf der Zielstufe verbunden.

3.2 Studienstruktur

Das Studium setzt sich aus den drei Bereichen «Fachwissenschaftliche Studien», «Fachdidaktische Studien» und «Erziehungswissenschaftliche Studien» zusammen und widerspiegelt damit den oben beschriebenen Bedarf nach sowohl fachlichen als auch fachdidaktischen Kompetenzen. Abbildung 2 zeigt eine vereinfachte Darstellung der Studienstruktur.

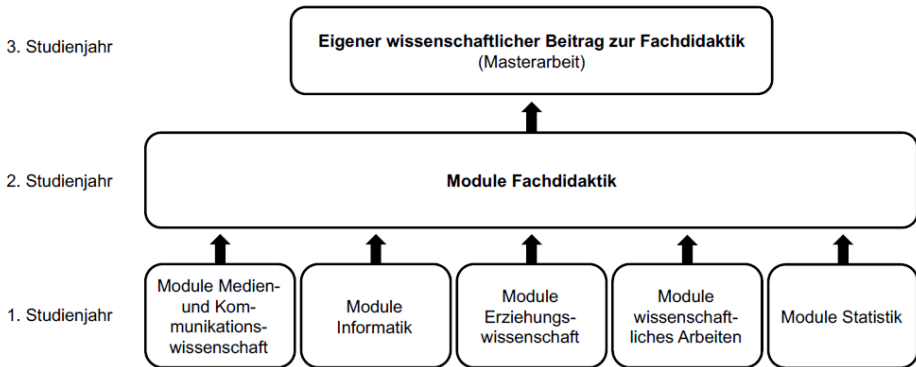


Fig. 2: Studienstruktur des Fachdidaktikmasters «Medien und Informatik»

3.3 Erstmalige Durchführung

Nach einer sehr kurzen Aufbauphase konnte der Masterstudiengang im September 2018 mit rund 30 Studierenden in einer ersten Durchführung starten. Für die im September 2019 startende zweite Durchführung liegen 20 Anmeldungen vor. Nach Abschluss des ersten Studienjahres lassen sich Herausforderungen in drei Bereichen beschreiben. Inhaltlich sind insbesondere die Anbieterinnen und Anbieter dieses Masterstudiengangs gefordert, inhaltliche Kohärenz in diesem neuen Fachgebiet herzustellen. Für die Studierenden zeigt sich die Multidisziplinarität als eine der Herausforderungen in ihrem Studium. Weiter gilt es, im Zusammenspiel von Hochschulen und Studierenden, unterschiedliche organisatorische Strukturen in Einklang zu bringen.

3.3.1 Inhaltliche Herausforderungen

Obwohl die Themen Medien und Informatik künftig fester Bestandteil der obligatorischen Volksschule sein werden, ist die Fachdidaktik Medien und Informatik erst im Aufbau [DM15]. Dieser Aufbauprozess muss auf drei Ebenen stattfinden. Auf einer ersten inhaltlichen Ebene muss festgelegt werden, welche Themen aus den Fachwissenschaften im Unterricht behandelt werden sollen und inwiefern diese Themen aus den beiden Fachwissenschaften Berührungspunkte aufweisen. Darauf aufbauend müssen auf einer zweiten Ebene anerkannte Bildungsstandards definiert werden, damit Unterricht auf diese ausgerichtet und evaluiert werden kann. Auf der dritten Ebene muss schliesslich das eigentliche fachdidaktische Handwerk erarbeitet werden, um die definierten Bildungsstandards im Unterricht zu erreichen.

Die Herausforderung besteht nun darin, dass zwar mit dem Lehrplan «Medien und Informatik» eine erste, mindestens teilweise bildungspolitisch beeinflusste Setzung auf der ersten Ebene erfolgt ist, die anderen beiden Ebenen sich aber erst entwickeln

müssen. Der Fachdidaktikmaster steht somit vor der Aufgabe, Elemente zu vermitteln, die teilweise erst erarbeitet werden. Im besten Fall ist der Fachdidaktikmaster eine hilfreiche Plattform zur Entwicklung und Etablierung dieser Ebenen.

3.3.2 Herausforderungen aus Studierendenperspektive

Bevor sich die Studierenden mit der eigentlichen Fachdidaktik auseinandersetzen können, müssen sie sich die Grundlagen in den beiden Fachwissenschaften aneignen. Sie steigen damit in ein fachdidaktisches Studium ein, befassen sich aber vorerst mit rein fachwissenschaftlichen Inhalten ohne konkreten Bezug zur didaktischen Umsetzung und zur Zielstufe. Diese Vertiefung in die Fachwissenschaften ist für die professionelle Reife zentral, erfordert aber auch Ausdauer seitens der Studierenden. Zudem arbeiten einige Studierende bereits im Berufsfeld, jedoch ohne passende formale Qualifikation. Sie müssen nun ihre Tätigkeit notgedrungen auch aus anderer Perspektive betrachten, da die zunehmende Standardisierung und stärkere Anbindung an die Fachwissenschaften neue Massstäbe und Schwerpunkte setzen. Noch mehr Studierende des ersten Jahrganges sind mit dem Anspruch der modernen Hochschullehre konfrontiert, dass Forschung und Entwicklung in engem Bezug zueinander stehen sollen. Diese Studierenden müssen sich nun zusätzlich Kompetenzen in den sozialwissenschaftlichen Forschungsmethoden aneignen.

3.3.3 Organisatorische Herausforderungen

Der Fachdidaktikmaster «Medien und Informatik» wird von vier Hochschulen getragen und hat in der ersten Durchführung Studierende mit sehr unterschiedlicher fachlicher Herkunft und Vorwissen angezogen. Dadurch war ein überdurchschnittlicher Koordinationsaufwand zum Abgleich aller Studienanforderungen und Reglemente der vier beteiligten Hochschulen notwendig. Die grosse Heterogenität der Studierenden erforderte oft individuelle Abklärungen sur dossier mit entsprechendem Aufwand. Als weitere organisatorische Herausforderung zeigt sich die terminliche Koordination der Lehrveranstaltungen zwischen vier Hochschulen. Es musste sichergestellt werden, dass die Module innerhalb zweieinhalb Präsenztage pro Woche stattfinden, damit das Studium auch als Teilzeitstudium berufs- und/oder familienbegleitend absolviert werden kann.

4 Erstes Zwischenfazit

Trotz den beschriebenen Herausforderungen ist ein erfolgreicher Start des Fachdidaktikmasters «Medien und Informatik» gelungen. Somit zeigt sich an einem kleinen Beispiel, dass neue Fächer nicht zwingend an fehlenden Kompetenzen in den Institutionen der Lehrerinnen- und Lehrerbildung scheitern müssen, sondern ein rascher Aufbau entsprechender Qualifikationsmassnahmen möglich ist. Es ist zu hoffen, dass dieser Fachdidaktikmaster mithelfen kann, eine inhaltlich und methodisch kohärente Fachdidaktik «Medien und Informatik» aufzubauen.

Literaturverzeichnis

- [Br16] Brinda, T. et al: Bildung in der digitalen vernetzten Welt – Dagstuhl-Erklärung. Deutsche Gesellschaft für Informatik, 2016. <http://tinyurl.com/dagstuhl3eck>
<https://doebe.li/t18567>
- [DEDK15a] Deutschschweizer Erziehungsdirektorenkonferenz (D-EDK): Lehrplan 21. Bereinigte Fassung, 2015. <https://www.lehrplan.ch>
<https://doebe.li/b5500>
- [DEDK15b] Deutschschweizer Erziehungsdirektorenkonferenz (D-EDK): Medien und Informatik. In: D-EDK: Lehrplan 21. Bereinigte Fassung, 2015.
https://projekt.lehrplan.ch/lehrplan/V5/ablage/FS1E_Modul_MI.pdf
<https://doebe.li/t17600>
- [DM15] Döbeli Honegger, B. & Merz, T.: Fachdidaktik Medien und Informatik - Ein Beitrag zur Standortbestimmung. In: Beiträge zur Lehrerinnen- und Lehrerbildung 33 (2015) 2, S. 256-263
<https://doebe.li/t18200>
- [Dö15] Döbeli Honegger, B.: Digitale Kompetenzen von Lehrpersonen für den Lehrplan 21, 2015. <https://doebe.li/t17550>
- [Dö16] Döbeli Honegger, B.: Mehr als 0 und 1. bern: hep Verlag, 2016.
<https://doebe.li/b6000>
- [DH17] Döbeli Honegger, B. & Hielscher, M.: Vom Lehrplan zur Lehrerinnenbildung. Erste Erfahrungen mit obligatorischer Informatikdidaktik für angehende Schweizer PrimarlehrerInnen. In: INFOS 2017, Lecture Notes in Informatics (LN) - Proceedings Series der Gesellschaft für Informatik (GI) Volume P-274, S. 97-107
<https://doebe.li/t16600>
- [HLF18] Hänggli, R.; Lenz, F. & Frey, T.: Umsetzung mediale und digitale Bildung in den Kantonen. Universität Freiburg, 2018.
<https://doebe.li/t20968>
- [LR18] Lamprou, A. & Repenning, A.: Teaching How to Teach Computational Thinking. In: ITiCSE 2018 Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education Pages 69-74
<https://doebe.li/t20894>
- [Sw17] swissuniversities: P-9: Aufbau der wissenschaftlichen Kompetenzen in den Fachdidaktiken 2017-2020.
<https://www.swissuniversities.ch/de/organisation/projekte-und-programme/p-9/>

Empfehlungen der GI – Kompetenzen für informatische Bildung im Primarbereich

Ausgestaltung der Empfehlungen der Gesellschaft für Informatik

Ludger Humbert,¹ Henry Herper, Alexander Best, Christian Borowski, Rita Freudenberg, Martin Fricke, Kathrin Haselmeier, Volkmar Hinz, Dorothee Müller, Andreas Schwill, Marco Thomas

Abstract: Vorstellung der Ergebnisse des Arbeitskreises »Bildungsstandards Informatik für den Primarbereich«. Darstellung und Diskussion der Motivation und der Restriktionen, die die »Standards« enthalten. Vorstellung und Besprechung von Unterrichtsbeispielen zur Umsetzung von Aspekten, die die Kompetenzentwicklung der Schülerinnen und Schüler befördern.²

Keywords: Primarbereich; Primarstufe; Grundschule; Informatikkompetenzen; informatische Allgemeinbildung; Phänomene, informatisch betrachtet; Informatikglossar; Unterrichtsbeispiele; digitale Bildung

1 Motivation

Informatische Bildung muss Eingang in die allgemeine Bildung finden, um dem gesellschaftlichen Bedarf nach einer fundierten, fachbezogenen Grundlage für die zunehmend von Informatik durchdrungene Welt einzulösen. Dass es dazu verpflichtende Informatik Elemente in jeder Schule braucht, findet zunehmend breitere Zustimmung und führt(e) in einigen Ländern zur Einrichtung eines eigenen, neuen Schulfachs in der Grundschule [Be18; Ga13].

Im Jahr 2000 wurde die vom Fachausschuss »Informatische Bildung an Schulen (IBS)« erarbeitete »Empfehlung für ein Gesamtkonzept zur informatischen Bildung an allgemein bildenden Schulen« [GI00] vorgelegt. Folgend wurden für alle weiterbildenden Schulen die *Grundsätze und Standards für die Informatik in der Schule – Bildungsstandards Informatik für die Sekundarstufe I* [GI08] ausgearbeitet und 2008 als Empfehlungen der GI verabschiedet. Die Empfehlungen für *Grundsätze und Standards für die Informatik in der Schule – Bildungsstandards Informatik für die Sekundarstufe II* [GI16] führten die durch das Gesamtkonzept implizierte Linie fort und gestalteten sie so weiter aus.

¹ Didaktik der Informatik, Wuppertal humbert@uni-wuppertal.de

² Dieser Beitrag enthält Elemente aus dem Projekt »Informatik an Grundschulen (IaG)«, die durch Unterstützung des Ministeriums für Schule und Bildung des Landes Nordrhein-Westfalen (MSB) realisiert werden konnten [MS17].

Mit den *Kompetenzen für informatische Bildung im Primarbereich* [GI19b] liegen numehr verabschiedete Empfehlungen auch für Anforderungen an die informatische Bildung für den Primarbereich bis zum Ende der Grundschule (4. Klasse) vor.

2 Einordnung

Informatik trägt wesentlich zur Allgemeinbildung bei. Es ist Aufgabe der Grundschule, die Fähigkeiten, Interessen und Neigungen von Kindern aufzugreifen und sie mit den Anforderungen fachlichen und fachübergreifenden Lernens zu verbinden. Eine bewusste Teilnahme am Leben in unserer Gesellschaft, aber auch die konstruktive Mitgestaltung der Lebenswelt setzen zunehmend informatische Kompetenzen voraus. Um Informatik für Kinder als kreativen Gestaltungsbereich fürs Problemlösen zugänglich zu machen, bedarf es einer altersgerechten Einbettung in den Primarbereich. Dies muss durch didaktisch gestaltete Fachkonzepte zur Erläuterung »informatischer Phänomene« erfolgen.

Die Empfehlungen für die *Kompetenzen für informatische Bildung im Primarbereich* stellen einen weiteren Baustein zur informatischen Bildung bereit. Alle drei Empfehlungen der GI detaillieren Kompetenzen für die fünf Prozess- und Inhaltsbereiche, die in Abbildung 1 dargestellt werden.



Abb. 1: Kompetenzstrukturmodell: Prozess- und Inhaltsbereiche

Damit liegt ein vollständiges Konzept für eine zeitgemäße und fachlich fundierte informatische Bildung für die formale Bildung vom Primarbereich bis zum allgemeinbildenden Abitur vor [vgl. GI19a].

3 Gestaltungsherausforderungen

Die für den Primarbereich vorliegenden Empfehlungen gilt es in der Grundschule umzusetzen. Dazu wurden während der Erarbeitungsphase Vorschläge diskutiert, allerdings muss – im Unterschied zur weiterführenden Schule und zur gymnasialen Oberstufe – davon ausgegangen werden, dass absehbar in Deutschland kein eigenes Schulfach Informatik in der Grundschule eingerichtet wird. Damit fehlen für den Informatikunterricht in der Grundschule – voraussichtlich auch in der Zukunft – primär qualifizierte Lehrkräfte. Ebenfalls fehlen Unterrichtserfahrungen, die – in der Breite vorhanden – Grundlage für Entwicklung und Gestaltung des Informatikunterrichts in der Grundschule darstellen und so einen wesentlichen Beitrag für die Unterrichtspraxis liefern würden. Umso wichtiger ist es, dass Elemente gestaltet, erprobt und evaluiert werden, die auf einer fachdidaktischen Basis zum Gelingen informatischer Bildung in der Grundschule beitragen.

Es ist unabdingbar, dass für die Gestaltung von Unterrichtselementen die Expertise von Grundschullehrkräften berücksichtigt wird und die Umsetzung in verschiedenen Kontexten erprobt wird, damit vermieden werden kann, dass unterrichtliche Ideen zwar fachlichen Anforderungen genügen, nicht aber in den Grundschulen umgesetzt werden (können). Daher stellen wir Elemente vor, die Zugänge zur Informatik bereitstellen, die in verschiedenen unterrichtlichen Kontexten erprobt wurden und es erlauben, informatische Kompetenzen bei den Kindern in der Grundschule zu entwickeln.

Beispiele³

Um die Erreichung der Kompetenzen zu befördern, bedarf es konkreter, erprobter Beispiele. Mit der Veröffentlichung der Entwurfsfassung der Kompetenzen in der LOG IN wurden exemplarisch Elemente und Ideen für die Umsetzung vorgestellt [HK18]. In dem Projekt »Informatik an Grundschulen (IaG)« des nordrhein-westfälischen Schulministeriums wurden darüber hinaus drei Module zur Umsetzung der Kompetenzen entwickelt, an Grundschulen erprobt und evaluiert [Ma18]. Dabei wurde insbesondere die Randbedingung, dass in diesem Projekt keine Informatiksysteme eingesetzt werden, berücksichtigt.

Damit diese Rahmenbedingung konstruktiv eingelöst werden kann, wurde mit [Hu18] eine pointierte Sicht auf die Kompetenzentwicklung ohne Informatiksysteme sowohl in den historischen Kontext eingeordnet als auch beispielhaft dargestellt. Zu dem Projekt IaG [MS17] wurden vom Schulministerium des Landes Nordrhein-Westfalen Lehrerhandreichungen, Hinweise zur Konzeption von Fortbildungsmaßnahmen sowie Materialien für die Schülerinnen und Schüler öffentlich bereitgestellt [Hu19b]. Darüber hinaus wurden von Teilnehmenden an der Erarbeitung der Kompetenzen parallel Forschungs- und Entwicklungsprozesse angestoßen, die zu weiteren Unterrichtsbeispielen führten [Ar16].

³ Teile dieses Abschnitts wurden ebenfalls in »Informatik – Kompetenzentwicklung bei Kindern« [Hu19a] veröffentlicht.

Nachfolgend werden Beispiele für den Primarbereich zu Automatenmodellen (Inhaltsbereich »Sprachen und Automaten«) und zur Kryptologie (Inhaltsbereich »Information und Daten«) skizziert. Während Automatenmodelle bislang nicht in der Grundschule thematisiert werden und Grundschullehrpersonen unbekannt sind, werden Themen zur Kryptologie bereits rudimentär im Sach-, Deutsch- und/oder Mathematikunterricht behandelt. Die Beispiele wurden mit (angehenden) Grundschullehrpersonen entwickelt und teilweise im Unterricht erprobt sowie evaluiert.

Automatenmodelle

Die Automatentheorie zeichnet sich durch einen starken Abstraktionsgrad aus, was wiederum hohe Anforderungen an die didaktische Aufbereitung entsprechender Themen für die Grundschule stellt. Schülerinnen und Schüler begegnen Automaten regelmäßig in ihrem Alltag, etwa über Getränkeautomaten [GI19b, S. 10]. Dass sich deren Abläufe formalisieren, über Automatenmodelle modellieren und mittels Zustandsübergangsdiagrammen oder -tabellen darstellen lassen, ist ihnen hingegen nicht bewusst. Alltagsautomaten stellen für die Schülerinnen und Schüler somit eine **Black-Box** dar, mit der sie interagieren, deren Funktionsweise sie jedoch nicht nachvollziehen können.

Ein schülerorientierter Kontext ist **Passwortsicherheit**. Der Sicherheitsgrad eines Passworts kann anhand der Anzahl umgesetzter Kriterien definiert werden. Der Sicherheitsgrad sei über die Zustände *rot*, *gelb* und *grün* definiert. Die Anzahl erfüllter Kriterien stellt die Zustandsübergänge dar. Kriterien könnten *Ziffern* oder *Klein- und Großbuchstaben* sein. Neben der Bewertung des Sicherheitsgrads stellen die Schülerinnen und Schüler den Bewertungsvorgang auch in einem Zustandsübergangsdiagramm dar, um erstens die Automatisierbarkeit der Bewertung zu verstehen, zweitens den Bewertungsvorgang zu verinnerlichen und drittens das Zustandsmodell als kommunikationsstützende Notation zu erfahren.

Die Schülerinnen und Schüler werden in die Lage versetzt, das Zustandsübergangsdiagramm in ein mentales Modell und ihr mentales Modell in ein entsprechendes Zustandsübergangsdiagramm zu überführen. Zunächst wird das vorgestellte Automatenmodell als Akzeptor aufgefasst, mit dem die Schülerinnen und Schüler exemplarisch Passwörter auf ihren Sicherheitsgrad, also die Anzahl erfüllter Kriterien, prüfen. Anschließend erweitern sie das Automatenmodell um selbstgewählte Kriterien sowie Zustände und modellieren diese im bestehenden Zustandsübergangsdiagramm. Dies könnte etwa über den Zustand *orange* und das Kriterium *Wortlänge > 7 Zeichen* erfolgen.

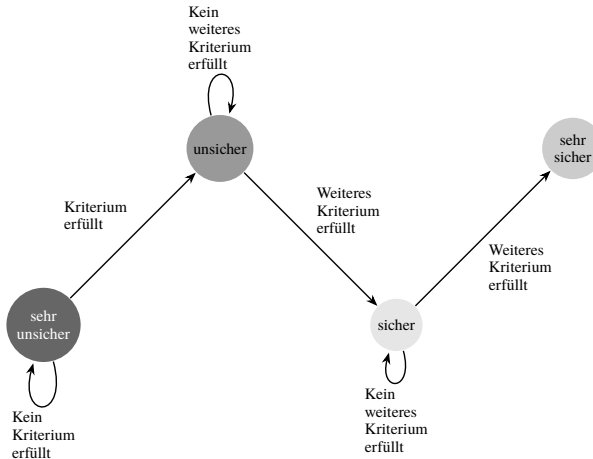


Abb. 2: Zustandsübergangsdiagramm für die Bewertung eines Passworts über drei Sicherheitskriterien

Kryptologie

Kinder im Grundschulalter nutzen bereits einfache kryptologische Verfahren, um geheime Botschaften untereinander auszutauschen. Dazu zählen sowohl Transpositions- als auch Substitutionsverfahren. Anknüpfend an diese Vorerfahrungen können zunächst zwei einfache symmetrische Verfahren eingeführt werden: Caesar-Chiffre und Skytalen. Insbesondere der handlungsbasierte (**enaktive**) Zugang [Br64, S. 2] bietet sich für die Grundschule an, um an das entwicklungspsychologische Stadium der Schülerinnen und Schüler anzuknüpfen (siehe Jean Piagets *Modell der kognitiven Entwicklung* [Pi74]). Über die Gegenstände Kästchen, Schlüssel (privater Schlüssel) und Schloss (öffentlicher Schlüssel) kann ein kontrastierendes asynchrones Verfahren eingeführt werden.

Verschiedene kryptologische Verfahren, neben den genannten u. a. Vigenère-Chiffre, wurden bereits erfolgreich an Grundschulen umgesetzt [Ar16; MS17]. Die Notwendigkeit verschlüsselter Kommunikation ergibt sich für die Schülerinnen und Schüler intuitiv aus ihrer Lebenswelt. Sowohl das Verständnis für einfache Verfahren als auch deren Anwendung, die Einführung fachsprachlicher Elemente wie *Verschlüsseln*, *Übermitteln* oder *Entschlüsseln* sowie die Reflexion eingesetzter Verfahren ist in den Klassen 3-4 möglich und führt zu hoher Motivation der Schülerinnen und Schüler. Erfahrungen zur Thematisierung von Automatenmodellen in der Grundschule liegen bislang nur sporadisch vor. Es existieren verschiedene Kontexte, die spezifisch für den Primarbereich entwickelt wurden. Dazu zählen die Kontexte »Farbmischung« [Se17] oder »Schatzinsel« [BFW06]. Erfahrungen zur Vermittlung von Themen der Automatentheorie aus der ersten (Hochschule) und dritten (Fortbildungen) Lehrerbildungsphase liegen vor und deuten darauf hin, dass (angehende)

Lehrpersonen sich in entsprechend aufbereitete Kontexte einfinden können. Ansätze von Studierenden liegen in Form schriftlicher Unterrichtsentwürfe vor [Ar16].

4 Perspektiven

Die Arbeit muss fortgeführt werden, wollen wir nicht auf der Beispielebene verharren. Es gilt, Modelle zur Umsetzung der Empfehlungen in der Grundschule zu entwickeln, die nachhaltig informatische Bildung befördern. Dabei stellt sich die Frage nach der Verankerung der informatischen Bildung in der Grundschule in anderer Form als in den weiterführenden Schulen, da der Widerstand gegen die Einführung eines eigenständigen Schulfachs Informatik zur Zeit noch breit aufgestellt ist.

Es ist daher dringend auch zu diskutieren, ob Informatik als weitere Perspektive in den existierenden *Perspektivrahmen Sachunterricht* der GDSU für den Sachunterricht aufgenommen wird, ob die von den Schülerinnen und Schülern zu erreichenden Kompetenzen im Zusammenhang in die Schulfächer Deutsch und/oder Mathematik eingegliedert werden können oder weitere/andere Möglichkeiten zu entwickeln oder zu erproben sind, um informatische Bildung in der Grundschule für alle Schülerinnen und Schüler zu ermöglichen. Für jede der zu diskutierenden Varianten ist allerdings unabdingbar, dass die Lehrkräfte, die die Umsetzung realisieren, fachlich und fachdidaktisch auf diese Herausforderung bestens vorbereitet werden. Dazu müssen sie in der ersten Phase ihrer Lehrerbildung die fachlichen und fachdidaktischen Dimensionen der Informatik erfahren. Andere Varianten werden scheitern, wie am Beispiel der informationstechnischen Grundbildung abgelesen werden kann, die zwar seit 30 Jahren in den Bildungsplänen der weiterführenden Schulen vorgesehen ist, aber nicht stattfindet, wie z. B. in [Be19] ausgeführt wird.

Erste Informatikveranstaltungen für zukünftige Grundschullehrkräfte zeigen, dass der Aufwand und die Gestaltungsanforderungen durchaus eine Herausforderung für die Universitäten darstellen, wenn erreicht werden soll, dass nicht nur grundlegende Konzepte entwickelt werden sollen, sondern auch mögliche Optionen für die eigene Unterrichtsgestaltung als Ziel ausgewiesen wird, wie in »Informatik an Grundschulen – Stellschraube Lehrerbildung« [Ha19] deutlich wird. Die in einem anderen, parallelen Veranstaltungskontext entwickelten – universitären Ansprüchen genügenden – Kompetenzen verdeutlichen die mit der Informatik verbundenen Herausforderungen für die Lehrerbildung »Informatische Bildung für alle Lehramtsstudierenden – Reformprozess einer allgemeinbildenden Informatikveranstaltung in der universitären Lehrerbildung« [LH19]. Hoffnung ergibt sich, wenn wir betrachten, dass die Schülerinnen und Schüler, nachdem sie Informatikunterricht in der Grundschule erfahren haben, in der weiterführenden Schule eine höhere Selbstkompetenz in Informatik zum Ausdruck bringen, wie in »Interesse an Informatik und Informatikselbstkonzept bei Schülerinnen und Schülern zu Beginn der fünften Jahrgangsstufe des Gymnasiums – Eine empirische Untersuchung« [Mü19] deutlich wird.

Literatur

- [Ar16] Arbeitsbereich Didaktik der Informatik der WWU Münster, Hrsg.: Projekt Informatik in der Grundschule, Mai 2016, URL: <https://t1p.de/qhab>, Stand: 10. 05. 2019.
- [Be18] Berry, M.: »Computing« als neues Schulfach. Umsetzung des landesweiten Curriculums für das Fach Computing in England. LOG IN, Thema 38/189/190, hrsg. von Humbert, L.; Koerber, B., S. 20–26, 2018, ISSN: 0720-8642.
- [Be19] Bernewasser, J.: Informatik für alle, Digitalisierung an Schulen. Zeit online/, 9. Mai 2019, URL: <https://t1p.de/sp15>, Stand: 10. 05. 2019.
- [BFW06] Bell, T.; Fellows, M.; Witten, I. H.: Computer Science Unplugged, 2006, URL: <https://classic.csunplugged.org/>, Stand: 24. 03. 2019.
- [Br64] Bruner, J. S.: The course of cognitive growth. American Psychologist 19/1, S. 1–15, 1964.
- [Ga13] Gander, W.; Petit, A.; Berry, G.; Demo, B.; Vahrenhold, J.; McGettrick, A.; Boyle, R.; Drechsler, M.; Mendelson, A.; Stephenson, C.; Ghezzi, C.; Meyer, B.: Informatics education: Europe cannot afford to miss the boat, Apr. 2013, URL: <https://t1p.de/6jvf>, Stand: 23. 04. 2019.
- [GD13] GDSU, Hrsg.: Perspektivrahmen Sachunterricht. GDSU – Gesellschaft für Didaktik des Sachunterrichts, Julius Klinkhardt, Bad Heilbrunn, 2013, ISBN: 9783781519923.
- [GI00] Gesellschaft für Informatik e. V.: Empfehlung für ein Gesamtkonzept zur informatischen Bildung an allgemein bildenden Schulen. Informatik Spektrum 23/6, Beilage LOG IN 20 (2000) Heft 2, S. I-VII, S. 378–382, Dez. 2000, ISSN: 0170-6012, URL: <https://t1p.de/ntpf>, Stand: 19. 03. 2019.
- [GI08] Gesellschaft für Informatik e. V.: Grundsätze und Standards für die Informatik in der Schule – Bildungsstandards Informatik für die Sekundarstufe I, Erarbeitet vom Arbeitskreis »Bildungsstandards« – Beschluss des GI-Präsidiums vom 24. Januar 2008 – veröffentlicht als Beilage zur LOG IN 28 (2008) Heft 150/151, Apr. 2008, URL: <https://t1p.de/xb2w>, Stand: 30. 04. 2019.
- [GI16] Gesellschaft für Informatik e. V.: Grundsätze und Standards für die Informatik in der Schule – Bildungsstandards Informatik für die Sekundarstufe II, Erarbeitet vom Arbeitskreis »Bildungsstandards SII« – Beschluss des GI-Präsidiums vom 29. Januar 2016 – veröffentlicht als Beilage zur LOG IN 36 (2016) Heft 183/184, Apr. 2016, URL: <https://t1p.de/vxhy>, Stand: 30. 04. 2019.
- [GI19a] Gesellschaft für Informatik e. V.: GI-Radar 233: Informatik im Primarbereich, 2019, URL: <https://t1p.de/p129>, Stand: 22. 04. 2019.

- [GI19b] Gesellschaft für Informatik e. V., Hrsg.: Kompetenzen für informatische Bildung im Primarbereich, Empfehlungen der Gesellschaft für Informatik e. V. Erarbeitet vom Arbeitskreis »Bildungsstandards Primarbereich« – Beschluss des GI-Präsidiums vom 31. Januar 2019 – wird in gedruckter Form der LOG IN 39 (2019) Heft 191/192 beigelegt, 7. Feb. 2019, URL: <http://uni-w.de/1gm>, Stand: 09.03.2019.
- [Ha19] Haselmeier, K.: Informatik an Grundschulen – Stellschraube Lehrerbildung. In (Pasternak, A., Hrsg.): Informatik für alle, INFOS 2019, 18. GI-Fachtagung Informatik und Schule, 16.–18. September 2019, Dortmund, Germany. Bd. P288. LectureNotes in Informatics (LNI)-Proceedings, n. n. e., Gesellschaft für Informatik e. V. (GI), Bonn, xx-yy, 2019, ISBN: 978-3-88579-682-4.
- [HK18] Humbert, L.; Koerber, B., Hrsg.: Informatische Bildung im Primarbereich. Themenheft der Zeitschrift LOG IN, Bd. 38, 189/190, Aug. 2018.
- [Hu18] Humbert, L.; Müller, D.; Fricke, M.; Haselmeier, K.; Siebrecht, D.: »Because the music is not inside the piano«, Ist informatische Bildung ohne Informatiksysteme wünschenswert? LOG IN, Praxis & Methodik 38/189/190, hrsg. von Humbert, L.; Koerber, B., S. 67–72, 2018, ISSN: 0720-8642, URL: <http://uni-w.de/1aq>, Stand: 08.02.2019.
- [Hu19a] Humbert, L.; Best, A.; Micheuz, P.; Hellmig, L.: Informatik – Kompetenzentwicklung bei Kindern. Informatik Spektrum 42/, n. n. e. – Preprint, rs-tu, 2019, ISSN: 0170-6012, URL: <https://t1p.de/qhn7>, Stand: 10.05.2019.
- [Hu19b] Humbert, L.; Magenheimer, J.; Schroeder, U.; Fricke, M.; Bergner, N.: Handreichung für Lehrkräfte, Handreichungen und Unterrichtsmaterial. Hinweise zur Schulung/Fortbildung, hrsg. von Informatik an Grundschulen (IaG), 21. März 2019, URL: <https://t1p.de/iu9z>, Stand: 19.04.2019.
- [LH19] Losch, D.; Humbert, L.: Informatische Bildung für alle Lehramtsstudierenden – Reformprozess einer allgemeinbildenden Informatikveranstaltung in der universitären Lehrerbildung. In (Pasternak, A., Hrsg.): Informatik für alle, INFOS 2019, 18. GI-Fachtagung Informatik und Schule, 16.–18. September 2019, Dortmund, Germany. Bd. P288. LectureNotes in Informatics (LNI)-Proceedings, n. n. e.; Preprint-Version, die die eingereichte und angenommene Fassung darstellt und bis zur Veröffentlichung noch überarbeitet wird, Gesellschaft für Informatik e. V. (GI), Bonn, 1. Feb. 2019, ISBN: 978-3-88579-682-4, URL: <https://uni-w.de/11j>, Stand: 04.05.2019.
- [Ma18] Magenheimer, J.; Müller, K.; Schulte, C.; Bergner, N.; Haselmeier, K.; Humbert, L.; Müller, D.; Schroeder, U.: Evaluation of Learning Informatics in Primary Education. In (Pozdniakov, S. N.; Dagienė, V., Hrsg.): Informatics in Schools, Fundamentals of Computer Science and Software Engineering. 11th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2018, Springer Verlag, St. Petersburg, Russia, S. 339–354, Okt. 2018, ISBN: 978-3-030-02749-0.

- [MS17] MSB NRW, Hrsg.: Informatik an Grundschulen – Ziele, Pilotprojekt zur Erprobung von Konzepten zur informatischen Bildung im Rahmen des Sachunterrichts an Grundschulen, MSB NRW – Ministerium für Schule und Bildung des Landes Nordrhein-Westfalen, Juli 2017, URL: <https://t1p.de/pods>, Stand: 25. 01. 2019.
- [Mü19] Müller, D.; Haselmeier, K.; Humbert, L.; Killich, K.: Interesse an Informatik und Informatikselbstkonzept bei Schülerinnen und Schülern zu Beginn der fünften Jahrgangsstufe des Gymnasiums – Eine empirische Untersuchung. In (Pasternak, A., Hrsg.): Informatik für alle, INFOS 2019, 18. GI-Fachtagung Informatik und Schule, 16.–18. September 2019, Dortmund, Germany. Bd. P288. LectureNotes in Informatics (LNI)-Proceedings, n. n. e.; Preprint-Version, die die eingereichte und angenommene Fassung darstellt und bis zur Veröffentlichung noch überarbeitet wird, Gesellschaft für Informatik e. V. (GI), Bonn, xx–yy, 2019, ISBN: 978-3-88579-682-4, URL: <http://uni-w.de/11t>, Stand: 04. 05. 2019.
- [Pa] Pasternak, A., Hrsg.: Informatik für alle, INFOS 2019, 18. GI-Fachtagung Informatik und Schule, 16.–18. September 2019, Dortmund, Germany, Bd. P288, LectureNotes in Informatics (LNI)-Proceedings, n. n. e., Bonn: Gesellschaft für Informatik e. V. (GI), ISBN: 978-3-88579-682-4.
- [Pi74] Piaget, J.: Der Aufbau der Wirklichkeit beim Kinde. Klett, Stuttgart, 1974.
- [Se17] Sellin, L.: Ausgewählte Elemente der theoretischen Informatik als Element der informatischen Bildung im Primarbereich, PDF: <https://uni-w.de/161>, Bachelorthesis, Wuppertal: Fachgebiet Didaktik der Informatik – Bergische Universität, 29. März 2017, URL: <https://uni-w.de/130>, Stand: 10. 05. 2019.

Einsatz von Jupyter Notebooks am Beispiel eines fiktiven Kriminalfalls

Florian Kapp¹, Carsten Schulte²

Abstract: In diesem Praxisbeitrag wird anhand eines Unterrichtsvorhabens zum Thema geographische Daten der Einsatz von Jupyter Notebooks beschrieben und reflektiert. Schülerinnen und Schüler lösen einen fiktiven Kriminalfall, indem sie geographische Daten auf einer Karte visualisieren, diese anschließend analysieren und abschließend Aspekte des Datenschutzes in Bezug auf geographische Daten diskutieren. Das Unterrichtsvorhaben verfolgt das Ziel, die Schülerinnen und Schüler für das Thema „Umgang mit geographischen Daten“ zu sensibilisieren. Anhand des Unterrichtsvorhabens möchten wir Besonderheiten und didaktische Möglichkeiten der Jupyter Notebooks erläutern und darlegen, dass Jupyter Notebooks eine geeignete Programmierumgebung für den Informatikunterricht in der Schule darstellen.

Keywords: Interaktion und Exploration; Kompetenzen; Explorieren

1 Einleitung

In der heutigen Zeit existieren neben zahlreichen unterschiedlichen Programmiersprachen auch diverse Programmierumgebungen. Doch welche sind für den Unterricht geeignet? Die Gesellschaft für Informatik (GI) formuliert dazu folgendes: „Die Wahl der Programmierumgebung bzw. der Programmiersprache orientiert sich dann an Kriterien wie Anschaulichkeit, einfachem Einstieg und vielfältigen Einsatzmöglichkeiten.“ [Pea08, S. 33]

Neben Anschaulichkeit und einem einfachen Einstieg sind also vielfältige Einsatzmöglichkeiten die Kriterien, die eine Programmierumgebung erfüllen sollte [Pea08]. In diesem Praxisbeitrag möchten wir anhand eines Beispiels darlegen, dass Jupyter Notebooks eben diese Kriterien erfüllen und somit eine geeignete Programmierumgebung für die Schule darstellen.

Im ersten Schritt werden wir erläutern, wie Jupyter Notebooks aufgebaut sind und wie sie als Entwicklungsumgebung installiert werden können. Anschließend beschreiben wir ein Unterrichtsvorhaben zum Thema geographische Daten, gehen auf unsere Erfahrungen bei der Erprobung des Unterrichtsvorhabens ein und beschreiben die Einsatzmöglichkeiten

¹ Universität Paderborn, Didaktik der Informatik, Fürstenallee 11, 33102 Paderborn, Deutschland FlorianKapp@uni-paderborn.de

² Universität Paderborn, Didaktik der Informatik, Fürstenallee 11, 33102 Paderborn, Deutschland carsten.schulte@uni-paderborn.de

der Jupyter Notebooks. Abschließend ziehen wir ein Resümee und reflektieren, inwiefern Jupyter Notebooks die oben erwähnten Kriterien für eine geeignete Programmierumgebung erfüllen.

2 Die Entwicklungsumgebung Jupyter Notebook

Jupyter Notebooks wurden ursprünglich entwickelt, um den Arbeitsablauf des wissenschaftlichen Rechnens und der Analyse von Daten zu unterstützen [K116]. Das Projekt Jupyter entstand aus dem IPython-Projekt [K116] und versucht, Jupyter Notebooks einem größeren Anwenderkreis zugänglich zu machen [K116]. Jupyter verfolgt das Ziel, Open-Source-Tools für interaktives wissenschaftliches Rechnen und Datenwissenschaft in Forschung, Bildung und Industrie zu entwickeln, wobei der Schwerpunkt auf Benutzerfreundlichkeit, Kollaboration und Reproduzierbarkeit liegt [PG15].

Aufbau eines Jupyter Notebooks

Ein Jupyter Notebook besteht aus Zellen, die individuell modifiziert und ausgeführt werden können [K116]. Dabei wird zwischen Markdown-Zellen und Code-Zellen unterschieden. Abbildung 1 zeigt den Aufbau eines Jupyter Notebooks.

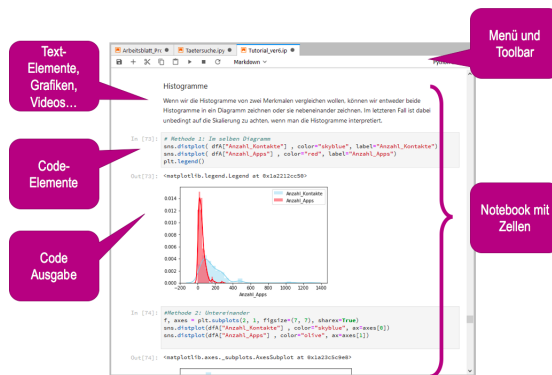


Abb. 1: Aufbau eines Jupyter Notebooks

Markdown-Zellen können unter anderem Text-Elemente, Grafiken und Videos enthalten und eignen sich besonders gut für erklärende Texte und Aufgabenstellungen. Code-Zellen hingegen beinhalten den Programmcode, wobei die Ausgabe jeder Code-Zelle direkt darunter erscheint und als Teil des Dokuments gespeichert wird [K116]. Die Ausgabe kann dabei neben Text auch umfangreiche Elemente wie Diagramme, mathematische Gleichungen oder interaktive Elemente und Grafiken enthalten [K116].

Installation von Jupyter Notebooks

Der Zugriff auf Jupyter Notebooks funktioniert über einen Webbrowser. [K116] Sie können lokal über eine Desktop Anwendung (z.B. Anaconda, weitere Hinweise zur Installation und Verwendung in [Ro18]) oder auf einem Remote Server ausgeführt werden [K116].

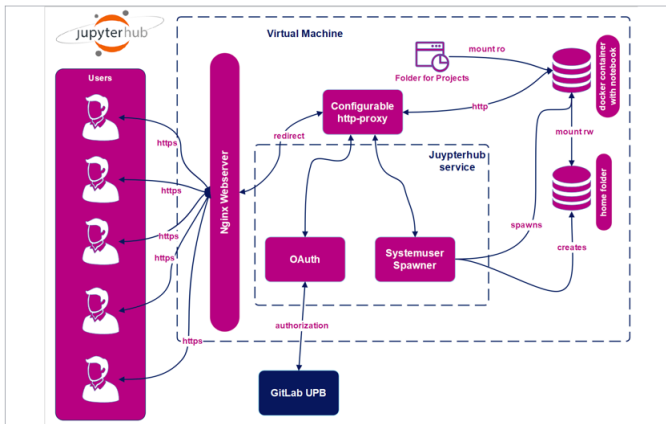


Abb. 2: Grafische Darstellung der Serverarchitektur

In Abbildung 2 ist die Serverarchitektur grafisch dargestellt, welche wir bei der Erprobung des Unterrichtsvorhabens genutzt haben. Den Schülerinnen und Schülern musste nur noch Zugriff auf unsere GitLab Instanz gewährt werden, welche die Autorisierung verwaltet. Anschließend konnten sie die Jupyter Notebooks in einem Webbrowser bearbeiten.

3 Unterrichtsvorhaben - Dem Täter auf den Fersen

Das Unterrichtsvorhaben beschreibt eine Möglichkeit, Jupyter Notebooks gewinnbringend im Informatikunterricht einzusetzen. Die Schülerinnen und Schüler lösen einen fiktiven Kriminalfall, indem sie programmieren, analysieren und mit den Jupyter Notebooks interagieren. Anhand von diesem Anwendungsfall diskutieren sie anschließend Aspekte des Datenschutzes in Zusammenhang mit geographischen Daten.

Zielsetzung

Das Ziel des Unterrichtsvorhabens ist es, Schülerinnen und Schüler für das Thema „Umgang mit geographischen Daten“ zu sensibilisieren. Sie sollen verstehen, dass es wichtig ist, auf seine eigenen Daten zu achten und ziehen eventuelle Rückschlüsse für ihren eigenen Umgang mit Daten. Zur Erreichung des Ziels nutzen die Schülerinnen und Schüler Jupyter Notebooks zur Auswertung und Analyse von geographischen Daten und lernen an einem Beispiel, dass ein verantwortungsbewusster Umgang mit Daten notwendig ist.

Voraussetzungen und Zielgruppe

Die Schülerinnen und Schüler arbeiten in dem Unterrichtsvorhaben mit der Programmiersprache Python. Diese bietet viele Möglichkeiten zur Datenauswertung und ist deshalb gut geeignet. Die Schülerinnen und Schüler sollten bereits Kenntnisse im Bereich des Programmierens in einer textuellen Programmiersprache besitzen und vor allem die Konzepte Schleifen, Objekte und Klassen kennen.

Anknüpfend an die Voraussetzungen richtet sich das Unterrichtsvorhaben vorrangig an Schülerinnen und Schüler der Sekundarstufe II. Falls Schülerinnen und Schüler der Sekundarstufe I die erforderlichen Vorkenntnisse besitzen, so kann das Unterrichtsvorhaben ebenfalls mit ihnen durchgeführt werden. Alternativ kann das Unterrichtsvorhaben mit Schülerinnen und Schülern der Sekundarstufe I ohne die Programmierphase durchgeführt werden, wenn ihnen das Ergebnis der Programmierphase - eine interaktive Karte (vergleiche Abbildung 3) - zusätzlich zur Verfügung gestellt wird.

Einordnung des Unterrichtsvorhabens

Zur Einordnung des Unterrichtsvorhabens orientieren wir uns an den Bildungsstandards der GI (vergleiche [Rea16]). In den Bildungsstandards sind verschiedene Inhalts- und Prozessbereiche formuliert. Das Unterrichtsvorhaben ist hauptsächlich in den Inhaltsbereich „Informatik, Mensch und Gesellschaft“ einzuordnen. Darüber hinaus kann das Unterrichtsvorhaben aufgrund der Programmierphase anteilig den Inhaltsbereich „Algorithmen“ und den Prozessbereich „Modellieren und Implementieren“ abdecken. Durch die didaktische Ausgestaltung des Unterrichtsvorhabens lässt sich das Unterrichtsvorhaben zudem in die Prozessbereiche „Kommunizieren und Kooperieren“, „Darstellen und Interpretieren“ und „Begründen und Bewerten“ einordnen.

Kurzbeschreibung des Unterrichtsvorhabens

Das Unterrichtsvorhaben besteht aus vier Unterrichtssequenzen, deren Durchführungsdauer in der Praxis variieren kann. Dies ist von den Vorkenntnissen der Schülerinnen und Schüler abhängig.

Die erste Unterrichtssequenz dient als thematischer Einstieg. Die Schülerinnen und Schüler sammeln zum Thema geographische Daten Stichpunkte auf Karteikarten, erläutern diese und gruppieren sie an der Tafel. Anschließend werden Oberbegriffe für die Gruppen von Kategorien erarbeitet. Daraufhin erklärt der Lehrende den Aufbau von geographischen Daten und zeigt den Schülerinnen und Schülern ein Video, welches eine Einführung in den Kriminalfall darstellt. Abschließend wird den Schülerinnen und Schülern das weitere Vorgehen erläutert.

Amanda Weber war Filialeiterin eines Lidl's und wurde am Hauptbahnhof ermordet. Die Polizei hat die geographischen Daten des Tattages der Verdächtigen gesammelt und möchte

diese nun untersuchen, um den Täter oder die Täterin zu überführen. Leider haben sie bei der Übermittlung der Daten einen Fehler gemacht und die Daten sind nicht mehr zuzuordnen.

In der zweiten Unterrichtssequenz geht es um die Auswertung der geographischen Daten. Die Schülerinnen und Schüler bearbeiten ein Jupyter Notebook mit verschiedenen Programmier- und Analyseaufgaben. Das Ziel der Aufgaben ist es, die geographischen Daten den Verdächtigen zuzuordnen. Zunächst nutzen die Schülerinnen und Schüler von uns zuvor programmierte Klassen, um damit eigene Karten zu programmieren. Diese werden zur Visualisierung der geographischen Daten verwendet. Im nächsten Schritt werden diese Karten analysiert, um anhand von Steckbriefen die geographischen Daten den Verdächtigen zuzuordnen.

Die dritte Unterrichtssequenz beschäftigt sich mit der Tätersuche. In der ersten Phase bearbeiten die Schülerinnen und Schüler ein Jupyter Notebook mit dem Ziel den Täter oder die Täterin zu überführen. Dazu erhalten sie Hinweise in Form von Informationstexten (vergleiche Abbildung 5) und müssen diese mit Hilfe einer Karte (vergleiche Abbildung 3) bearbeiten.

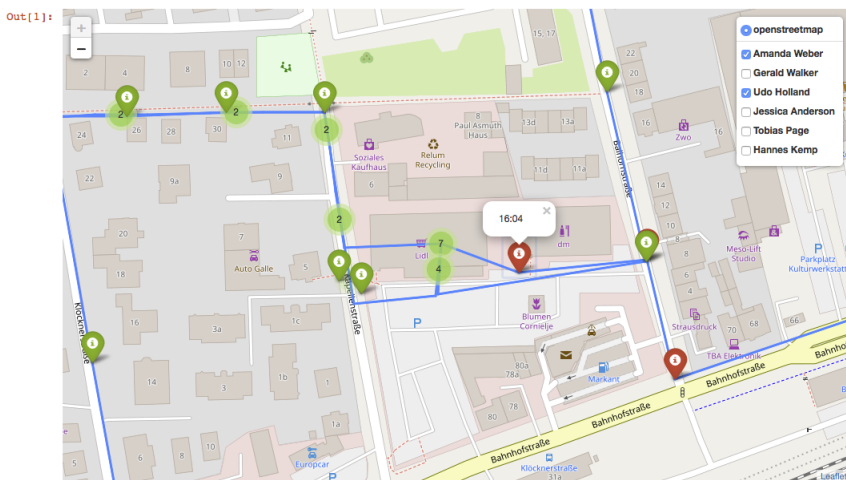


Abb. 3: Interaktive Karte zur Analyse der geographischen Daten

In der zweiten Phase sollen die Schülerinnen und Schüler in Gruppenarbeit auf einem Plakat begründet darstellen, wer der Täter oder die Täterin ist. Dabei ist zu beachten, dass auf Basis der geographischen Daten zwei mögliche Täter in Frage kommen, damit die Schülerinnen und Schüler erkennen, dass Daten nicht eindeutig sind.

In der letzten Unterrichtssequenz diskutieren die Schülerinnen und Schüler auf Basis der bisherigen Ergebnisse Aspekte des Datenschutzes in Bezug auf geographische Daten. Dabei werden, ausgehend von der Frage, wo geographische Daten gesammelt werden,

folgende Aspekte angesprochen: a) Chancen, Risiken und Missbrauchsmöglichkeiten, b) gesellschaftliche Folgen, c) Rückschlüsse für das eigene Verhalten.

Erfahrungsbericht

Im Rahmen eines Seminars wurde das Unterrichtsvorhaben mit Kursen verschiedener Jahrgangsstufen der Sekundarstufe II erprobt und reflektiert. Dabei konnten wir einige Erfahrungen sammeln, die wir im Folgenden darstellen möchten.

Die meisten Schülerinnen und Schüler haben während der gesamten Durchführung sehr konzentriert und ruhig gearbeitet. Gerade bei der Bearbeitung der Jupyter Notebooks waren sie sehr motiviert. Trotz begrenzter Zeit waren alle Lernenden an der Weiterarbeit am Jupyter Notebook interessiert.

Die Phase der Programmierung hat häufig sehr lange gedauert und den Schülerinnen und Schülern bei der Durchführung die meisten Probleme bereitet. Dies lag zum einen daran, dass die Schülerinnen und Schüler weniger Vorkenntnisse hatten als erwartet und zum anderen an der geforderten selbstständigen Arbeit.

Die Phase der Tätersuche hat allen Schülerinnen und Schülern viel Spaß gemacht. Während der Bearbeitung des Jupyter Notebooks konnte man ausgiebige Diskussionen beobachten, die bei der Präsentation der Ergebnisse neu entfacht sind. Allerdings muss man als Lehrender ebenfalls den Fall bedenken, dass alle Gruppen den gleichen Täter vermuten, sodass man selbst die entgegengesetzte Meinung einnehmen muss, damit eine Diskussion entsteht.

Die abschließende Diskussion verlief in den meisten Fällen sehr schleppend. Dies lag daran, dass die Schülerinnen und Schüler zwar eingesehen haben, dass der Umgang mit den eigenen Daten wichtig ist. Sie waren allerdings auch häufig der Meinung, dass sie nichts zu verstecken haben und Unternehmen ihre Daten sowieso bekommen würden, wenn sie die Daten haben wollen.

Den Rückmeldungen der Schülerinnen und Schüler konnten wir entnehmen, dass ihnen die Arbeit mit den Jupyter Notebooks Spaß gemacht hat und sie gerne häufiger damit arbeiten würden. Nach einer kurzen Eingewöhnungsphase haben sich die Schülerinnen und Schüler gut mit den Jupyter Notebooks zurecht gefunden. Gerade die interaktiven Elemente, welche sie nutzen konnten, waren sehr beliebt und wurden häufig genutzt. Bei der Arbeit mit dem Server sind allerdings zunächst Probleme aufgetreten. Es ist daher darauf zu achten, wie viele Ressourcen die Arbeit in den Jupyter Notebooks benötigt, um den Server nicht zu überlasten.

Die Programmiersprache Python war für die Schülerinnen und Schüler zunächst sehr ungewohnt. Jedoch nach einigen Tipps und Hinweisen zum Einstieg wurden die anfänglichen Probleme schnell überwunden, sodass sich die Schülerinnen und Schüler anschließend recht gut damit zurechtfinden und hinterher zurückmeldeten, dass Python „besser“ als Java sei, da die Syntax weniger komplex ist.

4 Einsatzmöglichkeiten von Jupyter Notebooks

Jupyter Notebooks besitzen eine Vielzahl von Einsatzmöglichkeiten im Informatikunterricht. Im Folgenden möchten wir vorstellen, welche der Möglichkeiten wir in dem beschriebenen Unterrichtsvorhaben genutzt haben und wie diese umgesetzt sind.

Entwicklungsumgebung

Wie bereits beschrieben sind Jupyter Notebooks so aufgebaut, dass die Schülerinnen und Schüler ihren Programmcode in Code-Zellen schreiben, diesen ausführen und direkt darunter Rückmeldung erhalten. Der entscheidende Vorteil von Jupyter Notebooks ist, dass ein komplexes Programm durch den zellenweisen Aufbau leicht in kleinere Teilprobleme aufgeteilt werden kann. Für die Schülerinnen und Schüler ist es dann möglich diese Teilprobleme schrittweise zu lösen, um ein in sich geschlossenes Programm zu erhalten. Diese Herangehensweise fördert die Interaktion der Schülerinnen und Schüler mit der Entwicklungsumgebung.

Arbeitsblatt

Jupyter Notebooks können ebenfalls als Arbeitsblatt genutzt werden. Dazu werden Arbeitsanweisungen und erklärende Texte einfach in eine Markdown-Zelle geschrieben und ausgeführt. In Abbildung 4 ist eine Aufgabe aus dem Jupyter Notebook zur Datenauswertung dargestellt.

Tagesablauf auf einer Karte einzeichnen

Mit Hilfe der Heatmap konntest du dir bereits einen sehr guten Überblick über den Tag verschaffen. Allerdings ist eine Heatmap nicht sehr detailliert und man kann auch nicht erkennen, wann eine Person einen bestimmten Ort besucht hat. Aus diesem Grund geht es nun darum, die Tagesabläufe der Personen auf der Karte einzuzichnen, damit detaillierte Informationen zugrunde liegen. Dazu wird ebenfalls die Klasse `Karte` verwendet. Die Methoden der Klasse könnt ihr in der Dokumentation unter Aufgabe 2 nachlesen.

Aufgabe 4

1. Erstelle ein Objekt der Klasse `Karte` mit geeignetem Mittelpunkt.
2. Zeichne alle Standorte des Tattages auf der Karte ein.
3. Verbinde alle Standorte miteinander.
4. Lass dir die Karte anzeigen.

Hinweis 1 ein/ausblenden

Hinweis 2 ein/ausblenden

Hinweis 3 ein/ausblenden

Hinweis 1 - Experimente

Experimentiere zunächst mit der Karte, indem du verschiedene Standorte der Standortdaten einzeichnest und miteinander verbindest. Du kannst dabei folgendermaßen vorgehen:

1. Erstelle ein Objekt der Klasse `Karte` mit beliebigem Mittelpunkt.
2. Zeichne beliebige Standorte auf der Karte ein (z.B. die Standorte an den Stellen 1, 5, 7, 9)
3. Verbinde nun die Standorte miteinander, sodass ein Viereck entsteht.
4. Lass dir die Karte anzeigen.

1 ▾ #Dein Code für Aufgabe 4

Abb. 4: Beispielaufgabe aus dem Jupyter Notebook zur Datenauswertung

Wie auf einem herkömmlichen Arbeitsblatt gibt es zunächst einen erklärenden Text, gefolgt von einer konkreten Arbeitsanweisung. Jupyter Notebooks bieten die Möglichkeit, im Anschluss an eine Arbeitsanweisung eine Code-Zelle oder eine Markdown-Zelle zu erstellen, um die Ergebnisse direkt unter der Aufgabe festzuhalten. Jupyter Notebooks bieten

so die Möglichkeit, den Programmcode auszuführen und die Aufgaben experimentell in kleineren Aufgaben zu bearbeiten.

Kontrollmethode

Als Lehrender kennt man die Situation, wenn ein Schüler oder eine Schülerin in einer Arbeitsphase nur aufzeigt, um zu fragen, ob die Lösung richtig ist. Gerade in Programmierphasen ist dies schwierig zu beantworten und mit einem erhöhten Aufwand verbunden, da es nicht die eine richtige Lösung gibt. Jupyter Notebooks können hier Abhilfe schaffen und bieten die Möglichkeit, dass Schülerinnen und Schüler selbst kontrollieren können, ob ihre Lösung richtig ist. Im Gegensatz zu einem Compiler kann ebenfalls die Korrektheit der Lösung überprüft werden.

Hinweis 2 - Verbindungen zum Opfer

Nachdem **Tobias Page** als Täter nicht in Frage kommt wurden die verbleibenden Verdächtigen ins Polizeirevier eingeladen, um eine Zeugenaussage zu machen. Die Verdächtigen wurden darauf angesprochen, dass sie zum Tatzeitpunkt in der Nähe des Hauptbahnhofs waren und dass sie die Möglichkeit hatten, die Tatwaffe im **Lidl** in der Nähe des Hauptbahnhofs zu erwerben. Hier sind die Aussagen der Verdächtigen:

Hannes Kemp:

Es ist nicht ungewöhnlich, dass ich in der Nähe des Hauptbahnhofs bin. Ich bin sehr häufig dort, um Passanten nach Geld zu fragen oder mir in der Schenke im Hauptbahnhof ein paar Bierchen zu trinken. Es stimmt zwar, dass ich im Lidl war, allerdings frage ich mich, was ich mit einem Messer soll? Ich war im Lidl, um Pfand wegzubringen und Bierdosen zu kaufen.

Udo Holland:

Ich war kurz am Hauptbahnhof, um mir bei Mc Donalds etwas zu essen zu holen. Ich wusste nicht, wieso das verwerflich ist. Ich wohne in der Nähe des Lidl's und gehe dort regelmäßig einkaufen. Ich denke, dass ich da nicht der einzige bin, es könnten also tausende Menschen dieses Messer dort gekauft haben.

Jessica Anderson:

Da ich Kundin bei der Sparda-Bank bin, halte ich mich häufig in der Nähe des Hauptbahnhofs auf. Ich arbeite im Lidl in der Nähe des Hauptbahnhofs, aber ich habe in dieser Woche kein neues Messer gekauft. Glauben Sie mir, ich weiß, woher die Messer kommen und welche Qualität sie haben.

Gerald Walker:

Als Vertreter verschiedener Immobilienbüros verbringe ich eine große Zeit an Bahnhöfen, weshalb ich auch an diesem Abend am Hauptbahnhof in Paderborn war. Ich war etwas in Eile, da ich den Zug beinahe verpasst hätte. Ich habe morgens im Lidl mein Mittagessen für den Tag gekauft, da ich wusste, dass ich ein längeres Meeting habe.

Zusammenfassung der Zeugenaussagen

Die verbleibenden Verdächtigen haben erklärt, warum sie im Lidl und in der Nähe des Hauptbahnhofs waren. Als Ermittler stellt sich die Frage, ob man ihnen glauben kann. Um einen Mordfall aufzuklären, muss man auch die Verbindungen zwischen dem Opfer und den Verdächtigen untersuchen. Schaut euch die Karte nochmal an und untersucht dabei folgende Aspekte:

- Sind die Zeugenaussagen plausibel?
- Gab es Verbindungen zwischen den Verdächtigen und dem Opfer?

Schlussfolgerungen Hinweis 2

Der **zweite Hinweis** hat dafür gesorgt, dass eine weitere Person aus dem Verdächtigenkreis gestrichen werden kann. Kurz nachdem der **zweite Hinweis** bearbeitet wurde, hat die Polizei eine sehr interessante Entdeckung gemacht. Im **dritten Hinweis** geht es um diese Entdeckung. Um den **dritten Hinweis** zu erhalten, müsst ihr erneut den Namen der Person eintragen, die durch den **zweiten Hinweis** ausgeschlossen werden konnte.

Hinweis 3 anzeigen	Trage den Namen der Person ein!
--------------------	---------------------------------

Abb. 5: Ausschnitt aus dem Jupyter Notebook zur Tätersuche

In Abbildung 5 ist ein Ausschnitt aus dem Jupyter Notebook zur Tätersuche dargestellt. Am unteren Rand kann man einen Button „Hinweis 3 anzeigen“ und ein Textfeld erkennen. Die Schülerinnen und Schüler müssen in das Textfeld den Namen des Verdächtigen eintragen, der ausgeschlossen werden kann, damit sie den nächsten Hinweis erhalten. Dieser wird anschließend in der gleichen Form wie Hinweis 2 (vergleiche Abbildung 5 oben) eingeblendet.

Button und Textfeld sind von uns programmiert worden und der dazugehörige Programmcode wurde mit einer Erweiterung³ für das Jupyter Notebook versteckt, damit die Schülerinnen und Schüler den Namen nicht dem Programmcode entnehmen können. Beim Klick auf den Button wird zunächst abgefragt, ob der vorherige Hinweis bearbeitet wurde und ob der

³ vergleiche https://github.com/kirbs-/hide_code

eingetragene Name richtig ist. Ist dies der Fall, wird eine Textdatei ausgegeben, welche den nächsten Hinweis enthält. Ansonsten wird eine Fehlermeldung ausgegeben. Dieses Vorgehen ermöglicht es, dass die Schülerinnen und Schüler die Hinweise erst bekommen, wenn es an der Zeit ist. Ansonsten könnten sie direkt den letzten Hinweis durchlesen und wären fertig.

Hilfe zur Selbsthilfe

In der heutigen Gesellschaft spielt Selbstständigkeit eine große Rolle. Schülerinnen und Schüler müssen demnach auch in der Lage sein, sich selbst zu helfen und selbstständig zu arbeiten. Die Nutzung der Jupyter Notebooks erlaubt es dem Lehrenden, seine Schülerinnen und Schüler dabei zu unterstützen, selbstständiger zu werden.

Die eben angesprochene Methode zur Selbstkontrolle ist ein Beispiel, allerdings bieten Jupyter Notebooks noch mehr Möglichkeiten. In Abbildung 4 kann man neben den bereits angesprochenen erklärenden Texten, den Arbeitsanweisungen und dem Programmcode noch drei Buttons erkennen. Diese Buttons ermöglichen es, dass sich die Schülerinnen und Schüler drei gestufte Hilfestellungen ein- und ausblenden können. Sie können die Hilfestellungen selbstständig durchlesen und bearbeiten und werden so immer selbstständiger. Der Vorteil, den die Jupyter Notebooks hierbei bieten, ist, dass Aufgaben und Hilfestellungen in einer Umgebung sind und so keine Medienbrüche entstehen.

Die Umsetzung der Buttons funktioniert im Grunde wie die Umsetzung der Kontrollmethode. Beim Klick auf einen der Buttons wird eine entsprechende Textdatei als Ausgabe eingeblendet. Buttons und Button Events sind von uns programmiert und der Programmcode wurde mit einer Erweiterung versteckt. Dies wurde ergänzt durch Abfragen, die überprüfen, ob eine Hilfestellung angezeigt wird, damit nicht zwei Hilfestellungen gleichzeitig eingeblendet werden.

5 Fazit & Ausblick

Am Anfang des Praxisbeitrags haben wir uns die Frage gestellt, ob Jupyter Notebooks eine geeignete Programmierumgebung für den Einsatz in der Schule sind. Laut GI müssen Programmierumgebungen Kriterien wie Anschaulichkeit, einfachem Einstieg und vielfältigen Einsatzmöglichkeiten erfüllen [Pea08].

Nach einer kurzen Eingewöhnungsphase sind die Schülerinnen und Schüler sehr gut mit den Jupyter Notebooks zurecht gekommen. Außerdem ist es sehr hilfreich, dass für die Nutzung der Jupyter Notebooks lediglich ein Webbrowser nötig ist, falls die entsprechende Serverfunktionalität gegeben ist. Die Schülerinnen und Schüler müssen sich demnach auch zuhause keine zusätzliche Software installieren.

Durch die Möglichkeit Bilder, Videos oder interaktive Elemente in ein Jupyter Notebook zu integrieren ist ebenfalls das Kriterium der Anschaulichkeit erfüllt. Mit diesen Möglichkeiten könnten Schülerinnen und Schüler auch ihre eigene Benutzeroberfläche erstellen.

Weiter haben wir am Beispiel des beschriebenen Unterrichtsvorhabens viele verschiedene Einsatzmöglichkeiten von Jupyter Notebooks aufgezeigt. Gerade die sequenzielle Ausführung der Code-Zellen, die Minimierung der Medienbrüche, der Zugriff über einen Webbrowser und die interaktive Auseinandersetzung mit der Entwicklungsumgebung sind wesentliche Vorteile der Jupyter Notebooks.

Mit den in diesem Praxisbeitrag beschriebenen Einsatzmöglichkeiten sind diese noch nicht erschöpft. Jupyter Notebooks könnten ebenfalls zur Differenzierung genutzt werden, indem Buttons Aufgaben mit unterschiedlichem Schwierigkeitsgrad einblenden. Außerdem gibt es einige Erweiterungen für Jupyter Notebooks, die z.B. die Einbindung von LaTeX oder eine Nutzung zur Präsentation ermöglichen.

Bei allen Vorteilen und Einsatzmöglichkeiten besitzen Jupyter Notebooks auch einen wesentlichen Nachteil. Die Einrichtung des Servers ist eine große Herausforderung und hat uns viele Probleme bereitet. Mit der oben dargestellten Struktur kann dieses Problem allerdings gelöst werden.

Zusammenfassend kann man also sagen, dass Jupyter Notebooks die geforderten Kriterien in vollem Maße erfüllen und für den Einsatz in der Schule geeignet sind. Sobald die Einrichtung des Servers abgeschlossen ist, erhalten Lehrerinnen und Lehrer ein Werkzeug, welches das Lernen der Schülerinnen und Schüler sehr gut unterstützen kann.

Literaturverzeichnis

- [K116] Kluver, Thomas; Ragan-Kelley, Benjamin; Pérez, Fernando; Granger, Brian E.; Bussonnier, Matthias; Frederic, Jonathan; Kelley, Kyle; Hamrick, Jessica B.; Grout, Jason; Corlay, Sylvain; Ivanov, Paul; Avila, Damián; Abdalla, Safia; Willing, Carol; et al.: Jupyter Notebooks - a publishing format for reproducible computational workflows. In: Positioning and Power in Academic Publishing: Players, Agents and Agendas, 20th International Conference on Electronic Publishing, Göttingen, Germany, June 7-9, 2016. S. 87–90, 2016.
- [Pea08] Puhlmann, Hermann; et al.: , Grundsätze und Standards für die Informatik in der Schule. Bildungsstandards Informatik für die Sekundarstufe I, 2008.
- [PG15] Pérez, Fernando; Granger, Brian E.: Project Jupyter: Computational narratives as the engine of collaborative data science. Retrieved September, 11, 2015.
- [Rea16] Röhner, Gerhard; et al.: , Bildungsstandards Informatik für die Sekundarstufe II, 2016.
- [Ro18] Rossant, C.: IPython Interactive Computing and Visualization Cookbook: Over 100 hands-on recipes to sharpen your skills in high-performance numerical computing and data science in the Jupyter Notebook, 2nd Edition. Packt Publishing, 2018.

Informatikunterricht in Baden-Württemberg

Entwicklung an allgemein-bildenden Gymnasien

Andreas Koch¹, Alexander Mittag²

Abstract: In den beiden vergangenen Jahren wurde der Informatikunterricht in der Sekundarstufe I an allgemein-bildenden Gymnasien in Baden-Württemberg durch Einführung des Pflichtfachs „Aufbaukurs Informatik“ und des Profulfachs „IMP“ verstärkt. Auch die Kursstufe wird in den kommenden Jahren weiterentwickelt werden.

Keywords: Aufbaukurs Informatik, Baden-Württemberg, Entwicklung, IMP, Informatik, Kursstufe

1 Einführung

Der Informatikunterricht ist in den beiden vergangenen Jahren in der Sekundarstufe I an allgemein-bildenden Gymnasien in Baden-Württemberg deutlich ausgebaut worden. Die Erweiterungen sind in Tab. 1 grau hinterlegt dargestellt.

Klassenstufe	7	8	9	10	KS1	KS2
Aufbaukurs Informatik	1					
Profulfach IMP		2	1	1		
AG				2		
Wahlfach 2-stündig					2	2
Wahlfach 4-stündig					4	4

Tab. 1: Wochenstundenverteilung im Fach Informatik seit dem Schuljahr 2018/19

¹ Seminar für Ausbildung und Fortbildung der Lehrkräfte Tübingen (Gymnasium), Mathildenstr. 32, 72072 Tübingen, Deutschland, andreas.koch@seminar-tuebingen.de

² Zentrum für Schulqualität und Lehrerbildung, Fasanenweg 11, 70771 Leinfelden-Echterdingen, Deutschland, alexander.mittag@zsl.kv.bwl.de

2 Informatikunterricht in Sekundarstufe I

2.1 Aufbaukurs Informatik

Seit dem Schuljahr 2017/18 findet erstmals verpflichtender, versetzungsrelevanter Informatikunterricht für alle Schülerinnen und Schüler an allgemein-bildenden Gymnasien in Baden-Württemberg statt. Das fächerintegrative Fach „Informationstechnische Grundbildung (ITG)“ gibt es nicht mehr. Die Kontingenzstundentafel weist nun eine Stunde für den „Aufbaukurs Informatik“ aus. Der Namensteil „Aufbaukurs“ suggeriert, dass die Schülerinnen und Schüler in den Klassenstufen 5 und 6 bereits Kompetenzen im Bereich der Informatik erworben haben. Allerdings findet in diesen beiden Klassenstufen kein Informatikunterricht statt. Der insgesamt 35 Schulstunden umfassende „Basiskurs Medienbildung“ in Klassenstufe 5 definiert im zugrundeliegenden Bildungsplan [Mi16-2] keine Kompetenzen, die dem Fach Informatik explizit zuzuordnen sind.

Der Bildungsplan des Aufbaukurses Informatik [Mi16-1] besitzt auch für Haupt-/Werkrealschulen, Realschulen und Gemeinschaftsschulen Gültigkeit. Die inhaltsbezogenen Kompetenzen gliedern sich in vier Bereiche, die in Tab. 2 dargestellt werden.

Bereich		Auswahl der Inhalte ³
1	Daten und Codierung	Alltägliche Codierungen (z.B. Morse-Code) Codierungen von Bild und Text Umrechnung von Binär- und Dezimalzahlen
2	Algorithmen	Programmierung mithilfe von Anweisungen, Bedingungen, Sequenzen, Schleifen, Variablen und Verzweigungen Visuelle Entwicklungsumgebung wie „Scratch“ wird empfohlen [Le16]
3	Rechner und Netze	Grundlegende Struktur lokaler Netze und des Internets Grundlegender Ablauf der Internetkommunikation Möglichkeiten der Datenspeicherung
4	Informationsgesellschaft und Datensicherheit	Einfache Verschlüsselungsverfahren (z.B. Caesar-Verfahren) Angriffe auf einfache Verschlüsselungsverfahren Umgang mit (digitalen) Daten an gegebenen Fallbeispielen

Tab. 2: Inhaltsbezogene Kompetenzbereiche des Bildungsplans des Profillfachs IMP in Informatik

Einige der Inhalte aus den Bereichen (1) und (4) wie „Codierungen von Bild und Text“ bzw. (2) und (3) wie „Programmierung“ und „Verschlüsselungsverfahren“ sind im vormaligen gültigen Bildungsplan für ITG [Mi04, S. 309-313] bzw. im aktuell noch gültigen Bildungsplan für die Kursstufe [Mi04, S. 437-443] enthalten und sind im Rahmen eines

³ Aus Gründen der besseren Lesbarkeit werden die Kompetenzformulierungen in Tabellen stets als Inhaltsstichpunkte wiedergegeben.

spiralcurricularen Unterrichts vorgezogen worden. Diese Bildungspläne befinden sich momentan auch deshalb in der Überarbeitung.

2.2 Profilfach IMP

Seit dem Schuljahr 2018/19 können zugelassene allgemein-bildende Gymnasien das Profilfach „Informatik Mathematik Physik (IMP)“ für die Klassenstufen 8 bis 10 anbieten. Aktuell bieten 56 Gymnasien IMP an [Mi18-4]. Der Unterricht findet in Informatik, Mathematik und Physik separat und gemäß der Stundenverteilung in Tab. 3 statt. Am Schuljahresende wird eine einzelne Note für IMP ausgewiesen [Mi18-3].

Klassenstufe	8	9	10
Informatik	2	1	1
Mathematik	1	1	2
Physik	1	2	1

Tab. 3: Wochenstundenverteilung im Profilfach IMP

Der Bildungsplan für IMP [Mi18-2] definiert für jedes der drei Fächer separate inhaltsbezogene Kompetenzen, wobei Verschränkungen bestehen, z.B. zwischen der Graphentheorie in Mathematik und Algorithmen in Informatik. Die inhaltsbezogenen Kompetenzen knüpfen an denen des Aufbaukurses Informatik bzw. denen des Vorjahresunterrichts in IMP an (vgl. Tab. 4), vertiefen und erweitern diese.

Bereich		Auswahl der Inhalte		
		Klassenstufe 8	Klassenstufe 9	Klassenstufe 10
1	Daten und Codierung	Fehlerkorrektur Graphen	Datenkompression Dijkstra-Algorithmus	
2	Algorithmen	Datenstrukturen Softwareprojekt	Algorithmen-Entwurf Unterprogramme	GUI-Erstellung Mehrdim. Arrays
3	Rechner und Netze	Adressierung Namensauflösung		Routing Schaltnetze
4	Informations-gesellschaft und Datensicherheit	Vigenère-Verfahren Umgang mit personen-bezogenen Daten	Backupstrategien Datensicherheit	Asymmetrische Verschlüsselung Zertifizierung

Tab. 4: Inhaltsbezogene Kompetenzbereiche des Bildungsplans des Aufbaukurses Informatik

In Bereich (2) ist ab Klassenstufe 9 die Verwendung einer textuellen Programmiersprache vorschrieben [Mi18-2, 3.2.1.2].

3 Informatikunterricht in der Kursstufe

In der Kursstufe werden derzeit ein zweistündiger Wahlkurs, der von allen Schülerinnen und Schülern gewählt werden kann, und ein vierstündiger Wahlkurs, dessen Wahlvoraussetzung die Teilnahme an der zweistündigen AG in Klassenstufe 10 ist, angeboten. Der vierstündige Wahlkurs findet im Rahmen eines Schulversuchs statt und ermöglicht seit dem Schuljahr 2016/17 eine schriftliche Abiturprüfung. Die Schulverwaltung macht hierzu folgende statistische Angaben: Aktuell nehmen 33 Gymnasien am Schulversuch teil. Im Jahr 2018 haben 77 Schülerinnen und Schüler die Prüfung abgelegt, 15 mehr als im Vorjahr.

Zum Schuljahr 2019/20 tritt eine Reform der Kursstufe in Kraft [Mi18-1], in deren Rahmen auch die Informatikangebote der gymnasialen Kursstufe überarbeitet werden. Mit Beginn des Schuljahres 2021/22 wird sich so in Kombination mit den bereits implementierten Informatikangeboten der Sekundarstufe I voraussichtlich die in Abb.1 dargestellte Gesamtstruktur des Informatikunterrichts in Baden-Württemberg ergeben. Allgemein-bildende Gymnasien können weiterhin ein zweistündiges Wahlfach anbieten, das allen Schülerinnen und Schülern offen steht. Zugelassene Gymnasien können darüber hinaus ein dreistündiges Basisfach und ein fünfständiges Leistungsfach anbieten, deren Wahlvoraussetzung die Belegung von IMP oder die Teilnahme an der AG ist. Die zweistündige AG kann zukünftig auch jeweils einstündig in den Klassenstufen 9 und 10 angeboten werden. Schülerinnen und Schüler, die das Wahlfach belegen, können sich nur dann für eine mündliche Abiturprüfung entscheiden, wenn sie vorab die AG oder IMP besucht haben.

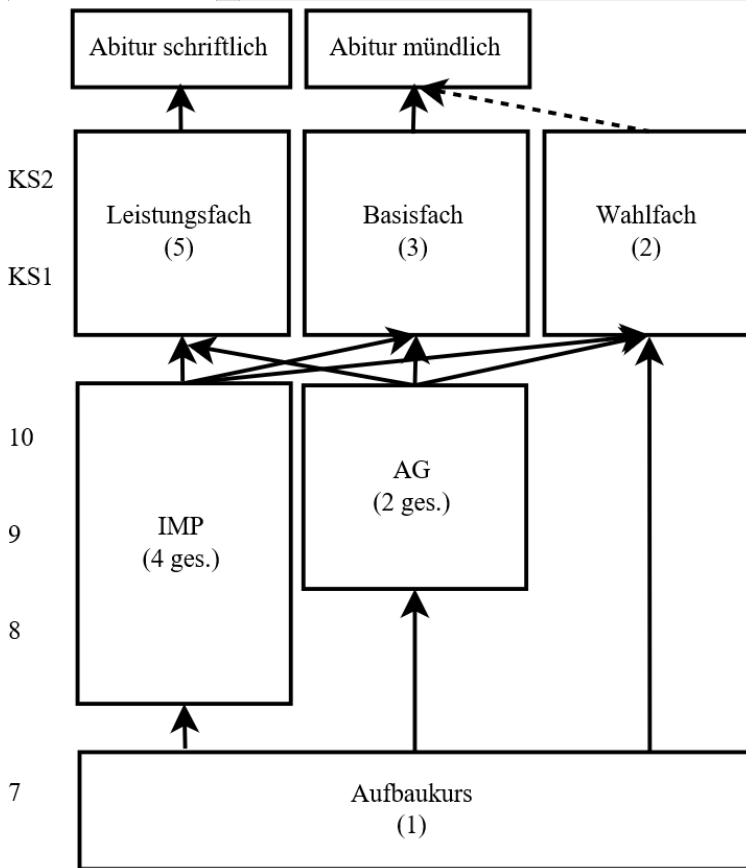


Abb. 1: Struktur des Informatikunterrichts in Baden-Württemberg mit der geplanten neuen Kursstufe ab dem Schuljahr 2021/22

Die Bildungspläne für die zukünftige AG und die drei neuen Fächer der Kursstufe befinden sich aktuell in der Entwicklung. Der momentan gültige Bildungsplan für die zweistündige Kursstufe [Mi04] enthält die Kompetenzbereiche „Information und Daten“, „Algorithmen und Daten“, „Problemlösen und Modellieren“, „Wirkprinzipien von Informatik-Systemen“ und „Informatik und Gesellschaft“. Der Bildungsplan für die vierstündige Kursstufe [Mi14] ist um den Bereich „Sprachen und Automaten“ ergänzt worden. Die Kompetenzen berücksichtigen die Bildungsstandards der Gesellschaft für Informatik [GI16] und umfassen u.a. die Themen „Datenbanksysteme“ und „Objektorientierte Modellierung und Programmierung“.

4 Ausblick

Die kommenden Jahre werden zeigen, ob sich IMP an den Schulen etablieren wird. Langfristig ist ein Übergang dieses Profulfachs in ein Pflichtfach wünschenswert, damit alle Schülerinnen und Schüler durchgängigen Informatikunterricht ab Klassenstufe 7 erhalten.

Das Vorziehen von inhaltsbezogenen Kompetenzen aus den aktuellen Bildungsplänen der Kursstufe in die der Sekundarstufe I bietet die Gelegenheit, bei der Überarbeitung der Kursstufenpläne Erweiterungen und Vertiefungen, z.B. in den Bereichen „Automaten und Sprachen“ und „Algorithmen“, vorzunehmen. Eine Ergänzung um einen neuen Teilkompetenzbereich „Projektmanagement“ wäre sinnvoll, um verschiedene inhalts- und prozessbezogene Kompetenzen der Schülerinnen und Schüler zusammenzuführen, zu erweitern und zu vertiefen.

Durch den erfolgten und geplanten Ausbau des Informatikunterrichts werden zusätzliche Fachlehrkräfte benötigt. Das Land Baden-Württemberg trägt diesem Ausbau mit Fortbildungsmaßnahmen wie dem „Kontaktstudium IMP“ und dem zweijährigen „Jahreskurs Informatik SII“ Rechnung und wird auf diese Weise auch in den kommenden Jahren weitere Lehrkräfte für den Informatikunterricht qualifizieren [Re19]. Abzuwarten bleibt, welche Rolle das neu gegründete Zentrum für Schulqualität und Lehrerbildung [Mi19] bei der Implementierung zukünftiger Fortbildungsmaßnahmen im Bereich der Informatik spielen wird.

Quellenverzeichnis

- [GI16] Gesellschaft für Informatik (GI) e.V.: Bildungsstandards Informatik für die Sekundarstufe II, 2016. https://www.informatikstandards.de/docs/Bildungsstandards_SII.pdf. Abruf am 11.10.2018.
- [Le16] Lehrerinnenfortbildung Baden-Württemberg: Programmieren mit Scratch. https://lehrerfortbildung-bw.de/u_matnatech/informatik/gym/bp2016/fb1/2_algorithmen/1_hintergrund/2_hintergrund/2_scratch/. Abruf am 11.10.2018.
- [Mi04] Ministerium für Kultus, Jugend und Sport Baden-Württemberg: Bildungsplan 2004. Allgemein-bildendes Gymnasium, 2004. http://www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lsw/Bildungsplaene/Bildungsplaene-2004/Bildungsstandards/Gymnasium_Bildungsplan_Gesamt.pdf, Abruf am 11.10.2018.
- [Mi14] Ministerium für Kultus, Jugend und Sport Baden-Württemberg: Bildungsplan Informatik Baden-Württemberg (vierstündige Kursstufe), 2014. http://www.bildungsplaene-bw.de/site/bildungsplan/get/documents/lsw/Bildungsplaene/Bildungsplaene-2004/Bildungsstandards/Schulversuch%20INF-4-std/Gym_Inf_wb_Schulversuch_bs.pdf. Abruf am 11.10.2018.
- [Mi16-1] Ministerium für Kultus, Jugend und Sport Baden-Württemberg: Bildungsplan 2016, Gemeinsamer Bildungsplan für die Sekundarstufe I, Aufbaukurs Informatik, 2016. <http://www.bildungsplaene-bw.de/Lde/LS/BP2016BW/ALLG/SEK1/INF7>. Abruf am 11.10.2018.

- [Mi16-2] Ministerium für Kultus, Jugend und Sport Baden-Württemberg: Bildungsplan 2016, Gemeinsamer Bildungsplan für die Sekundarstufe I, Basiskurs Medienbildung, 2016. <http://www.bildungsplaene-bw.de/,Lde/LS/BP2016BW/ALLG/SEK1/BMB>. Abruf am 11.10.2018.
- [Mi18-1] Ministerium für Kultus, Jugend und Sport Baden-Württemberg: Anpassung des Schulversuchserlasses „Informatik 4-stündig“ in den Jahrgangstufen an die neue AGVO ab dem Schuljahr 2019/2020, 01.10.2018. AZ 37-6521.-15-INF/100/1.
- [Mi18-2] Ministerium für Kultus, Jugend und Sport Baden-Württemberg: Bildungsplan 2016 Gymnasium, Endfassung, Profulfach Informatik Mathematik Physik (IMP), 2018. <http://www.bildungsplaene-bw.de/,Lde/LS/BP2016BW/ALLG/GYM/IMP>. Abruf am 11.10.2018.
- [Mi18-3] Ministerium für Kultus, Jugend und Sport Baden-Württemberg: Bildungsplanreform, Häufig gestellte Fragen hierauf, Einführung des Profulfachs Informatik, Mathematik, Physik (IMP). <http://www.km-bw.de/,Lde/Startseite/Schule/Haeufig+gestellte+Fragen>, Abruf am 11.10.2018.
- [Mi18-4] Ministerium für Kultus, Jugend und Sport Baden-Württemberg: Pressemitteilung, Neues Profulfach IMP, 14.08.2018. <https://www.km-bw.de/,Len/Startseite/Service/2018+08+14+Neues+Profulfach+IMP>, Abruf am 11.10.2018.
- [Mi19] Ministerium für Kultus, Jugend und Sport Baden-Württemberg: Neue Wege für mehr Bildungschancen: Landtag beschließt Qualitätskonzept. https://km-bw.de/,Lde_DE/Startseite/Service/2019+02+13+Landtag+beschliesst+Qualitaetskonzept+fuer+Schulen. Abruf am 01.06.2019.
- [Re19] Regierungspräsidium Stuttgart: Informatik – Fortbildungen. <https://rp.baden-wuerttemberg.de/rps/Abt7/Ref75/Fachberater/Seiten/Informatik.aspx>. Abruf am 01.06.2019.

TPA-Modell

Vom Denken zum Handeln in der informatischen Bildung

Urs Leo Meier¹

Abstract: Durch die Einführung des neuen Fachs Medien und Informatik im Rahmen des Lehrplans 21 an Deutschschweizer Volksschulen halten u.a. informatische und medienpädagogische Inhalte Einzug in den Unterricht ab Kindergartenstufe. Lehrpersonen aller Schulstufen sind daher gefordert, ihre eigenen Kompetenzen weiterzuentwickeln, um ihre Schüler*innen im Lernprozess zu unterstützen. Insbesondere der Programmierunterricht benötigt viele Fertigkeiten auf verschiedenen Ebenen. Um die Abläufe eines Problemprozesses beim Programmieren für Lehrpersonen verständlich aufzuzeigen, hat der Autor ein Modell entwickelt. Das sogenannte TPA-Modell (Thinking, Processing, Acting) zeigt die grundlegenden Arbeitsschritte beim Lösen eines Problems aus dem Alltag bis hin zur Verwirklichung in einem physikalischen Modell wie z.B. einem Roboter oder Microboard. Nach der Entwicklung des TPA-Modells wurde es in der Praxis eingesetzt, um problembasierte Aufgaben zu reduzieren und die grundlegenden Programmierkonzepte zu vermitteln.

Keywords: Computational Thinking, CS unplugged, Algorithmen, Programmieren, Physical Computing, Problembasierte Aufgaben

1 Einleitung

Aktuell wird an Deutschschweizer Volksschulen mit Inkraftsetzung des Lehrplans 21 das Fach Medien und Informatik ab Kindergartenstufe eingeführt. Inhaltlich wird dabei zwischen medienpädagogischen und informatischen Inhalten sowie Anwendungen unterschieden. Im Bereich Informatik sind Begriffe wie Computer Science Unplugged (folgend als „unplugged“ bezeichnet), Algorithmen, Programmieren etc. neu. In Weiterbildungen bei Lehrpersonen hat der Autor festgestellt, dass die genannten Begrifflichkeiten oft nicht verstanden sowie falsch angewendet werden. Aus diesem Grund hat der Autor ein Modell entwickelt, welches die Begriffe richtig verortet und einen Überblick vermittelt. Das Modell zeigt erstens das Denken (Thinking oder Computational Thinking) auf, welches anschliessend zu einem Programm (Processing) führt und in einem dritten Schritt im Physikalischen (Acting) umgesetzt wird. Das TPA-Modell zeigt im Überblick die grundlegenden Arbeitsschritte beim Lösen eines Problems aus dem Alltag bis hin zum Verwirklichen in einem physikalischen Modell wie einem Roboter oder Microboard.

¹ Pädagogische Hochschule Luzern, Dozent Medien und Informatik, Sentimatt 1, 6003 Luzern,
urs.meier@phlu.ch

2 TPA-Modell

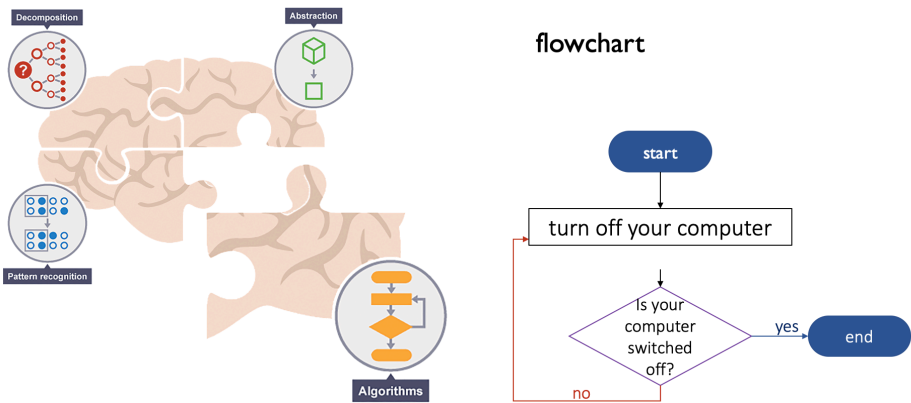
2.1 Thinking

Das Denken steht im Modell an erster Stelle. Ein Problem aus der analogen Welt soll so modelliert werden, damit es von einem Computer gelöst werden kann. Im Paper *New frameworks for studying and assessing the development of computational thinking* haben Brennan und Resnik (2012) folgendes Zitat für Computational Thinking verwendet: *“the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent”*. [BR12, p. 2]

Das Zitat setzt den Schwerpunkt auf einen Denkprozess, der vorrangig ein Problem analysiert und aufbereitet, damit das Problem von einem Computer verarbeitet werden kann.

Das vorliegende TPA-Modell stützt sich bildlich wie inhaltlich auf das Computational-Thinking-Modell aus England, das nach der Bildungsinitiative im Jahre 2014 von BBC [BBC14] veröffentlicht wurde. Das Computational-Thinking-Modell der BBC ist heute weit verbreitet und wird unter anderem auch bei Google im Education-Programm [CT15] eingesetzt und basiert auf folgenden vier Begriffen.

- **Decomposition:**
Grosse Problemstellungen auf kleinere lösbarere reduzieren.
- **Pattern Recognition:**
Erkennen von Mustern und Regelmässigkeiten in Daten.
- **Abstraction:**
Unnötige Informationen finden, sie eliminieren und somit Informationen auf Wichtiges reduzieren.
- **Algorithm:**
Entwickeln von Anweisungen, die das Problem oder ähnliche Probleme Schritt für Schritt lösen.



Thinking

Abbildung 1 TPA-Modell: Thinking

Im ersten Teil Thinking wird ein Problem so analysiert und aufbereitet, damit als Ziel eine Abfolge von Anweisungen (Algorithmen) ausgearbeitet wird, die einem Computer übergeben werden können. Als symbolische Darstellung für den Algorithmus eignet sich ein Flussdiagramm oder Pseudocode.

2.2 Processing

Die Erkenntnisse aus dem Denkprozess können nun an einen Computer übergeben werden, der sie ausführen kann. Die Kommunikation vom Menschen mit dem Computer wird Programmieren genannt. Hromkovič (2009) [Hr09, p. 34] bezeichnet das Programmieren als eine *Sprache zur Kommunikation mit dem Rechner*.

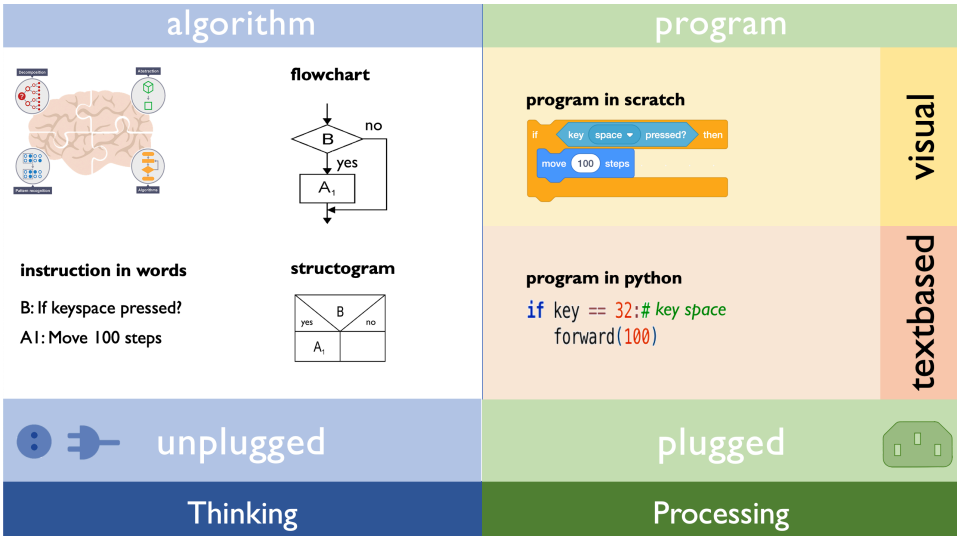


Abbildung 2 TPA-Modell – Processing

Das TPA-Modell zeigt einen Überblick der Tätigkeiten ohne Computer (unplugged) und mit dem Computer (plugged) auf. Des Weiteren wird die Unterscheidung von Algorithmen und Programmen deutlich und verständlich.

Je nach Aufgabe und Unterrichtsziel eignet sich eine textbasierte beziehungsweise eine visuelle Programmiersprache besser für die Anweisungen an den Computer.

Um die Arbeitsweise von Computern zu verdeutlichen, wird im TPA-Modell in Abbildung 3 eine zusätzliche kommunikative und technische Ebene eingeführt. Sie zeigt das Zusammenspiel von Software und Hardware. Die Anweisung der menschlichen Sprache führt über die Programmiersprache zu den Anweisungen in Maschinensprache, die das Ausführen eines Programmes mit der Recheneinheit ermöglicht. Auf die Unterscheidung von Interpreter und Compiler sowie JiT (Just in Time) -Compiler wird bewusst nicht eingegangen, sondern der Vorgang wird einfachheitshalber als Kompilieren dargestellt. Wichtig ist hier die Information, dass der Programmcode nochmals in einen Maschinencode übersetzt wird, der anschliessend von der Recheneinheit ausgeführt werden kann. Die Recheneinheit wird zusätzlich mit einem Transistor als kleinste Schalteinheit abgebildet.

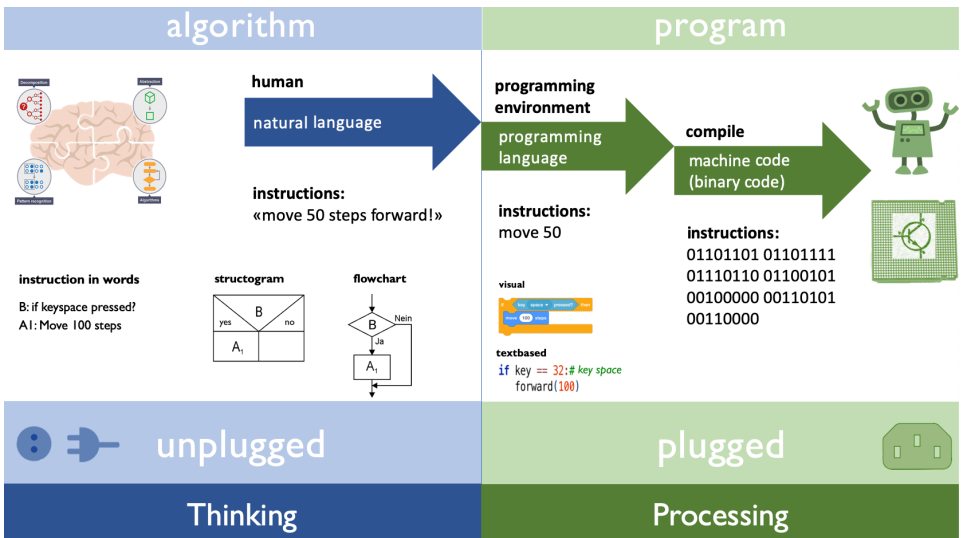


Abbildung 3 TPA-Modell - Processing and Compiling

2.3 Acting

Das Acting zeigt das Programm in Aktion bzw. Handlung und dient als Rückmeldung, ob das eingangs gestellte Problem gelöst wird. Das EVA-Prinzip [RE17] wird hier vertikal dargestellt und zeigt die physikalische Eingabe mit Tastatur oder Sensor, deren Verarbeitung in einem Prozessor sowie die Ausgabe auf einem Bildschirm bzw. einem Aktor.

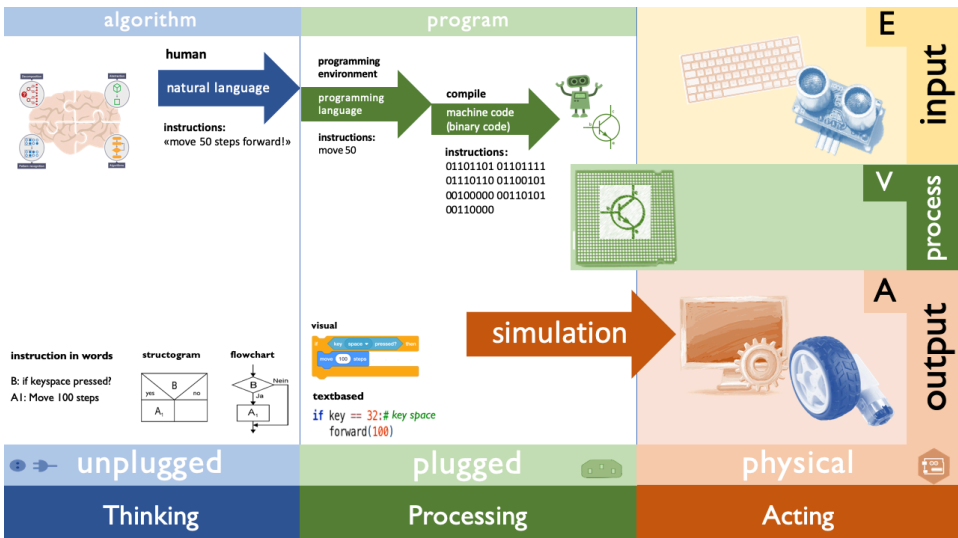


Abbildung 4 TPA-Modell – Acting

Das TPA-Modell zeigt im Überblick auf, wie vom Denkprozess bis hin zur Anzeige auf dem Bildschirm viele kleine Schritte nötig sind. Daraus resultierend kann aufgezeigt werden, wie und wo Fehler bei der Entwicklung von Software entstehen können. Besonders wertvoll ist die Sichtweise für Lehrpersonen, die zukünftig Algorithmen und Programmieren unterrichten. Der Überblick zeigt auf, wo bei Projekten von Schüler*innen nach möglichen Fehlern gesucht werden kann. Fehlt beispielsweise eine physikalische Verbindung beim Input, habe ich mich vertippt (plus/minus) im Programmcode oder ist meine Überlegung im Flussdiagramm falsch?

Für die Fachdidaktik bietet das TPA-Modell die Möglichkeit, einzelne Aspekte aufzunehmen und auf diese im Unterricht einzugehen.

3 Umsetzung

Das TPA-Modell wird in der Aus- und Weiterbildung im Rahmen von Semestermodulen oder CAS Medien und Informatik für Lehrpersonen eingesetzt. Es hilft den Teilnehmenden, Begriffe richtig zu verorten und einen Überblick zu gewinnen. Als Schwerpunkt wird das TPA-Modell bei problembasierten Aufgaben eingesetzt. Es wird bewusst bei kleinen und einfachen Problemen angewendet. Die grundlegenden Programmierkonzepte können dadurch vermittelt werden.

Im konkreten Einsatz bei problembasierten Aufgaben werden folgende Ziele verfolgt:

- Algorithmisches Denken fördern
- Ein Problem in kleine lösbare Probleme reduzieren
- Lösungen für Probleme in Worten und mit Diagrammen darstellen
- Umsetzung von Algorithmen in Programmen
- Testen der angedachten Lösungsstrategien in Programmen

3.1 Aus- und Weiterbildung

Das TPA-Modell eignet sich sehr gut, damit die Begriffe der Informatik korrekt eingesetzt und verwendet werden. Das TPA-Modell zeigt den Lehrpersonen einerseits die Komplexität des Informatik- bzw. Programmierunterrichts, hilft aber andererseits mit dem Überblick, auch Ängste und Unsicherheiten abzubauen sowie Vertrauen zu schaffen.

Weiter soll das TPA-Modell aufzeigen, wie fachdidaktische Begriffe erklärt und richtig verortet werden können. Es wird unter anderem dargelegt, wie man Programmieren in Schritten vermitteln kann. Ein Schwerpunkt ist die Förderung des algorithmischen Denkens und das Lösen von Problemen.

Konkret wird das TPA-Modell in Aus- und Weiterbildungen herangezogen, um aus einer Problemstellung kleine Teilprobleme herauszuschälen und zu vermitteln.

Sollen Schülerinnen und Schüler beispielsweise ein Rennspiel programmieren, könnten sie folgende Teilproblemstellungen ermitteln:

- Steuerung eines Autos mit Tasten
- Das Auto so steuern, damit es immer auf der Bahn bleibt
- Runden zählen
- Tempo messen
- Akkustand bei Elektro-Formel 1 anzeigen
- Hindernisse einbauen
- ...

Für eine konkrete Umsetzung wird hier das Teilproblem eines Akkustands dargelegt. Das Teilproblem wird reduziert auf das Zählen mit einer Variablen. Damit das Problem analysiert werden kann, müssen in einem ersten Schritt Fragen gestellt werden.

- Welche Informationen benötige ich?
- Wie zähle ich zurück?
- Wo starte ich?
- Welche Schritte nehme ich?
- Wie benenne ich die Variable?
- ...

In den nächsten Schritten wird das Teilproblem in Worten (Pseudo-Code), als Flussdiagramm und als blockbasiertes Programm dargestellt.

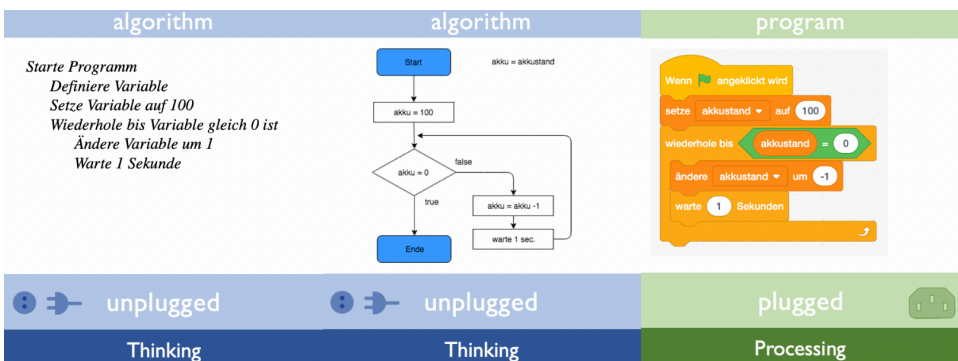


Abbildung 5 Problembasierte Aufgaben - Teilschritte

Das so analysierte Teilproblem zeigt die verschiedenen Darstellungsformen mit entsprechender Abstraktionsebene gut auf und hilft, die erwähnten Ziele zu verfolgen.

3.2 Beispiel-Umsetzung

Das vorgestellte TPA-Modell wurde für problembasierte Aufgabenstellungen in der Ausbildung angehender Lehrer*innen für die Sekundarstufe 1 während eines Semestermoduls eingesetzt. Exemplarisch wird folgend eine gelungene Semesterarbeit von Berger [Be18] aufgezeigt. Der vorliegende Ausschnitt zeigt auf, wie bei einem Spiel das Teil-Problem «zufälliges Auftauchen eines Käfers» analysiert und schrittweise in einem Programm umgesetzt wird.

Informatik und Medien – Programmierung des Zufalls – Arbeitsblatt

Zufälliges Auftauchen

Wie erscheint der Käfer unvorhersehbar am linken Spielfeldrand?


SCHRITT 1	SCHRITT 2	SCHRITT 3
<p>Überlegungen</p> <ul style="list-style-type: none"> • Zeitliche Variabilität durch Zufallsverzögerung (Warte 1 bis 6 Sekunden) • Örtliche Variabilität durch Platzierung des Klons mit Zufallswert auf der y-Achse <p>Formulierung</p> <ul style="list-style-type: none"> • Wenn Spiel startet: Käfer erstellt Klon • Wiederhole fortlaufend folgende Schritte: <ul style="list-style-type: none"> • Warte zwischen 1 und 6 Sekunden <i>(SuS können auch andere Werte und deren Auswirkungen testen)</i> • Gehe zu einem anderen zufälligen Anfangspunkt (y-Variabel) • Erstelle Klon 	<p>Flussdiagramm (ohne Endpunkt)</p> <pre> graph TD Start([Start]) --> ErstelleKlon[Erstelle Klon] ErstelleKlon --> Warte[Warte (1s bis 6s)] Warte --> GeheZu[Gehe zu x = -225 y = -160 bis +160] GeheZu --> ErzeugeKlon[Erzeuge Klon] ErzeugeKlon --> Warte </pre>	<p>Programmierung Ergänzung des Codes «Käfer» in vorliegender Scratch-Datei.</p> 

Abbildung 6: Modul problembasierte Aufgaben – Ausschnitt Semesterarbeit Berger

3.3 Unterricht SEK I (7-9 Schuljahr)

Das TPA-Modell mit dem Schwerpunkt der problembasierten Aufgaben wird im Frühjahr 2019 an zwei Klassen des 9. Schuljahres an einem Gymnasium umgesetzt. Relevante Erkenntnisse können an der *GI-Fachtagung Informatik und Schule "Informatik für alle"* im Rahmen des Workshops präsentiert werden.

4 Fazit und Ausblick

Als erstes Fazit lässt sich auf Grund der Semesterarbeiten der Studierenden feststellen, dass die Fokussierung auf das Problemlösen in kleinen Teilschritten für den Lernprozess unterstützend war und die Programmierkonzepte so besser vermittelt werden konnten.

Bei der Weiterbildung von Lehrpersonen konnte mit dem TPA-Modell der Bereich Informatik des Dag-Stuhl-Dreiecks [GI16] klarer vermittelt und vor allem die Begriffe Algorithmen und Programmieren deutlicher verortet werden. Weiter wird mit der Unterscheidung von *unplugged* und *plugged* die Denkweise vermittelt, dass Informatikunterricht auch ohne Computer stattfinden kann.

Die praktische Umsetzung auf der Zielstufe wird zeigen, wie nachhaltig die Art und Weise des Programmierunterrichts gestaltet werden kann.

5 Literaturverzeichnis

- [BR12] K. Brennan und M. Resnick, „harvard.edu,“ 2012. [Online]. Available: <http://scratched.gse.harvard.edu/ct/files/AERA2012.pdf>. [Zugriff am 26 01 2019].
- [BBC14] BBC, „<https://www.bbc.com/bitesize/guides/zp92mp3/revision/1>,“ 2014. [Online]. Available: <https://www.bbc.com/bitesize/guides/zp92mp3/revision/1>. [Zugriff am 26 01 2019].
- [CT15] computationalthinkingcourse, „computationalthinkingcourse.withgoogle.com,“ google.com, 2015. [Online]. Available: <https://computationalthinkingcourse.withgoogle.com/unit?lesson=8&unit=1>. [Zugriff am 26 01 2019].
- [Hr09] J. Hromkovič, Sieben Wunder der Informatik, Wiesbaden: Vieweg+Teubner, 2009.
- [RE17] P. D. M. Rieger, P. Eisoldt, D. Schlichtenberger und T. Scheible, Applied Computer Systems, Hochschule Albstadt-Sigmaringen: Institut für wissenschaftliche Weiterbildung, 2017.
- [Be18] J. Berger, „Leistungsnachweis Modul pbA 3. Semester Medien und Informatik,“ PHLU, Luzern, 2018.
- [GI16] I. (GI), „[gi.de Website](https://gi.de/themen/beitrag/dagstuhl-erklaerung-bildung-in-der-digital-vernetzten-welt-1/),“ 2016. [Online]. Available: <https://gi.de/themen/beitrag/dagstuhl-erklaerung-bildung-in-der-digital-vernetzten-welt-1/>. [Zugriff am 02 02 2019].

Mischen possible. Didaktische und unterrichtspraktische Überlegungen zum Mischen im Informatikunterricht.

Peter Micheuz¹


Abstract: In diesem Praxisbericht geht es exemplarisch um Zugänge, Werkzeuge und Methoden zum Thema Mischen. Verglichen mit der Vielfalt an Sortieralgorithmen, wird dieses Thema im Bereich digitaler Datenverarbeitung kaum explizit gemacht. Aber warum eigentlich? Dieser Beitrag befasst sich mit der Didaktik des „Anordnens“ von digitalen Objekten als Teil eines zeitgemäßen Informatikunterrichts, in dem unterrichtspraktische Überlegungen und Konzepte mit unterschiedlichen Softwarewerkzeugen und Entwicklungsumgebungen demonstriert und implementiert werden.

Keywords: Algorithmen, Computational Thinking, Programmierung, Sortieren und Mischen,

1 Einleitung

Es begab sich kürzlich im Informatikunterricht einer 5. Klasse (9. Jahrgangsstufe), in dem das „Schweizer Messer“ unter den Softwarewerkzeugen, nämlich Excel, als Vehikel für informatische Aktivitäten behandelt wurde: Der Autor fragt die Schüler, wie eine nach den Punkten des Biberbewerbs an der Schule absteigend sortierte Liste mit Schülernamen und Klasse wieder schnell „durchmischt“ werden kann, um die Sortierung nach Klasse und Punkten als Übungsaufgabe erneut stellen zu können.

Klasse	Nachname	Vorname	Punkte
1B	Nachname1	Jakob	112
1B	Nachname2	Maja Kathrina	108
1B	Nachname3	Filiz	98
1B	Nachname4	Anna-Giulia	96
1B	Nachname5	Moritz	94
1B	Nachname6	Leonie	91
1B	Nachname7	Fabienne	88
1B	Nachname8	Elias Noel	87
1B	Nachname24	Lara	30
1C	Nachname26	Viktoria	108
1C	Nachname27	Lara	100
1C	Nachname28	Benjamin	92
1C	Nachname29	Gerald Martin	75
1C	Nachname30	Georg Erich	72
1C	Nachname31	Tobias Felix	68



Klasse	Nachname	Vorname	Punkte
5B	Nachname326	David	164
6B	Nachname364	Elias	94
4D	Nachname278	Manuel	148
2A	Nachname113	Kilian Noel	36
2B	Nachname130	Christoph	56
3D	Nachname219	Maximilian	100
5A	Nachname314	Minea	100
1C	Nachname26	Viktoria	108
2C	Nachname151	Melissa Chiara	70
2D	Nachname175	Daniel Luca	63
3D	Nachname212	Mikka	128
1B	Nachname24	Lara	30
2A	Nachname107	Shu Yi	84
2A	Nachname103	Kimmy Gloria Mari	92
1D	Nachname54	Lin	84
3F	Nachname267	Lukas	43

Abbildung 1: Von der Ordnung zur Unordnung

¹ Alpen-Adria Gymnasium Völkermarkt, Pestalozzistraße 1, A-9100 Völkermarkt, peter.micheuz@aon.at

Ich habe nicht schlecht gestaunt, als mir gleich mehrere Schüler, ohne lange überlegt zu haben, folgende Lösung vorschlugen: „Eine leere angrenzende Spalte mit Zufallszahlen befüllen, die Daten nach diesen Zufallszahlen sortieren und anschließend die Spalte mit den Zufallszahlen wieder löschen“. Auf meine Frage, wie sie so schnell draufgekommen sind, gab es eine überraschende Antwort: „Sie haben es uns vor einiger Zeit einmal so nebenbei gezeigt ...“.



Auf meine weitere Frage, warum die Entwickler von Microsoft nicht an einen „Mischbutton“ neben den Sortieroptionen in den komplexen Menübändern gedacht haben, gab es noch keine schlüssigen Antworten. Diese gab es (auch noch) nicht auf die Fragen, wie im Computer Pseudozufallszahlen erzeugt werden, und wie die Inhalte der

Zellen so angeordnet bzw. zielgerichtet vertauscht werden, dass die Werte in den Zellen auf- oder absteigend sortiert sind.

Im Fokus der weiteren Überlegungen liegt in diesem Beitrag, bis auf die Erwähnung des Mergesort, nicht das zielgerichtete Sortieren, sondern die zufallsgesteuerte Produktion von Unordnung.

2 Lottozahlen

Wenn von Zufall und unsortierten Zahlen die Rede ist, kommen einem fast unweigerlich Lottozahlen in den Sinn. Die offene Aufgabe, wie einem Computer beizubringen ist, Lottozahlen zu erzeugen, wurde im Informatikunterricht mit Sicherheit schon tausende Male gestellt und mit verschiedenen Werkzeugen, Entwicklungsumgebungen, Bibliotheken, unterschiedlichen Algorithmen und Datenstrukturen, und mehreren Programmierparadigmen sowie auf diversen Abstraktionsebenen behandelt und realisiert. Eine auch nur annähernd erschöpfende Behandlung all dieser Zugänge ist an dieser Stelle nicht möglich, aber ein Anknüpfungspunkt und eine Analogie zum Verfahren in der Einleitung soll näher beschrieben werden. Warum nicht die Lottozahlen nach der „Katalysatormethode“ (temporär angelegtes Feld mit Zufallszahlen, siehe Einleitung) als Mischung von sechs unterschiedlichen Zahlen in einem bestimmten Zahlenbereich (in Deutschland 1-49, in Österreich 1-45) erzeugen?

Abb. 2 zeigt die spaltenmäßige Anordnung (Spalte A: Lottozahlen, Spalte B: Zufallszahlen) des „einfachen“ Lottozahlengenerators, der mittels der Neuberechnungstaste F9, dem Selektieren der beiden Spalten und der richtigen Anwendung des Sortierassistenten die Lottozahlen in den Zellen B2:B7 anzeigt. Auf die Möglichkeit der Automatisierung dieser wenigen linearen Handlungsschritte in Form eines „Excel-Makros“ soll an dieser Stelle nicht eingegangen werden, obschon „Makroprogrammierung“ eine empfehlenswerte

Möglichkeit darstellt, einen Grundpfeiler der Informatik, nämlich die Automatisierung, schrittweise erfahrbar zu machen.

	A	B	C	D	E	F	G
1							
2		1	0,44063		2		26
3		2	0,02593		31		42
4		3	0,74237		44		12
5		4	0,34609		37		30
6		5	0,17009		34		34
7		6	0,61382		21		18
8		7	0,70308				
9		8	0,87186				
10					
11					
12		44	0,96311				
13		45	0,54996				

Abbildung 2: Varianten von Lottozahlengeneratoren in MS-Excel

Einen vertieften und lohnenden Ausblick auf den „Zauber“ [FL15] funktionaler Modellierung und deklarativer Programmierung, die allein mit der zielgerichteten und kompetenten Anwendung der Bibliotheksfunktionen von MS-Excel (oder ähnlicher Tabellenkalkulationssoftware) möglich sind, bieten die beiden anderen Varianten der Lottozahlenerzeugung (siehe Abb. 2 rechts) mittels der verketteten Funktionen VERGLEICH, KKLEINSTE, ZEILE sowie der Verwendung der algorithmisch interessantent Funktion RANG.

3 Mischen in der Praxis – „Real World“-Anwendungen

Bevor wir uns mit den für das Mischen notwendigen Elementaralgorithmen und Daten(strukturen) beschäftigen, sei hier eine weitere - möglicherweise unter Informatiklehrenden wenig bekannte und daher möglicherweise exotische – (My)SQL-Anweisung zur Produktion von sechs Lottozahlen angeführt:

```
select zahlen limit 6
from lottozahlen
order by rand()
```

Die Beschreibung dieses Mischbefehls in einem MySQL-Tutorial im Web [MY19] in Abb. 3 führt uns wieder zum Konzept und die Methode der Sortierung nach Zufallswerten zurück und demystifiziert diesen scheinbar widersprüchlichen Befehl „order by rand()“. Diese Widersprüchlichkeit gibt es vordergründig auch beim Mergesort („Sortieren durch Verschmelzen/Mischen“).

MySQL Select Random Records - MySQL Tutorial

www.mysqltutorial.org > MySQL Tips ▾ Diese Seite übersetzen

MySQL select random records using ORDER BY RAND() The RAND() function generates a random value for each row in the table. The ORDER BY clause sorts all rows in the table by the random number generated by the RAND() function. The LIMIT clause picks the first row in the result set sorted randomly.

Abbildung 3: Beschreibung der ORDER BY RAND()- Anweisung im Web

An dieser Stelle darf nicht verschwiegen werden, dass dieser SQL-Befehl ein bestimmtes Maß an wohl vorbereitetem „Umfeldwissen“ in Form von konkreten Werkzeugkenntnissen voraussetzt, wenn er sinnvoll z.B. in einem zeitgemäßen Web-User-Interface eingesetzt werden soll. Das V (Verarbeitung) im EVA-Prinzip ist mit dem obigen SQL-Befehl der leichtere Teil der Übung, weil er als Blackbox verwendet wird und die Realisierung des zugrundeliegenden Misch-Algorithmus bisher noch kein Thema war. Sowohl die Eingabe der Daten als auch die Ausgabe der Daten setzen viel Unterrichtsarbeit voraus, wenn von den Schülern aktive Entwicklerleistungen in Form von Modellierung, Programmierung und dem Gestalten von bereits einfachen Benutzerschnittstellen eingefordert werden.

Anhand von drei verschiedenen Zufallsfragen, die aus einem größeren Fragepool auf einem Webportal nach jedem Seitenaufruf neu generiert werden, soll dies illustriert werden.

- ▶ Was bedeutet True Color? Was bedeutet RGB im Zusammenhang mit Bildschirmfarben?
- ▶ Welche Anwendungsprogramme gibt es? Ist ein Betriebssystem auch ein Anwendungsprogramm?
- ▶ Was bedeutet "etwas zippen"? Wann macht es besonders viel Sinn, wenn etwas gezippt wird?

Für diese Ausgabe ist das folgende „Codeschnipsel“ auf einem Webserver als „Mischung“ von HTML, PHP und (My)SQL verantwortlich:

```
mysql_select_db("unterstufe");
$abfrage=mysql_query("select frage
                    from basisfragen
                    order by rand()");
for ($i=0;$i<3;$i++) {
    $row = mysql_fetch_row($abfrage);
    echo("<center><b><img src='pfeil.gif'>".$row[0].
    </b></center>");
}
```

Auf der gleichen Portalseite [SI19] werden nicht nur Texte durchmischt, sondern nach jedem Seitenaufruf drei verschiedene Hardwarebausteine angezeigt (siehe Abb. 4). Diese Anzeige wird aus didaktischen Gründen unter Ausnutzung einer der vielen Bibliotheksfunktionen und der Datenstruktur ARRAY der serverseitigen und weitverbreiteten Programmiersprache PHP realisiert.

... viele Hardwarebausteine



Abbildung 4: Zufällige Anordnung von Bildern

Im folgenden kurzen, aber gehaltvollen und leistungsfähigen serverseitigen HTML-PHP-Skript ist der SHUFFLE-Befehl als PHP-Bibliotheksfunktion nicht zu übersehen:

```
echo("<p><h2>... viele Hardwarebausteine</h2>");
$zz = range(0,18);
shuffle($zz);
$bild1="bilder/" . $zz[0] . ".gif";
$bild2="bilder/" . $zz[1] . ".gif";
$bild3="bilder/" . $zz[2] . ".gif";
echo("<p><center><img src=' $bild1 '>
      <img src=' $bild2 '>
      <img src=' $bild3 '>
    </center></p>");
```

Puristen und Dogmatiker unter den Informatikern werden einwenden, dass Skriptsprachen wie PHP, mit ihrer schlampigen Art mit Datentypen umzugehen, zu einem schlechten Programmierstil verleiten. Ich lasse das unkommentiert stehen. Der geneigte Leser möge sich dazu selbst ein Urteil bilden, sich an Expertenforen wie z.B. quora.com beteiligen und daraus die Konsequenzen für den (eigenen) Informatikunterricht ableiten.

Die Anweisung `$zz = range(0,18)` legt ein Array `$zz[0]=0, $zz[1]=1, ... $zz[18]=18` an, der Befehl `shuffle($zz)` durchmischt zufällig die Inhalte der Felder, so einfach ist das! Die hier nicht offensichtlichen Tücken des Skripts liegen weniger darin, die Wirkung des Shuffle-Befehls zu verstehen. Probleme bereiten da schon eher die kunstvolle Anwendung („Maskierung“) der Anführungszeichen und das Konzept von externen, auf dem Server im Ordner „bilder“ gespeicherten Hardware-Bildern im GIF-Format (0.gif, 1.gif, ...18.gif) sowie das Konzept ihrer internen Anbindung an das Array `$zz`.

Nach diesem kurzen Ausflug in die Webprogrammierung inklusive werkzeugabhängiger Zugänge zum Mischen von Zahlen, Text und Bildern mit der „Katalysatormethode“ (siehe oben), der (My)SQL-Anweisung „order by rand()“ und dem Shuffle-Befehl aus der PHP-Funktionsbibliothek, verlassen wir die Komfortzone der beschreibenden Methode und betreten die prozedurale Welt elementarer Algorithmen.

4 Mischen als zufälliges Tauschen von Variableninhalten

Dieser Beitrag darf auch als späte und logische Fortsetzung eines INFOS-Beitrages „Ein kleiner Schritt für LehrerInnen, ein großer Schritt für SchülerInnen“ aus dem Jahr 2005 gesehen werden [MI05], in dem es im 6. Kapitel „Ringtausch und Umfüllaufgabe“ hieß: „In diesem Kapitel ist nur angedeutet worden, welch reichhaltiges, didaktisch wertvolles Potential bereits in (einfachen) Aufgabenstellungen - wie etwa dem Tausch der Inhalte zweier Variablen - steckt. Die Legitimation, den "Platztausch" im Informatikunterricht zu thematisieren, ist aber auch durch unzählige lebensweltliche Anwendungen und die Tatsache gegeben, dass er das Fundament vieler Algorithmen darstellt, in denen es um Misch- sowie Sortierverfahren geht.“

Beginnen wir mit einer konkreten Programmieraufgabe, diesmal in Javascript, einer Sprache, die derzeit einen wahrhaftigen Boom erlebt und an der man nicht vorbeikommt, wenn es um die Entwicklung von Frontend-Anwendungen im Web geht. Unsere Aufgabenstellung lautet so: *„Eine bestimmte Anzahl an Schülerbildern sind auf einer Schulwebsite in tabellarischer Form nach jedem Seitenaufruf durchmischt anzuordnen.“*



Anmerkung: Die vier Menschen links existieren in Wirklichkeit nicht. Sie sind der auf Artificial Intelligence und Deep Learning-Algorithmen beruhenden Website „thispersondoesnotexist.com“ entnommen.

Im Unterschied zur serverbasierten Skriptsprache PHP verfügt die Programmiersprache Javascript zwar über eine Methode „sort“ der Objektklasse „array“, aber leider (oder aus didaktischer Sicht Gottseidank) keine Methode „shuffle“, die es mit einer Anweisung gestatten

würde, die Bildnummern zu durchmischen. Es wird an dieser Stelle auch von der Methode des „Katalysatorfeldes“ (Sortierung eines temporären Zufallsfeldes) Abstand genommen. Der Leser ist eingeladen auszuprobieren, ob dieser Workaround einfach zu realisieren ist.

Stattdessen bewegen wir uns (notgedrungen) auf die Ebene des „Dreiecks- bzw. Ringtausch“ und realisieren den Mischvorgang durch wiederholtes Tauschen der Array-Inhalte. Wenn es in einer Entwicklungsumgebung keine (Bibliotheks)Funktion oder keine entsprechende Methode gibt, steht es jedem frei, sich im algorithmischen Denken und in der Implementierung derselben zu versuchen. Das Kernstück des folgenden HTML-Files, das vollständig in Javascript codiert und lauffähig ist, wenn die Fotos im gleichen Ordner wie das HTML-Skript zur Verfügung stehen, ist die Funktion „shuffle“ mit dem Array „fotos“. Dieses dient sowohl als Eingabeparameter als auch als Ausgabefeld.

```
<script>
function shuffle(fotos) {
    var laenge = fotos.length, temp, index;
```



```

while (laenge > 0) {
    index = Math.floor(Math.random() * laenge);
    laenge = laenge - 1;
    temp = fotos[laenge];
    fotos[laenge] = fotos[index];
    fotos[index] = temp;
}
return fotos;
}

var bilder = new Array()
bilder[0] = 'name_a.jpg'
bilder[1] = 'name_b.jpg'
bilder[2] = 'name_c.jpg'
bilder[3] = 'name_d.jpg'
shuffle(bilder);
var p = bilder.length;
for (var i = 0; i <= 3; i++)
    { var bildnummer = i;
      document.write('<img src="'+bilder[bildnummer]+'>');
      if (i % 2 == 1) {
          document.write('<br>');
      }
    }
}
</script>

```

Der in der Funktion „SHUFFLE“ implementierte Algorithmus ist optimal mit einer Komplexität von $O(n)$ und wird auch als Fisher-Yates-Shuffle bezeichnet.

```

function randperm(n)
    P = [1:n]           // Initialisierung mit der identischen Permutation
    for i = n downto 1 // Schleife über die Einträge von P
        z = random(i) // Gleichverteilte Zufallszahl zwischen 1 und i
        swap(P(i),P(z)) // Vertausche die Zahlen an den Stellen i und z
    end
    return P

```

Abbildung 5: Pseudocode, Quelle: https://de.wikipedia.org/wiki/Zufällige_Permutation

Ronald Aylmer Fisher (1890-1962) und Frank Yates (1902-1994) waren bedeutende Statistiker des 20. Jahrhunderts. Dass das optimale Mischen nach ihnen benannt ist, ist kein Zufall. Zufallsanordnungen, mathematisch ausgedrückt zufällige Permutationen, werden unter anderem bei der Analyse von Sortierverfahren, in der Kryptographie, bei randomisierten Algorithmen und vor allem bei Simulationen benötigt.

Übrigens eignet sich der Fisher-Yates-Algorithmus auch vorzüglich für die (optimale) Erzeugung von Lottozahlen, wenn diese mit der Datenstruktur Array realisiert wird. Dies passiert weltweit in tausenden Lottoannahmestellen beim Kauf von computergenerierten Zufallstipps.

5 Simulationen und weitere Unterrichtsbeispiele

Interessante Anmerkungen zum Bildungswert der Informatik finden sich bei Rechenberg [RE94], der im Kapitel „Informatik und Kultur“ vier wesentliche Bereiche der Informatik herausstreicht. Erstens das algorithmische Denken, das beim Verstehen von Konzepten jeder Art hilft [vgl. KN74], aber „keinem auf Anhieb gelingt, und mühsam mit großer Geduld erworben werden muß“, zweitens das Studium und die Meisterung komplexer Zusammenhänge, und drittens, dass der Computer „ein so vielseitiges Instrument zur Simulation und zum Ausführen von Gedankenexperimenten ist“. Der vierte Bildungswert der Informatik besteht für Rechenberg darin, zu zeigen, was man mit Computern vor allem nicht automatisieren kann. Damit hat er nicht zuletzt die aktuelle Diskussion um die (Auswirkungen der) Artificial Intelligence bereits vor 25 Jahren vorausgesehen.

Im Folgenden werden Aufgaben und Anregungen zum Thema Mischen im Zusammenhang mit einfachen Simulationen und unterrichtspraktischen Anregungen gestellt.

Sitzanordnungen und BOGO-Sort

Die Antwort auf die Frage, wie viele Sitzanordnungen es in einer Klasse mit z.B. 25 Stühlen gibt, verblüfft nicht nur die SchülerInnen, die sich - das erste Mal mit dieser Frage konfrontiert - erfahrungsgemäß extrem verschätzen.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
D	E	K	A	B	O	J	L	C	T	W	H	U	M	G	N	F	Y	I	X	P	Q	V	S	R
J	U	C	L	K	V	Q	G	T	W	I	O	R	H	D	E	P	X	N	A	S	M	Y	F	B
E	R	S	F	Y	N	P	I	L	C	T	V	B	O	A	K	Q	G	J	D	H	U	X	M	W
O	A	C	R	K	Q	E	N	F	V	U	P	X	G	S	L	D	J	I	Y	H	W	B	M	T
...
...

For stelle = 25 To 1 Step -1
 zufall = Int(stelle * Rnd) + 1
 h = Cells(1, zufall)
 Cells(1, zufall) = Cells(stelle)
 Cells(stelle) = h

Next

Abbildung 6: Permutation von 25 Buchstaben

Die Zahl $25! = 15.511.210.043.330.985.984.000.000$ sprengt das menschliche Vorstellungsvermögen und sollte einen ambitionierten Klassenlehrer davor bewahren, in einem Schuljahr alle Mischvarianten physisch ausprobieren zu wollen, und wohl auch den Informatiklehrer, alle Permutationen digital anzeigen, geschweige denn ausdrucken zu wollen. Sehr wohl können die Computer im Informatikraum damit beschäftigt werden, alle(!) Buchstaben-Mischungen zu produzieren und darauf zu warten, bis der Buchstabensalat wieder sortiert ist. Dieses Sortierverfahren - solange zufällig mischen, bis alle Objekte sortiert sind - führt laut Wikipedia (<https://de.wikipedia.org/wiki/Bogosort>) die Bezeichnungen Bogosort, Alexsort, Monkeysort oder Stupidsort. Im krassen Unterschied dazu sei

hier das hocheffiziente Mergesort (Sortieren nach dem Prinzip Teile und Herrsche mit anschließendem Verschmelzen/Mischen) angeführt, zu dem es mit Stand 10.2.2019 über eine Million Google-Treffer mit unzähligen guten Beschreibungen und Erklärvideos gibt.

Magisches Quadrat und eine Biber-Aufgabe

Fachdidaktisch und methodisch interessant im Zusammenhang mit dem Mischen ist auch die Suche nach dem „Magischen Quadrat“ in allen Varianten (siehe Abb. 7).

	A	B	C	D	E	F	G	H	I	J	K
1	5	2	8	MAGISCHES QUADRAT							
2	3	1	4	Die Zahlen von 1-9 in Matrix schreiben							
3	9	6	7	Vertauschen							
4				Mischen							
5				Lösung suchen - Brute force Methode							
6	17	9	19	15	8	22	13	18	Summen		
7	Minimum	8									
8	Maximum	22	850								

Der Biber-Informatiker wählt Balken nach folgenden Regeln aus:

1) Wenn der genommene Balken kürzer als der folgende ist, wird er für den Fußboden verwendet.

2) Wenn er länger ist, wird er für das Dach verwendet.

Er nimmt die Balken der Reihe nach und wendet wiederholt die 1. und die 2. Regel an.

Welche Balken werden für das Dach verwendet?



Abbildung 7: Magisches Quadrat und Biber Aufgabe

Die Abbildung links zeigt eine von unzähligen Implementierungen des Magischen Quadrats (hier in Excel). Die Benutzerschnittstelle mit vier Buttons und einigen berechneten Feldern weist auf einen interessanten didaktischen Zugang zur Lösung dieser Aufgabe hin. Es handelt sich hier um eine Mischform der von Excel bereitgestellten Funktionen (verdeckte Algorithmen: Summen, Minimum und Maximum) und in VBA (Visual Basic for Applications) codierten Tausch- und Mischalgorithmen, die im Hintergrund arbeiten.

Biber-Aufgaben bieten vielfältige Anknüpfungspunkte für informatisches Denken und Handeln, so auch exemplarisch die in Abb. 6 dargestellte originelle „Sortieraufgabe“. In der Grundschule sollten die Balken/Stäbe haptisch (aus)sortiert werden, in der Sekundarstufe I bietet sich funktionale Modellierung an, und in der Sekundarstufe II ist eine Simulation inklusive Programmierung und Codierung empfehlenswert.

Was hat das mit dem Mischen zu tun? Sehr viel. Warum nicht computergestützte Experimente automatisiert durchführen, die den Zusammenhang zwischen der Anzahl von Dachhölzern und allen(!) $11! = 39.916.800$ möglichen Anordnungen herstellen?

6 Schlussanmerkungen

Dem aufmerksamen Leser wird sicher aufgefallen sein, dass bisher ein weitverbreiteter lebensweltlicher Aspekt des Mischens ausgespart wurde. Es geht um die Kultur der Kartenspiele, bei denen der Zufall und das Kartenmischen eine erhebliche Rolle spielen. Bei dieser haptischen Erfahrung ist die wohl weitverbreitetste Methode jene des „Packelns“

(entspricht dem mehrmaligen Abheben), bei dem mit beiden Händen öfters jeweils zwei nach Gefühl unterschiedliche Stapel vertauscht werden. Dabei soll es schon vorgekommen, dass nicht ordentlich gemischt und getrickst wird. In diesem Zusammenhang sei auf eine ausführliche Quelle für Kartentricks mit informatischem Background „Die Magie der Informatik: Kartentricks“ [CS19] verwiesen, in der u.a. auch der Trick „Perfektes Mischen“ vorgestellt wird.

Kartenspielen und haptisches Kartenmischen sollten im Unterricht der Primarstufe einen Fixplatz haben. Später in der Sekundarstufe I kann die Haptik z.B. in einer Tabellenkalkulation als Teil einer Bedienerschulung (Verschieben, Kopieren, siehe Abb. 8) simuliert

1	2	3	4	5	6	7	8	9	10
5	6	7	8	9	10	1	2	3	4
8	9	10	1	2	3	4	5	6	7

Abbildung 8: Simulation des Abhebens

und dabei versucht werden, mit möglichst wenig Mischungsvorgängen eine gute Durchmischung zu erreichen. Die programmiertechnische Realisierung und die Diskussion, was „gut durchmischt“ heißt, kann in der Sekundarstufe II weitergeführt werden.

Literaturverzeichnis

- [CS19] Die Magie der Informatik: Kartentricks im Rahmen des CS4FN-Projekts (Computer Science for Fun), online unter: <http://www.cs4fn.org/magie>, Stand 10.2.2019
- [FL15] Fleckenstein J. u.a.: Excel – Das Zauberbuch. Markt&Technik, 2015.
- [KN74] Knuth D.E: Computer Science and its Relation to Mathematics. American Mathematical Monthly 1974.
- [MI05] Micheuz P.: Ein kleiner Schritt für LehrerInnen, ein großer Schritt für Schüler. In: Steffen F.: Unterrichtskonzepte für informatische Bildung, INFOS-Tagungsband, 11. GI-Fachtagung Informatik und Schule, 28.-30. September 2005.
- [MY19] MySQL-Tutorial: <http://www.mysqltutorial.org/select-random-records-database-table.aspx>, Stand: 10.2.2019
- [RE94] Rechenberg P.: Was ist Informatik? Hanser-Verlag, 2. Auflage, 1994.
- [SI19] Portal Schulinformatik: <https://www.gym1.at/schulinformatik/unterstufe>, Stand 10.2.2019

Entwicklung und Reflexion einer Unterrichtssequenz zum Maschinellen Lernen als Aspekt von Data Science in der Sekundarstufe II

Simone Opel, Michael Schlichtig, Carsten Schulte¹, Rolf Biehler, Daniel Frischemeier, Susanne Podworny, Thomas Wassong²

Abstract: Die Bereiche „Data Science“ und „Big Data“ sowie ihre technischen, ethischen und gesellschaftlichen Auswirkungen werden zunehmend nicht nur in der Wissenschaft, sondern auch in diversen Medien diskutiert und somit verstärkt auch zu einem wichtigen Thema für alle. Um den Schülerinnen und Schülern der Sekundarstufe II einen theoretisch und fachwissenschaftlich fundierten Einstieg in diesen Themenbereich zu ermöglichen, wurde ein erster Entwurf eines interdisziplinären Curriculums entwickelt, das neben fachlichen Aspekten von Data Science einen Fokus auf sich hieraus ergebende gesellschaftliche Fragestellungen legt. Es werden neben der Konzeption des Kurses die bisherigen Erfahrungen aus der Durchführung – insbesondere in Hinsicht der darin enthaltenen Unterrichtseinheit zum Maschinellen Lernen - berichtet, sowie die sich hieraus ergebenden Implikationen für die Weiterentwicklung dargestellt und diskutiert.

Keywords: Data Science; Maschinelles Lernen; KI; Künstliche Neuronale Netze; Entscheidungsbäume; Big Data; Curriculum

1 Einleitung

Data Science sowie Fragen zu Maschinellern Lernen (ML) und Künstlicher Intelligenz (KI) sind inzwischen in großem Maß in verschiedene Systeme implementiert und werden in allen Bereichen intensiv diskutiert. Auch die gesellschaftliche Diskussion über Funktion, Nutzen und Gefahren dieser Systeme nimmt inzwischen viel Raum ein. Es erscheint uns wichtig, dass diese Fragestellungen in den Unterricht integriert werden. Daher entwickelten wir auf Basis curricularer Ideen aus der Informatik und Mathematik einen ersten Entwurf eines Curriculums für die Sekundarstufe II und führten dies in ein konkretes Unterrichtskonzept über, das im Rahmen des durch die Deutsche Telekom Stiftung ermöglichten Projekts „ProDaBi³“ mit einem Oberstufenkurs erprobt wird. Im Rahmen dieses Artikels stellen wir neben diesem Kurs auch erste Erfahrungen der Umsetzung – insbesondere der Einheiten aus KI und ML – sowie die hinter dem Entwicklungsprozess liegenden Ideen vor.

¹ Universität Paderborn, Didaktik der Informatik, Fürstenallee 11, 33102 Paderborn, vorname.nachname@uni-paderborn.de

² Universität Paderborn, Didaktik der Mathematik, Warburger Str. 100, 33098 Paderborn, nachname@math.uni-paderborn.de

³ ProDaBi – Projekt Data Science und Big Data in der Schule, Projektwebseite: <https://www.prodabi.de>

2 Data Science und Maschinelles Lernen – Aspekte für die Bildung

Data Science, ML und der Umgang mit Big Data geht weit über technische und wissenschaftliche Aspekte unterschiedlicher Disziplinen hinaus und trägt auch ethische, gesellschaftliche und soziale Auswirkungen in sich – daher werden hier nicht umsonst sehr verschiedene Kontexte diskutiert. Um möglichst viele Aspekte zu verstehen und ein gemeinsames Verständnis von *Data Science* zu entwickeln, wurden im Rahmen eines interdisziplinären, internationalen Symposiums⁴ [Pa18] diese Aspekte diskutiert und zwei für uns curricular relevante Bereiche wurden auf Basis der dort geführten Diskussionen identifiziert: Für die *Informatikdidaktik* sind das neben der Entwicklung von Computational Thinking [TD16] insbesondere die Auswirkungen verschiedener Bereiche der Mensch-Maschine-Interaktion auf die Gesellschaft – und damit auch unser Umgang mit Big Data sowie den Methoden und Auswirkungen von Data Science [SBS18]. Im Bereich der *Statistikdidaktik* sollten die fundamentalen Ideen der Statistik um verschiedene Aspekte der *statistischen Kompetenz* [Ri16] erweitert werden. Daher stehen diese Ideen und Aspekte im Zentrum der Entwicklungen.

2.1 Daten und Datenprozesse als strukturgebende Komponenten

Im Gegensatz zur praktischen Informatik stehen im Bereich der *Data Science* nicht die Entwicklung von algorithmischen Strukturen sowie deren (algorithmische) Modellierung und Implementierung im Vordergrund, sondern der Umgang mit *Daten*. Das heißt, die Komplexität der Fragestellungen wird nicht nur durch die eingesetzten Algorithmen bestimmt, sondern wird in hohem Maße durch die Daten und die in ihnen implizit und explizit enthaltenen Informationen beschrieben. Nach der Definition der Empfehlungen GI zu den Bildungsstandards in der Sekundarstufe sind „Daten eine Darstellung von Information in formalisierter Art [...]. Daten werden wieder zu Information, wenn sie in einem Bedeutungskontext interpretiert werden“ ([Rö16], S. 9). Ein Informatiksystem verarbeitet somit nur Daten, die darin enthaltene Information wird durch Interpretation durch den Menschen gewonnen. Daten sind also „nicht nur Zahlen, sie sind Zahlen mit einem Kontext“ [CM97]. Es stellt sich die Frage, wie Schülerinnen und Schüler *Datenkompetenz* erwerben können, die Voraussetzung für einen kompetenten Umgang mit fehlerbehafteten oder unterschiedlich strukturierten Daten innerhalb eines Kontextes ist. Ridsdale et al. definieren *Datenkompetenz* als die prozessorientierte „Fähigkeit, Daten kritisch zu sammeln, zu analysieren, zu bewerten und anzuwenden“ ([Ri15], S. 3). Damit liegt der Schwerpunkt der Handlungen der Lernenden im Umgang mit verschiedenen Daten, so dass „Prozessmodelle zur Datenanalyse“ einen sinnvollen Ausgangspunkt zur Planung von unterrichtlichem Handeln bieten. Der Definition von Ridsdale et al. ähnelt das CRISP-DM-Modell⁵ [Ch00], das ein vollständiges Verfahren zum Umgang mit Daten

⁴ „Perspectives for data science education at school level – Educational contributions from statistics, computer science and sociocultural studies“; <http://go.upb.de/SymposiumProDaBi>

⁵ CRISP-DM = **C**ross-**I**ndustry Standard **P**rocess for **D**ata **M**ining, Phasen: Verstehen der Fragestellung – Verstehen der Daten – Aufbereitung der Daten – Modellbildung – Bewertung des Modells – Einsatz

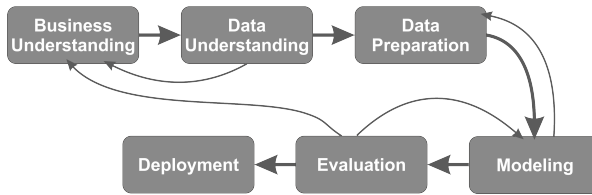


Abb. 1: Das CRISP-DM-Modell als Standardprozessmodell für Data Mining

beschreibt (vgl. Abb. 1). Dieses Prozessmodell erscheint uns einen sinnvollen Rahmen zur Erarbeitung eines Curriculums und entsprechenden Unterrichtsmaterials zu bieten, so dass entschieden wurde, das CRISP-DM-Modell als Basis weiterer Entwicklungen zu verwenden, da dieses Modell den Datenprozess umfassender darstellt als zum Beispiel der ebenfalls häufig verwendete PPDAC-Zyklus⁶.

2.2 Relevanz sozialer und gesellschaftlicher Aspekte

Gesellschaftliche und soziale Implikationen besitzen eine große Relevanz im Bereich der *Data Science* (vgl. Kapitel 1). Diskussionen über gesellschaftliche Aspekte im Informatikunterricht werden – wenn überhaupt – meist nur entkoppelt von technologischen Fragen der Unterrichtsinhalte geführt und sind nicht fest in die Arbeit in Softwareprojekten und Lernaufgaben integriert. Auch eines der wenigen Data Science-Curricula und Kompetenzmodell für das schulische Umfeld [GR] stellt die verschiedenen Aspekte von Data Science zwar umfassend, aber im Wesentlichen aus fachwissenschaftlicher Sicht dar. Daher stellt die Frage, wie diese Aspekte integraler Bestandteil des Unterrichtsmaterials und des Curriculums werden können, eine wichtige, schrittweise zu lösende Herausforderung dar.

2.3 Auf dem Weg zum Data Science-Kurs – Didaktische Ansätze

Eine wichtige Erkenntnis der Vorarbeiten war, dass – mehr als in der praktischen Informatik – eine gemeinsame Sicht auf Daten sowie die Entwicklung eines gemeinsamen Grundverständnisses aller Begriffe und Verfahren notwendig ist (vgl. [Tv09]) – ein auch während der ersten Durchführung des Projektkurses nicht abgeschlossener Schritt. Weiterhin wurden schon existierende Materialien und Curricula auf ihre Einsatzmöglichkeit hin evaluiert. Allerdings sind sämtliche, meist hochschulische Curricula gut strukturiert, aber nur auf technologische Aspekte fokussiert. Daher ergibt sich die Notwendigkeit, diese Materialien selbst zu entwickeln und die relevanten Inhalte aus den Beiträgen des in Kap. 1 erwähnten Symposiums zu generieren. Das Auffinden relevanter Information aus Data Mining und Statistik auf der einen Seite, sowie die notwendigen Kompetenzen aus dem Bereich des ML

⁶ PPDAC = Problem – Plan – Daten – Analyse – K(C)onclusion

sowie dem Design Künstlicher Neuronaler Netzwerke (KNN) auf der anderen Seite, ist dank der breiten Basis aller analysierten Unterlagen relativ einfach. Schwieriger gestaltet sich das Einbinden der gesellschaftlichen, sozialen und interdisziplinären Fragestellungen (vgl. Abschnitt 2.2). Daher entschieden wir uns, in diesem Projekt den Ansatz des „*Design-Based Research*“ (DBR) [Co03] zu verfolgen: Ausgehend von einem ersten Entwurf eines Data Science-Curriculums wird ein hierauf basierender Kurs entwickelt, durchgeführt und evaluiert. Aus den Erkenntnissen der Durchführung wird der Kurs und damit auch das anfangs noch sehr skizzenhafte Curriculum in mehreren Zyklen weiterentwickelt. Die im folgenden beschriebene Version des Data Science-Kurses wurde unter Verwendung dieses Ansatzes entwickelt und wird als sog. „Projektkurs“ in Kooperation mit einem Gymnasium vor Ort während des SJ 2018/19 erprobt und evaluiert.

3 Der Data Science-Kurs

Der so entstandene Kurs ist modular angelegt, wobei die beiden Bereiche „Data Mining und Statistik“ sowie „Künstliche Intelligenz und Maschinelles Lernen“ klar abgegrenzt sind und dem CRISP-DM-Modell (vgl. Abschnitt 2.1) folgend aufeinander aufbauend gestaltet werden. Zum Erwerben von Kompetenz, zumindest einfache Data Science Projekte selbst durchzuführen, wird als drittes Modul ein Projektmodul entwickelt und durchgeführt, so dass der Kurs in seiner ersten Version aus drei Modulen besteht:

1. *Von Daten zu Informationen*: Dieses Modul ist eine Einführung in Data Science und den Umgang mit Big Data und zielt darauf ab, das statistische Denken zu verbessern und Datenkompetenz zu entwickeln. Es werden dabei statistische Methoden auf Daten zur Informationsgewinnung angewendet, die Erkenntnisse reflektiert und ihre Aussagekraft kritisch diskutiert.

2. *Künstliche Intelligenz und Maschinelles Lernen*: Im Rahmen dieses Moduls lernen die Schülerinnen und Schüler zwei unterschiedliche Methoden kennen, Erkenntnisse aus Daten zu gewinnen, indem sie exemplarisch Entscheidungsbäume als Vertreter einer Symbolischen KI und KNN (hier Back-Propagation-Netze) als typischen Vertreter überwachter Lernens kennenlernen, analysieren und auf eigene Beispiele anwenden. Ziel ist, nicht nur die informatischen Aspekte von KI kennenzulernen, sondern auch ihre Erkenntnisse auf vorhandene Systeme anzuwenden und deren Grenzen, Chancen und Risiken zu diskutieren.

3. *Datenprojekte*: Im Rahmen der Durchführung von Datenprojekten können die Schülerinnen und Schüler ihre Kompetenzen einsetzen, um reale Fragestellungen zu bearbeiten. Dabei werden sie motiviert, ihr Vorgehen im Sinne des CRISP-DM-Modells zu planen, um so zu für sie optimalen Ergebnissen kommen zu können und diese auch am Ende zu präsentieren und deren gesellschaftlichen Implikationen diskutieren zu können.

Es wurde bei der Entwicklung und Umsetzung immer auf eine enge Verflechtung der fachlichen Inhalte mit gesellschaftlichen und sozialen Aspekten geachtet, um eine mehrdimensionale und interdisziplinäre Sicht auf alle Aspekte der Themen zu erhalten.

3.1 Modul 1: Von Daten zu Informationen – Datendetektive

Folgend dem in Abschnitt 2.1 beschriebenen CRISP-DM-Modell ist für ein Vorhaben aus dem Bereich Data Science zunächst wichtig, die Fragestellung zu verstehen („Business Understanding“), entweder sinnvolle Daten zu erheben oder schon gesammelte Daten zu verstehen („Data Understanding“), und diese anschließend zu analysieren und aufzubereiten („Data Preparation“). Inspiriert von diesem Zyklus wird in diesem Modul (insg. 7 Wo.) im ersten Baustein zunächst die Verwendung von großen und offen verfügbaren Datenmengen diskutiert und anschließend an Hand von „Lärmdata“ (vgl. Abb. 2) erprobt, derartige Daten selbst zu analysieren, unter Verwendung von Jupyter Notebooks⁷ aufzubereiten und zu visualisieren. Im zweiten Baustein führen die Schülerinnen und Schüler explorative Datenanalysen unter Verwendung des multivariaten JIM-Datensatzes mit Hilfe des Online-Tools CODAP⁸ durch und präsentieren und diskutieren ihre Erkenntnisse.

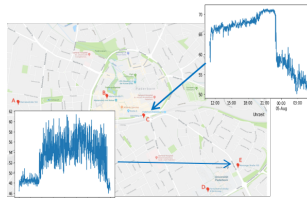


Abb. 2: Von den Schülerinnen und Schülern unter Verwendung von Python aufgearbeitete und verschiedenen Orten zugeordnete Lärmprofile

3.2 Modul 2: Künstliche Intelligenz und Maschinelles Lernen

Betrachtet man den CRISP-DM-Zyklus weiter, so folgt als nächster Schritt das Entwickeln eines Modells („Modeling“). Hier erkennen die Lernenden die Unterschiede zwischen klassischen algorithmischen Problemlösungsverfahren und datengetriebenen Prozessabläufen am Beispiel des ML (vgl. Abb. 3) und können die damit verbundene Rolle des Menschen innerhalb dieser Mensch-Maschine-Interaktion diskutieren und reflektieren (insg. 7 Wo.). Aus der Menge von Verfahren wurden im Vorfeld wichtige exemplarisch ausgewählt und in zwei Bausteinen mit den Schülerinnen und Schülern bearbeitet. Im ersten Baustein erwerben die Schülerinnen und Schüler unter Verwendung des „Sweet Learning Computers“ [Cu16b], einer Unplugged-Aktivität, ein grundsätzliches Verständnis von ML und diskutieren auf Basis dieses so erarbeiteten Wissens über aktuelle und zukünftige Chancen und Risiken dieser Technologien sowie ihren vielfältigen Einsatz. Dies wird vertieft durch die Einführung von *Entscheidungsbäumen*, die in relativ kurzer Zeit verstehbar sind und bei denen die wesentlichen Verfahren und Parameter zumindest im Grundsatz für die Lernenden transparent und erkennbar sind. Als Werkzeug wird hier wieder CODAP mit einem zusätzlichen Plug-In zum Darstellen von Entscheidungsbäumen

⁷ Jupyter Notebook = interaktive, browserbasierte Umgebung zur Programmierung; <https://jupyter.org/>

⁸ CODAP = Didaktisches Onlinetool zur Datenanalyse; <https://codap.concord.org>

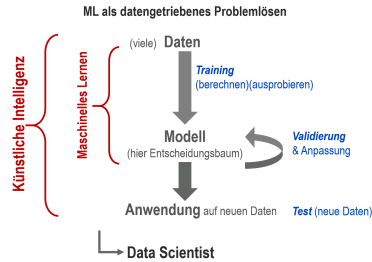


Abb. 3: Im Projektkurs erarbeitete Grafik zum Vorgehen zur Erstellung von KI-Modellen mittels ML

verwendet, das keine explizite Programmierung erfordert, sondern einen WYSIWYG-Editor bereitstellt. Allerdings stößt dieses Tool sehr bald an seine Grenzen. Insbesondere durch das manuelle Aufbauen des Entscheidungsbaumes kann kein tieferes Verständnis darüber erworben werden, wie Entscheidungsbäume algorithmisch erzeugt und zum automatischen Klassifizieren eingesetzt werden können. Daher werden im Anschluss mit Jupyter Notebook eigene Bäume zum schon bekannten JIM-Datensatz berechnet, wobei die Möglichkeit besteht, die Passung des entwickelten Baumes durch die Hinzunahme von Validierungsdaten („Evaluation“) zu überprüfen und anschließend zu optimieren. Während beim rein algorithmischen Problemlösen die Entwicklung eines stabilen und effizienten Algorithmus im Mittelpunkt stünde, ist hier die passende Wahl von Daten, Parametern und Optimierungsverfahren die größte Herausforderung für die Schülerinnen und Schüler, während die algorithmische Umsetzung der Bäume an sich durch die passende Wahl von Bibliotheken keine größere Herausforderung darstellt. Im folgenden zweiten Baustein werden zunächst (in Analogie zur Funktionsweise des Gehirns) durch die Unplugged-Aktivität „Brain in a Bag“ [Cu16a] die Grundbegriffe von *KNN* zusammen mit den Schülerinnen und Schülern erarbeitet. Anhand dieses durch weitere Erläuterungen und theoretische Inhalte ergänzte Spiel erkennen die Lernenden die grundlegenden Eigenschaften und Parameter von *KNN* ohne die Hürde eigener Programmierung. Um diese Erkenntnisse praktisch zu erproben und ein Gefühl für den Einfluss der verschiedenen Parameter zu erhalten, trainieren sie unter Verwendung des Online-Tools „Playground Tensorflow“⁹ unterschiedliche Netze und beobachten und erfassen dabei direkt die Auswirkungen ihrer Änderungen. Für ein tieferes Verständnis modellieren, trainieren und validieren sie selbst unter Verwendung von Jupyter Notebook eigene *KNN*. Hier kommt der freie Datensatz mit handschriftlichen Ziffern aus der MNIST-Datenbank [LCB98] zum Einsatz. Da die Ziffern der amerikanischen Schreibweise entsprechen, führt eine Validierung durch handgeschriebene Zahlen der Schülerinnen und Schüler zu schlechten Ergebnissen, so dass hier ein Anlass geschaffen wird, die Grenzen und Möglichkeiten von verschiedenen Ansätzen von ML zu diskutieren. Auch kommen hier tagesaktuelle Artikel und Berichte zum Einsatz, die auch problematische oder kontrovers diskutierte Einsatzgebiete von KI und *KNN* zum Thema haben, so dass hier verstärkt gesellschaftliche Fragestellungen bearbeitet werden.

⁹ Playground Tensorflow: Frei explorierbare Onlinevisualisierung von *KNN*; <https://playground.tensorflow.org>

3.3 Modul 3: Datenprojekte

Während in den ersten beiden Modulen die Erarbeitung neuen Wissens im Mittelpunkt stand, wird im dritten Modul ein gemeinsames Datenprojekt mit „realen“ Daten und Projektpartnern durchgeführt (insg. 10 Sitzungen). Dabei werden die in den vorherigen Modulen erworbenen Kompetenzen vertieft. Die Schülerinnen und Schüler erhalten den Projektauftrag, aus den Daten des örtlichen Parkleitsystems sowie aus den Bezahlsystemen eines Parkplatzes (Parkscheinautomat) und eines Parkhauses Vorhersagemodelle für die jeweilige Auslastung zu einem zukünftigen Zeitpunkt zu entwickeln und sich hierbei am Projektablauf des CRISP-DM zu orientieren. Zur Organisation und zur Unterstützung der arbeitsteiligen Arbeit an den Daten steht den Schülerinnen und Schülern ein Gitlab zur Verfügung, das sowohl zur Daten- als auch zur Aufgabenorganisation genutzt werden kann. Die betreuenden Personen fungieren im Rahmen dieses Projekts als Lernbegleiter und kümmern sich um den Kontakt mit den Projektpartnern sowie die interne Kommunikation und Organisation.

4 Erfahrungen der ersten Kursdurchführung

Der Projektkurs findet dreistündig mit 2 Schülerinnen und 17 Schülern der Jgst. 12, die alle Informatik belegt haben und daher über grundlegende Kenntnisse von Java verfügen, in einer Laborumgebung statt (insg. 24 Wo.). Die verwendete Programmiersprache Python sowie Jupyter Notebook waren für sie neu – der Einstieg war aber mit etwas Unterstützung gut zu bewältigen. Durch die Größe der Gruppe können die im Folgenden präsentierten Erkenntnisse nicht generalisiert werden, zeigen jedoch erste Hinweise, ob der Kurs an sich auch für andere Gruppen umsetzbar ist.

4.1 Erkenntnisse aus Modul 1: Daten und Informationen – Datendetektive

Das erste Modul dient zur Vermittlung von Kompetenzen zu statistischen Exploration, Verarbeitung und Darstellung von Daten (vgl. Abschnitt 3.1). Die Modulabschlusspräsentationen der Schülerinnen und Schüler zur statistischen Untersuchung von eigenen Fragestellungen anhand des JIM-Datensatzes zeigen ebenso wie verschiedene Diskussionsrunden, dass es ihnen gelungen ist, ein kritisches Verständnis zu Daten, Information und ihrer Visualisierung zu entwickeln. So wurde beispielsweise in den Präsentationen häufig die Größe des verwendeten JIM-Datensatzes kritisch gewürdigt und die damit verbundene geringe Aussagekraft der sich ergebenden kleinen Teilmengen eingeordnet. Im Rahmen der Feedbackrunde am Ende des Moduls wurde berichtet, dass einerseits die Auswertung der Lärmdaten mit Python spannender als das Arbeiten mit dem JIM-Datensatz angesehen wurde, aber beide Bausteine wurden von den Lernenden nicht als besonders persönlich relevant befunden. Daher ist zu überlegen, wie im nächsten Durchlauf die subjektiv empfundene Relevanz und damit Motivation noch erhöht werden können.

4.2 Erkenntnisse aus Modul 2: Künstliche Intelligenz und Maschinelles Lernen

In Modul 2 liegt der Schwerpunkt auf der Vermittlung der Grundideen von ML und KI sowie dem Erwerb der Kompetenz zum eigenständigen Programmieren von KI-Modellen (vgl. Abschnitt 3.2). Die beiden Unplugged-Einheiten haben sich als guter thematischer Einstieg in den jeweiligen Baustein erwiesen und waren für die Lernenden hilfreich zur Bildung eines ersten Verständnisses des Themenbereichs. Das Feedback zeigt, dass das angemessene schrittweise Anheben des Schwierigkeitsniveaus bei dem komplexen Themenfeld von ML und KI eine Herausforderung darstellt: Während der „Sweet Learning Computer“ durchweg positiv bewertet wurde, wurden die darauf folgenden Beispiele zum manuellen Erstellen von Entscheidungsbäumen mittels CODAP teils als monoton und wenig motivierend empfunden. Für die Erarbeitung der algorithmischen Darstellung von Entscheidungsbäumen wird der Begriff der Entropie nach Shannon benötigt. Die hierfür gestaltete Selbstlerneinheit mit den zugehörigen Aspekten der Informationstheorie in Jupyter Notebook hat sich als zu schwierig selbst für diese sehr leistungsstarke Lerngruppe erwiesen, so dass dies stattdessen gemeinsam im Plenum erarbeitet wurde. Beim Feedback zu diesem Baustein bemängelten die Schülerinnen und Schüler, dass – ebenso wie in Modul 1 – in diesem Baustein aus didaktischen Gründen nur kleine Datensätze mit nur begrenzter Aussagekraft zum Einsatz kamen. Die den zweiten Baustein eröffnende Unplugged-Einheit „Brain in a Bag“ sowie die im ersten Baustein erarbeitete Darstellung (vgl. Abb. 3) zum datengetriebenen Problemlösen ermöglichte den Lernenden bereits eine sachlich fundierte Diskussion zur gesellschaftlichen Bedeutung der Verwendung von KI-Systemen, in der Probleme wie Vorurteile in Trainingsdaten („Diskriminierende Algorithmen“) divers betrachtet und diskutiert wurden. Im weiteren Verlauf des Moduls erarbeiteten sich die Schülerinnen und Schüler das eigene Erstellen, Trainieren und Validieren von KNN unter Verwendung von Python und Jupyter Notebook mit dem MNIST-Datensatz zur Ziffernerkennung. Dass es den Schülerinnen und Schülern damit gelang, ein grundlegendes Verständnis zu ML und KI sowie der Programmierung eines passenden Netzes zu entwickeln, zeigte sich auch in der Modulabschlusspräsentation, in der sie für den schon bekannten JIM-Datensatz Modelle zur Vorhersage des Geschlechts der Studienteilnehmer implementierten und optimierten. Sie haben damit wichtige Kompetenzen entwickelt, die im anschließenden Projektmodul (vgl. Abschnitt 3.3) gefestigt und weiterentwickelt wurden.

4.3 Erkenntnisse aus Modul 3: Datenprojekt

Im Rahmen einer Kooperation mit zwei lokalen Unternehmen (RTB, Bad Lippspringe sowie ASP, Paderborn¹⁰) erhielten die Schülerinnen und Schüler den Auftrag, aus Daten des örtlichen Parkleitsystems sowie aus den Bezahlssystemen eines Parkplatzes (Parkscheinautomat) und eines Parkhauses Vorhersagemodelle für die jeweilige Auslastung zu einem zukünftigen Zeitpunkt zu entwickeln. Da sich die Daten von Parkhäusern und Parkplätzen

¹⁰ <https://www.rtb-bl.de> und <https://www.paderborn.de/microsite/asp/>

grundlegend unterscheiden, wurde die Aufgabe in zwei parallel bearbeitete Projekte aufgeteilt. Alle Daten lagen entweder als unbearbeitete Text-Dateien vor oder wurden direkt aus einer Datenschnittstelle entnommen. Die Schülerinnen und Schüler benötigten die ersten Wochen, um – folgend dem CRISP-DM-Prozess – die großen Datenmengen zu sichten, zu verstehen und aufzuarbeiten. Dazu wurden von den Gruppen Jupyter Notebooks entwickelt, die sie anschließend um die gewünschten Lerner zu erweiterten, wobei es den Gruppen freigestellt war, ob sie zur Modellierung Entscheidungsbaume oder KNN verwenden. Die größte Herausforderung stellte hier nicht die Implementierung der ML-Algorithmen dar, sondern die strategisch günstige Vorbereitung der Daten und die optimale Auswahl der Konfiguration des Lernalgorithmus. Bei der Abschlusspräsentation konnten beide Gruppen aber nicht nur funktionierende Vorhersagemodelle einschließlich einer sinnvollen Web-GUI für ihre jeweiligen Parkmöglichkeiten an die Projektpartner übergeben, sondern überzeugten auch in der abschließenden Diskussion mit ihrem grundlegenden Verständnis nicht nur über fachliche Problemstellungen, sondern auch durch ihre Fähigkeit, die hierzu gehörenden gesellschaftlichen Fragen fundiert erörtern können.

5 Ausblick und Fazit

Im Rahmen dieses Artikels beschreiben wir die Entwicklung eines Data Science-Kurses, der in der Sek II als Projektkurs unterrichtet werden kann. Nicht jede Lehrkraft, die gerne dieses Thema in den Unterricht einbinden möchte, hat diese Zeitressource. Daher planen wir, die Module auf Basis unserer bisherigen Erfahrungen weiterzuentwickeln, so dass es möglich wird, nur ausgewählte Teile durchzuführen. Im Moment werden die Erkenntnisse der aktuellen Durchführung sowie der Erprobungen einzelner Komponenten in die Kursmaterialien eingearbeitet. Im Anschluss daran werden die Materialien interessierten Lehrkräften unterschiedlicher Schularten zur Erprobung und Weiterentwicklung zur Verfügung gestellt. Zur Weiterentwicklung des Gesamtkurses im Sinne des DBR wird dieser im nächsten Schuljahr mit der bisherigen Partnerschule in modifizierter Form durchgeführt. Da die Schülerinnen und Schüler einerseits das durch die Modulstruktur bedingte „Vorratslernen“ bemängelten und andererseits an einigen Stellen Schwierigkeiten hatten, die erlernten Inhalte zielgerichtet auf die Projektfragestellungen anzuwenden, ist geplant, die Theoriephasen so mit der Arbeit an einem Datenprojekt zu verzahnen, dass die theoretischen Inhalte gezielt während des Projektablaufs eingebaut werden. Zudem werden die in den verschiedenen Modulen enthaltenen gesellschaftlichen, sozialen und ethischen Fragestellungen nochmals gebündelt in einem zusätzlichen Modul diskutiert und bearbeitet, um die Schülerinnen und Schüler noch tiefer mit den darin enthaltenen Fragestellungen vertraut zu machen. Dies alles erfordert noch einiges an konzeptioneller Arbeit, die sich jedoch durch einen höheren Kompetenzgewinn seitens der Lernenden bemerkbar machen wird. Zusammengefasst sind wir trotz aller noch zu leistenden Entwicklungsarbeit überzeugt, durch den vorgestellten *Data Science-Kurs* einen großen Schritt hin zu einem umfassenden Curriculum für diesen sehr komplexen Bereich getan zu haben, so dass wir gespannt auf die nächsten Durchführungen des Kurses und seiner Module sind.

Literaturverzeichnis

- [Ch00] Chapman, P.; Clinton, J.; Kerber, R.; Khabaza, T.; Reinartz, T.; Shearer, C.; Wirth, R.: Cross Industry Standard Process for Data Mining 1.0, Step-by-step Data Mining Guide. 2000.
- [CM97] Cobb, George W.; Moore, David S.: Mathematics, statistics, and teaching. *The American Mathematical Monthly*, 104(9):801–823, 1997.
- [Co03] Cobb, Paul; Confrey, Jere; diSessa, Andrea; Lehrer, Richard; Schauble, Leona: Design Experiments in Educational Research. *Educational Researcher*, 32(1):9–13, 2003.
- [Cu16a] Curzon, Paul: , Brain in a Bag, A CS4FN Computing Activity. https://www.youtube.com/watch?v=1ux_ybamClU, 2016. Accessed: 2018-06-15.
- [Cu16b] Curzon, Paul: , The Sweet Learning Computer, A CS4FN Computing Activity. www.cs4fn.org/machinelearning, 2016. Accessed: 2019-02-10.
- [GR] Grillenberger, Andreas; Romeike, Ralf: In: Proceedings of the 17th Koli Calling Conference on Computing Education Research. New York.
- [LCB98] LeCun, Yann; Cortes, Corinna; Burges, Christopher J.C.: , MNIST handwritten digit database. <http://yann.lecun.com/exdb/mnist/>, 1998. Accessed: 2018-06-15.
- [Pa18] Paderborn Symposium on Data Science Education at School Level 2017: The Collected Extended Abstracts, 2018.
- [Ri15] Ridsdale, Chantel; Rothwell, James; Smit, Michael; Ali-Hassan, Hossam; Bliemel, Michael; Irvine, Dean; Kelley, Daniel; Matwin, Stan; Wuetherick, Bradley: Strategies and best practices for data literacy education: knowledge synthesis report. 2015.
- [Ri16] Ridgway, Jim: Implications of the Data Revolution for Statistics Education. *International Statistical Review*, 84(3):528–549, 2016.
- [Rö16] Röhner, Gerhard; Brinda, Torsten; Denke, Volker; Hellmig, Lutz; Heußer, Theo; Pasternak, Arno; Schwill, Andreas; Seiffert, Monika: Bildungsstandards Informatik für die Sekundarstufe II. Beilage zu LOG IN, Heft, (183/184), 2016.
- [SBS18] Sentance, Sue; Barendsen, Erik; Schulte, Carsten: *Computer Science Educ.: Perspectives on Teaching + Learning in School*. Bloomsbury Academic, London, 3 2018.
- [TD16] Tedre, Matti; Denning, Peter J.: The Long Quest for Computational Thinking. In: Proceedings of the 16th Koli Calling International Conference on Computing Education Research. Koli Calling '16, ACM, New York, NY, USA, S. 120–129, 2016.
- [Tv09] Thijs, Annette; van den Akker, Jan: Curriculum in development. SLO - Netherlands Institute for curriculum development, 2009.

Blöcke, Blumen, Mikrocontroller und das Internet of Things

Ein Konzept zum kontextorientierten Einsatz von Parametric Design und Physical Computing in Mädchen-Förderworkshops

Nils Pancratz¹, Anatolij Fandrich¹, Christos Chytas¹, Mareike Daeglau¹, Ira Diethelm¹

Abstract: Die Art und Weise, mit der Informatik über die letzten Jahrzehnte von der Industrie, den Medien und teils sogar (öffentlichen) Bildungseinrichtungen vermarktet wurde, führte in westlichen Kulturen zu einer deutlichen Geschlechterkluft in informatischen Berufen wie auch Studiengängen. Studien zu diesem aus diversen Gründen anzugehenden Missstand lieferten in den letzten Jahren gehäuft Belege dafür, dass sich Mädchen und junge Frauen eher für Informatik begeistern lassen, wenn ihnen diese faszinierende Wissenschaft in Kreativität, Kollaboration und Kommunikation fordernden und fördernden Kontexten präsentiert wird. So führten diese Erkenntnisse zuletzt vermehrt zu Bemühungen, Mädchen über spezielle außerschulische Workshopformate anzusprechen. Um in solchen Formaten Programmieranfängerinnen und -anfängern zugleich grundlegende Programmierkenntnisse (bspw. über Schleifen, Variablen, Verzweigungen und Methodenaufrufe) zu vermitteln und trotzdem schnell motivierende Ergebnisse zu erzielen, wird zuletzt vermehrt auf das Konzept des *Computational Design* zurückgegriffen. Dies ermöglicht über die Idee der *Digital Fabrication* ebenso wie das in außerschulischen Workshops bereits etablierte *Physical Computing*, Informatik im wahrsten Sinne des Wortes begreifbar zu machen. Dieser Praxisbeitrag stellt ein Workshopkonzept vor, in dem über den Kontext des *Internet of Things* die Ideen hinter *Parametric Design* und *Physical Computing* miteinander kombiniert werden – und dabei beide Geschlechter gleichermaßen ansprechen kann und soll! Das Ziel ist, am Ende des Workshops, der bereits mehrmals erfolgreich sowohl in außerschulischen wie auch schulischen Formaten durchgeführt wurde, eine selbstgestaltete Lampe – bspw. in Blumenform – fabriziert zu haben, die in Abhängigkeit der über das Internet bezogenen Wetterdaten in eigens kreierten Lichteffekten strahlt.

Keywords: Gendersensitive Informatikdidaktik; Parametric Design; Physical Computing; Internet of Things; Informatik im Kontext

1 Einleitung

„Informatik für alle“ – bei einem Blick auf aktuelle Statistiken zu Studierendenzahlen wirkt der Titel dieses Tagungsbandes nahezu utopisch und unerreichbar: Schließlich war der ohnehin schon beklagenswert niedrige Frauen-Anteil unter denjenigen, die sich deutschlandweit für eine Neueinschreibung in einen Informatikstudiengang entschieden haben, zuletzt wieder rückläufig und lag zu Beginn des Wintersemesters 2018/2019 bei nur

¹ Carl von Ossietzky Universität Oldenburg, Didaktik der Informatik, Uhlhornsweg 84, 26129 Oldenburg, vorname.nachname@uni-oldenburg.de

22,5%². Die wenigen Frauen, die sich für ein Informatikstudium entscheiden, wählen dabei häufig „Bindestrich-Studiengänge“ wie bspw. Bioinformatik, Medizinische Informatik oder auch Medieninformatik [K116], was als Indiz dafür gelten könnte, dass ein stärkerer Kontextbezug (vgl. Abs. 2.2) besonders förderlich für das Interesse und die Begeisterung von Mädchen und jungen Frauen an der Informatik ist. Auch bei den geschlossenen Ausbildungsverträgen liegt der Frauenanteil je nach Ausbildungsberuf aktuell nur zwischen knapp 5% (Fachinformatiker/-in Systemintegration) und 15% (Informatikkaufleute)³.

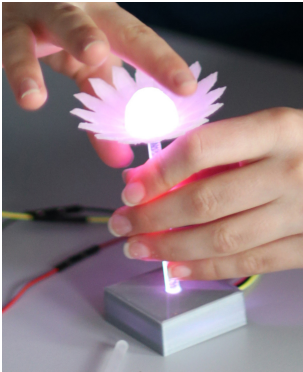


Abb. 1: Die Teilnehmerin eines dieser Workshops beim Zusammenstecken der Komponenten ihrer eigenständig gestalteten internetfähigen Lampe.

Der Nährboden für diese Geschlechterkluft ist zweifelsfrei bereits in der Schulinformatik zu suchen: In den zahlreichen Bundesländern, in denen Informatik ein Schattendasein als Wahlpflichtfach fristet, entscheiden sich erfahrungsgemäß nur wenige Mädchen, es zu belegen; in den Ländern mit verpflichtendem Informatikunterricht (IU) hingegen sind es im Vergleich zu ihren Geschlechtergenossen überproportional viele junge Frauen, die sich im Laufe ihrer Schulzeit gegen eine weitere Belegung entscheiden. So ruft diese offensichtliche Geschlechterkluft seit Jahren zahlreiche Studien zu möglichen Ursachen (vgl. Abs. 2.1) auf die Programme (inter-)nationaler Informatikdidaktik-Tagungen. Der vorherrschende Fachkräfte-Mangel im IT-Sektor ist hierzu sicherlich genauso als Ansporn zu begreifen wie der Anspruch, *allen* Schülerinnen und Schülern (SuS) die Vielfalt informatischer Bildung zu bieten. Da nicht nur im schulischen Informatikunterricht, sondern auch in entsprechenden außerschulischen Lernangeboten vorwiegend Jungen und junge Männer anzu-

treffen sind, wird zuletzt vermehrt auf außerschulische Formate gesetzt, die gezielt Mädchen ansprechen und ihnen so die klischeebehaftete Informatik aus einer anderen Perspektive aufzeigen sollen, um langfristig mehr Schülerinnen an die faszinierende Wissenschaft der Informatik heranzuführen. In diese Angebote reiht sich auch dieser Workshop ein, wenngleich die von den Autorinnen und Autoren dieses Beitrags umgesetzten Ideen ihrer Ansicht nach ausdrücklich für beide Geschlechter gleichermaßen förderlich sind. So berücksichtigt das in diesem Praxisbeitrag vorgestellte Workshopkonzept gezielt grundlegende Ansprüche an eine gendersensitive Informatikdidaktik (Abs. 2.1) und bedient sich innerhalb des Kontextes des *Internet of Things* (Abs. 2.2) den Ideen hinter *Digital Fabrication* (Abs. 2.3) und *Physical Computing* (Abs. 2.4). Am Ende des (in Abs. 3 beschriebenen) Workshopablaufs steht dabei eine internetfähige, selbstgestaltete und mithilfe von 3D-Druckern konstruierte Lampe (Abb. 1), die in Abhängigkeit der jeweiligen Wetterlage, die von dem verwendeten Mikrocontroller über einen Webservice abgerufen wird (Abs. 3), in unterschiedlichen Lichteffekten leuchtet.

² Quelle: Statistisches Bundesamt: „Bildung und Kultur (2018): Schnellmeldungsergebnisse der Hochschulstatistik zu Studierenden und Studienanfänger/-innen (vorläufige Ergebnisse) - Wintersemester 2018/2019“

³ Quelle: Deutscher Industrie- und Handelskammertag (2017): „Statistik Ausbildung 2017“

2 Didaktische Relevanz des Workshopkonzepts

Im Folgenden sollen die didaktischen Überlegungen, die bei der Konzeption dieses Workshops (WS) zum Tragen kamen, von den grundlegenden Aspekten gendersensitiver Informatikdidaktik (Abs. 2.1⁴) und Kontextorientierung (Abs. 2.2) über die gewählten Paradigmen des *Parametric Design* (Abs. 2.3) und *Physical Computing* (Abs. 2.4) dargelegt werden.

2.1 Aspekte gendersensitiver Informatikdidaktik

Auf der Suche nach Ursachen für die unverkennbare Geschlechterkluft in der Informatik (vgl. Abs. 1) begnügen sich nur wenige Studien mit der platten Erkenntnis, dass Mädchen und junge Frauen schlichtweg weniger Interesse an entsprechenden Themen hätten [An08, SM11]. Tiefgreifendere Untersuchungen hingegen kommen zu dem Ergebnis, dass sie die Informatik als attraktiveres Umfeld betrachten, sobald

- die behandelten Technologien in *soziale Kontexte* eingebettet sind, in denen sie selbst involviert sind bzw. sich selbst wiedersehen können [FMM97],
- Informationstechnologien verwendet werden, um *Menschen* in ihren *alltäglichen Situationen* zu unterstützen [ZS08],
- sie zu einem *übergeordneten Ziel* eingesetzt wird [FM02] und
- Informatik als *kreativer* [DKW14] und *kollaborativer* [ZS08, Eg16, MNS08] Prozess aufgezeigt wird, in dem den Mädchen die Möglichkeit gegeben wird, *sich miteinander auszutauschen* und *zusammenzuarbeiten* [ZS08].

Sowohl *schulische* als auch *außerschulische* Lernumgebungen bieten ausreichend Spielraum, diesen „besonderen“ Anforderungen an eine gendersensitive Informatikdidaktik gerecht zu werden und so situatives Interesse zu wecken, das sich bestenfalls zu langfristig anhaltender Motivation entwickeln kann [Ma17]. Vor allem letztere bieten die Möglichkeit, den zahlreichen vorherrschenden Klischees und Stereotypen, die noch immer über den Berufsalltag von Informatikern⁵ existieren, entgegenzuwirken. Doch dazu sollten sich diese Angebote explizit und ausschließlich an Mädchen und junge Frauen richten, denn außerschulische Angebote rund um die Informatik werden ansonsten leider immer noch grundsätzlich von Jungen wahrgenommen [SM11, Wi16]. So richtet sich auch das BMBF-geförderte Projekt *smile* (<https://www.smile-smart-it.de>), im Rahmen dessen dieses WS-Konzept entwickelt und durchgeführt wurde und wird, über ein Angebot an (außerschulischen) WS explizit an Mädchen und junge Frauen, um das übergeordnete Ziel zu verfolgen, langfristig und nachhaltig den Anteil an Frauen in informatischen Studiengängen und Berufen zu erhöhen.

⁴ Ausschnitte aus diesem Abs. 2.1 wurden bereits in [PD18] veröffentlicht.

⁵ Hier wird bewusst nur die männliche Form verwendet. Schließlich ist das Bild eines typischen Informatikers männlich, unattraktiv und sozial wenig kompetent [Ch13].

2.2 Informatik im Kontext und das „Internet of Things“

Im *Internet of Things (IoT)* unterstützen immer kleiner werdende Computer, die in immer mehr alltägliche Gegenstände wie Lampen, Kühlschränke, Rasierapparate, Staubsauger oder Autos eingebettet werden, den Menschen in unterschiedlichsten Situationen. Um genau diese „Bewältigung lebensweltlicher Herausforderungen in Verbindung mit Informatiksystemen“ [DKW11, 97] geht es in dem Konzept *Informatik im Kontext (IniK)* zur Planung, Durchführung und Auswertung von IU, in dem über eine Ausrichtung an der Lebenswelt der SuS eine zusätzliche Motivation verfolgt wird. Das IoT erfüllt zweifelsohne die von [DKW11] gestellten Anforderungen an einen guten Kontext im Sinne einer IniK-Reihe. So bewegen sich die Schülerinnen in dem in diesem Beitrag vorgestellten WS im Rahmen dieses Kontextes – wenn auch nicht stringent in den einzelnen Phasen, für die [DKW11] plädieren. Sie begreifen die Technologien, an die sie herangeführt werden, so im Sinne einer Gendersensitivität (vgl. Abs. 2.1) in soziale, lebensweltliche Kontexte eingebettet.

2.3 Parametric Design

Computational Design (CD) ist ein Prozess, der Merkmale der Programmierung zur Erzeugung und Modifikation von 2D- und 3D-Visualisierungen beinhaltet. Ein typisches Konstruktionsparadigma im Sinne von CD ist *Parametric Design (PD)*, das oft mit der Architektur und der Verwendung von Parametern zur Steuerung der Eigenschaften von 2D- oder 3D-Modellen zusammenhängt. Obwohl es einen anhaltenden Trend zum Einsatz von *Digital Fabrication (DF)* und insbesondere 3D-Druck in Bildungskontexten gibt, ist ihr Potenzial für den Anfangsprogrammierunterricht für viele Lehrerinnen und Lehrer immer noch etwas Neues. Schließlich lassen sich sowohl schulische als auch außerschulische programmierlastige Lernaktivitäten über CD und DF auflockern und so Lernprozesse über das Herstellen von eigens kreierte Artefakten unterstützen [CDT18]. Zusätzlich schärft die Verbindung der Gestaltung von 2D- und 3D-Formen mit grundlegenden Programmierkonzepten wie Schleifen, (konditionalen) Verzweigungen und Methodenaufrufen Fähigkeiten des *Computational Thinking* wie Zerlegung und Mustererkennung [CTD19]. Parametrische Entwurfswerkzeuge gehen also deutlich über die Möglichkeiten von nicht programmierbaren CAD-Werkzeugen hinaus und ermöglichen die Erstellung komplexer Formen und Geometrien wie Fraktale oder Bäume mit Zweigen, die rekursive oder repetitive Elemente beinhalten können [CDT18]. CD findet vereinzelt bereits Einsatz in Lernaktivitäten: So konzentrieren sich bspw. [DKW14] und [JB13] auf die Verwendung der Programmiersprache *Processing* zur Erzeugung von 2D-Formen, die mit Lasercuttern hergestellt werden können, und ihre Auswirkungen; in dem Ansatz von [KKR17] werden *Turtle Graphics* erzeugt und mithilfe von 3D-Druckern als motivierende Unterstützung genutzt. Der Einsatz von PD-Tools wie *BlocksCAD* (vgl. Abs. 3) ist bislang jedoch noch deutlich seltener. So darf und soll dieser Praxisbeitrag auch als Plädoyer für einen verstärkten Einsatz von PD verstanden werden.

2.4 Physical Computing

Eingebettete Systeme bieten durch das Zusammenspiel von Hardware- und Softwaresystemen und den damit verbundenen Möglichkeiten zur Interaktion zwischen Mensch und Maschine eine Vielzahl von Chancen, Informatik in einem praktischen Kontext zu vermitteln. *Physical Computing (PC)* beschreibt dabei den Prozess, ein solches System zu erschaffen. Neben den bekannten, positiven Effekten der Kontextualisierung und dem Einsatz von Fallstudien in Ingenieurwissenschaften [YAM19], wirkt sich auch die Entwicklung solcher physischer Systeme positiv auf Lerngruppen aus. Sentence et al. fassen diese Effekte folgendermaßen zusammen: Die Motivation steigt, da die Lernergebnisse nicht virtuell existieren, sondern in Form eines praktischen Produktes sichtbar werden. Zudem hilft die Greifbarkeit dieser Systeme den Lerngruppen beim Debuggen und Optimieren, Programmierkonzepte besser zu verstehen. Die Arbeit lässt sich außerdem sinnvoll auf mehrere Personen aufteilen, z. B. kümmert sich eine Person um Schnittstellen und Algorithmen, während eine andere ein Gehäuse zum 3D-Druck modelliert. So wird das Arbeiten im Team geschult. Zudem bietet das PC die Möglichkeit, sich kreativ zu entfalten [Se17]. Speziell Mädchen profitieren von dem Einsatz solcher Systeme im schulischen Kontext. So gaben Mädchen, die sich mit PC beschäftigt haben, an, dass sie Spaß am Programmieren gefunden haben und sich außerdem selbstbewusster fühlen [Se17].

3 Workshoptools und -ablauf

Um eine günstige IoT-Anwendung zu entwickeln, wurde das Entwicklungsboard *D1 mini* vom Hersteller *WEMOS Electronics* gewählt. Dieses basiert auf den 32-Bit Mikrocontroller *ESP8266EX* von *Espressif* und zeichnet sich unter anderem durch die native Unterstützung der 801.11 b/g/n Funknetzwerkstandards und der Netzwerkprotokolle TCP, UDP und HTTP aus. Darüber hinaus bietet der Hersteller zahlreiche pin-kompatible Erweiterungen für die Boards der D1-Reihe an, welche unkompliziert auf das Entwicklungsboard gesteckt werden können. Dadurch entfallen potentielle Fehlerquellen beim Anschließen neuer Sensoren bzw. Aktoren durch Wackelkontakte oder fehlerhafte Verkabelungen. Um die über das Internet gewonnene (s. u.) Außentemperatur auch optisch anzuzeigen, wurde ein 0.66 Zoll großes OLED-Display mit einer Auflösung von 64x48 Pixel, ebenfalls vom Hersteller *WEMOS*, verwendet. Nachdem man zwei Reihen von Stiftleisten an die OLED-Display Erweiterung gelötet hat, kann man dieses auf das Entwicklungsboard stecken. Zusätzlich wurde zur visuellen Darstellung des aktuellen Wetters eine WS2812B RGB-LED verwendet. Der Vorteil gegenüber konventionellen RGB-LEDs, welche drei Kathoden bzw. Anoden zur Darstellung der verschiedenen Farben benötigen, ist die flexible Programmierbarkeit. Während zur Darstellung der Farben an drei Pins verschiedene Spannungspegel mittels PWM gesetzt werden müssten, kann die WS2812B LED die Farben mit Hilfe der Arduino kompatiblen FastLED Bibliothek über eine einzige Datenleitung und in verschiedenen Farbräumen darstellen. Dies erleichtert auch die Erstellung von eigenen Animationen deutlich. Um die LED mit dem Entwicklungsboard zu verbinden,

wurde ein Steckbrett verwendet. Die aktuellen Wetterdaten werden über die kostenfreie *Openweathermap* (<https://openweathermap.org/>) API aus dem Internet bezogen. Mit Hilfe der HTTP-Methode GET wird das Wetter für den entsprechenden Standort angefragt und die Antwort des Servers vom Mikrocontroller empfangen, geparkt und anschließend mit einer vorher festgelegten Farbe oder Animation dargestellt. Ein Plexiglasstab, welcher mittels einer Halterung aus einem 3D-Drucker unmittelbar über der LED befestigt ist, leitet das Licht in eine von den TeilnehmerInnen selbst modellierten Blume (siehe Abb. 1), welche ebenfalls mit einem 3D-Drucker gefertigt wurde. In beiden Fällen wurde PLA als Druckmaterial verwendet. Die Firmware für den Mikrocontroller wurde innerhalb der *ArduinoIDE* (<https://www.arduino.cc/en/main/software>) geschrieben. Zusätzlich haben die WorkshopeteilnehmerInnen ihre Animationen mit dem blockbasierten Werkzeug *mBlock3* (<http://www.mblock.cc/mblock-software/>) erstellt. Die darin verwendeten Blöcke wurden eigens für den Workshop entwickelt und zur Verfügung gestellt. Die Modellierung der Blumen erfolgte parametrisch mit dem Tool *BlocksCAD* (<https://www.blocks3d.com/>), welches spielerisch Programmierkonzepte durch die Visualisierung von Formen vermittelt.

Der **Ablauf** des in diesem Praxisbeitrag vorgestellten WS^{6,7} gliedert sich in drei Teile: An den ProgrammierEinstieg über PD (Teil 1) schließt sich die Phase zum PC (Teil 2) an, bevor die einzelnen Komponenten abschließend zur fertigen Blume zusammengesteckt werden und eine Präsentation ihres Entstehungsprozess vorbereitet wird (Teil 3). In Abhängigkeit der Vorerfahrung der SuS bzw. des gewünschten Grads an Anleitung lässt sich die Gestaltung der Teile 1 und 2 bzgl. der einzuplanenden Zeit anpassen (vgl. Abb. 2), sodass sich der WS insgesamt auf 18 bis 30 (Zeit-)Stunden beläuft und so sowohl für den außerschulischen (bspw. für die Durchführung in einwöchigen WS oder Projektwochen) als auch schulischen (bspw. als Unterrichtsreihe zum Programmier-Anfangsunterricht) Einsatz eignet.

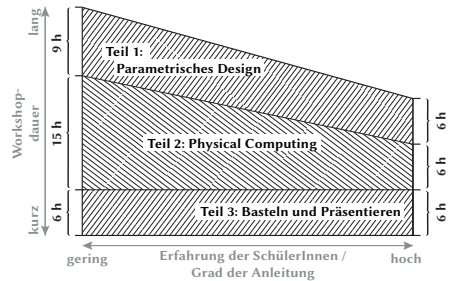


Abb. 2: Überblick über das zeitliche Verhältnis der einzelnen WS-Abschnitte

Im **ersten Teil** des WS lernen die Teilnehmenden nach einer Einführung in die Oberfläche des verwendeten Tools *BlocksCAD* (vgl. Abs. 3), wie man grundlegende geometrische Formen (Würfel, Kugeln und Zylinder) über *Parameter* oder ein Anwenden von *Operationen* wie bspw. Skalierungen anpassen kann. Im Anschluss daran begreifen sie, dass sich diese Grundkörper durch *Mengenoperationen* wie bspw. Vereinigungen oder Differenzen zu nahezu jeder vorstellbaren Gestalt formen lassen. Über eine Verkettung mehrerer solcher Operationen wird dann mit einem Arbeitsblatt dazu angeleitet, die Grundform eines

⁶ (Stunden-)Verlaufspläne, Arbeitsblätter, verwendete Libraries und Code-Beispiele finden sich im Git-Repository des Projekts (<https://github.com/projekt-smile/bloeccke-blumen-mikrocontroller-und-das-internet-of-things>) oder über den Download-Bereich der Projekt-Webseite (<https://www.smile-smart-it.de>).

⁷ Dieser WS wurde als Anschluss an den in [PD18] beschriebenen Einstieg in Sensorik und Aktorik mithilfe von *littleBits* konzipiert und auch mehrmals im Anschluss daran durchgeführt.

Blattes zu modellieren sowie den entstehenden *Code* über die Deklaration einer *Methode* auszulagern und so an mehreren Stellen des *Programms* aufrufbar zu machen (vgl. Abb. 3, *links*). Anschließend lernen die SuS das Prinzip hinter *Schleifen* kennen und können so bspw. Blüten statt über repetitive Methodenaufrufe durch Verwendung eines einzigen zusätzlichen Blocks aus den zuvor modellierten Blättern kreieren (vgl. Abb. 3, *rechts*). Über die zusätzliche Ergänzung einer *Verzweigung* (falls *i* ist gerade) lässt sich optional an dieser Stelle ein weiteres Programmierkonzept einführen und so bspw. jedes zweite Blatt in der Reihe länglicher gestalten. Um die Blume 3D-druckbar zu gestalten, muss sie nach einer individuellen Anpassung (optional nach einer Online-Recherche) durch die S(uS) mit Hilfe der Differenz-Funktion von unten abgeflacht werden. Außerdem sollte eine Knospe, die von einem Zylinder ausgepart ist und in die sich so später ein Acrylglasstab stecken lässt (vgl. Abb. 1), ergänzt werden. Nach dieser (verhältnismäßig) angeleiteten Einstiegsphase lässt sich optionalerweise der Arbeitsauftrag geben, weitere parametrische Designs zu recherchieren und mit *BlocksCAD* zu modellieren, um sie anschließend für die personalisierten Lampen zu verwenden. Für den gesamten Prozess des Gestaltens der Blume bzw. einer anderen beliebigen Form empfehlen die Autorinnen und Autoren dieses Beitrags dabei einen zeitlichen Rahmen von 6 bis 9 Zeitstunden (s. o.). Selbstverständlich lässt sich dieser Teil des WS dabei auch isoliert von dem restlichen WS durchführen und so ein kreativer Einstieg in die Vermittlung von Programmiergrundlagen finden.

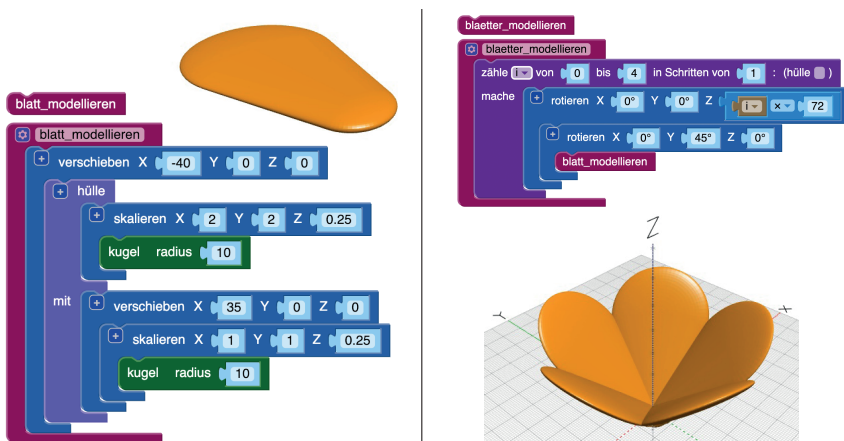


Abb. 3: Methode (*links unten*) zur Modellierung eines Blattes (*links oben*) und Methode (*rechts oben*) zur Modellierung einer Blüte (*rechts unten*) über die Verwendung einer Schleife in *BlocksCAD*

Der **zweite Teil** des WS behandelt die Programmierung des Mikrocontrollers. Dazu erschließen sich die Teilnehmenden nach einem Einstieg in die verwendete blockbasierte Programmierumgebung *mBlock* (vgl. Abs. 3) zunächst aus den ihnen aus *BlocksCAD* bekannten Schleifen die Bedeutung der loop-Funktion in Programmen, indem sie die onboard LED des verwendeten Mikrocontrollers (vgl. Abs. 3) über die *delay*-Funktion in einem beliebigen Intervall zum Blinken bringen. Der Bogen zum textuellen Programmieren wird kontinuierlich dadurch gezogen, dass *mBlock* den blockbasiert einzugebenden Code

on the fly als textuelle Beschreibung in Arduino-Code bereitstellt. Das Anschließen und Ansteuern des OLED-Displays⁸ (vgl. Abs. 3) stellt den nächsten Abschnitt im WS dar: So sollen die S(uS) zunächst ihren Namen auf dem Display anzeigen lassen, bevor sie (optional) kleine Bitmap Grafiken erstellen können, die in Hinblick auf das übergeordnete Ziel, eine smarte Wetterlampe zu kreieren, verschiedene Wetterlagen (Sonne, Wolken, Schnee...) symbolisieren. Dieses übergeordnete Ziel findet sich auch im nächsten Abschnitt zur Gestaltung von Lichteffekten (bspw. in Farbübergängen von Gelb- zu Rottönen für *sonnig* oder Farbverläufe in Blautönen mit kurzen gelben *Blitzen* bei *Gewitter*) wieder, in denen die anzuschließende externe RGB-LED leuchten soll⁹. Dazu bietet es sich an, die Gruppe in Zweier- oder Dreierteams aufzuteilen und jeder Gruppe einen Effekt zuzuordnen. So entsteht gleichzeitig der Nebeneffekt, dass die S(uS) (Software-)Entwicklung als kollaborativen Prozess (vgl. Abs. 2.1 und 4) begreifen, indem alle etwas zum Endprodukt beitragen. Auch dieser zweite Teil des WS lässt sich zeitlich variieren: Die Autorinnen und Autoren veranschlagen je nach Vorerfahrung der Teilnehmenden, Grad der Anleitung und je nachdem, ob das OLED-Display behandelt wird oder nicht, für diesen Teil 6 bis 15 Zeitstunden.

Während die vorangegangenen WS-Abschnitte hinsichtlich ihrer Länge relativ variabel sind, hat die Erfahrung gezeigt (vgl. Abs. 4), dass der **dritte Teil**, in dem die Schülerinnen ihre Ergebnisse präsentieren sollen, vergleichsweise konstant 6 Zeitstunden beansprucht. Die Präsentation sollte dabei den Anspruch erfüllen, multimedial (ob in Form eines VLOGs, eines kurzen „Werbe“-Videos o. Ä.) und ansprechend gestaltet zu sein. Projektseitig (vgl. Abs. 2.1) wird über diesen Teil das Ziel verfolgt, die Artefakte der WS-Teilnehmerinnen auf Veranstaltungen und Social-Media-Kanälen adressatengerecht zu präsentieren, um so über einen Multiplikatoreffekt weitere WS-Teilnehmerinnen zu akquirieren.

4 Darstellung der Erfahrungen, Reflexion und mögliche Abweichungen

Wie bereits erwähnt, steht im Fokus dieses Workshops, die Zielgruppe junger Mädchen und Frauen für informatische Inhalte nicht nur temporär zu begeistern, sondern auch nachhaltiges Interesse zu festigen. Während der zweite Aspekt zu diesem Zeitpunkt noch nicht hinreichend beantwortet werden kann, soll abschließend eine erste Evaluation hinsichtlich der motivierend gestalteten Inhalte und insbesondere deren ansprechender Vermittlung gegeben werden. Ein Kernanliegen dieses Konzepts ist es, die Artefakte möglichst preisgünstig zu realisieren, damit die Teilnehmerinnen diese nach Ende des Workshops ohne Zuzahlung mit nach Hause nehmen können. Hintergrund dafür ist die Annahme, dass die Schülerinnen sich stärker mit ihren Projekten identifizieren und somit das persönliche Engagement adressiert wird. Die Wetterblume soll hierbei einen kreativ gestalteten Alltagsgegenstand darstellen, der zudem noch den praktischen Anwendungsfall der Wettervorhersage miteinbezieht. Ein weiterer Effekt der Artefaktbildung ist die Möglichkeit, informatische Inhalte eng an kreative

⁸ Um die Kosten der einzelnen Lampen zu reduzieren, lässt sich alternativ auch auf das Display verzichten.

⁹ *mBlock-Libraries*, die die Ansteuerung der in Abs. 3 beschriebenen Hardwarekomponenten ermöglichen und u. a. die Effektgestaltung stark vereinfachen, finden sich in dem Git-Repository des Projekts (s. Fußnote 6).

Prozesse zu binden und somit den Einstieg in diese zu erleichtern. Das blockbasierte Design von *BlocksCAD* führt die Schülerinnen bereits an die spätere Programmierung heran und die Funktionsweise der Blöcke wird direkt sichtbar im Editor. In einer früheren Version des Workshopangebots wurde die Programmierung der Lichteffekte textuell mit der *Arduino IDE* realisiert. Insbesondere bei jüngeren Schülerinnen stieß diese auf Ablehnung oder war nur schwer umsetzbar. Inspiriert von den deutlich kostenintensiveren, aber attraktiven Systemen wie *Calliope* und *microbit* wurde folglich auf eine blockbasierte Form der Programmierung mit *mBlock* umgestiegen. Diese Art der Programmierung löste mehr Begeisterung aus und führte dazu, dass die Schülerinnen deutlich mehr von sich aus ausprobierten. Die parallele Übersetzung der gesetzten Blöcke in textuelle Programmiersprache und die Möglichkeit, diese direkt auf den Mikrocontroller hochzuladen, ermöglicht gleichzeitig eine Binnendifferenzierung in einer Gruppe mit heterogenen Informatikkenntnissen. Das spätere gemeinsame Zusammenführen der individuell gestalteten Lichteffekte für die gewählten Wetterphänomene kennzeichnet einen zentralen Punkt für das Gemeinschaftsgefühl der Gruppe; jede einzelne Schülerin trägt einen Teil zum späteren Effektrepertoire bei und ist doch unabhängig in ihren Entscheidungen. Zusammenfassend lässt sich schlussfolgern, dass die beschriebenen Kriterien an ein gendersensitives Workshopkonzept von diesem Ansatz erfüllt werden. Ob diese Einordnung den gewünschten Motivationseffekt herbeiführen wird, bedarf weiterer Evaluation und Erprobung. In diesem Zusammenhang sei darauf verwiesen, dass der hier beschriebene Ansatz Spielraum für zahlreiche Variationen – sowohl für eine Verwendung in außerschulischen als auch schulischen Kontexten – lässt: denkbar wäre die Anbindung anderer APIs wie *Instagram* anstatt der *openweathermap* und andere parametrische Formen, wie bspw. Regenschirme, gestalten zu lassen.

Abschließende Anmerkung: Dieser Workshop wurde im Rahmen des Projekts *smile*, das vom Bundesministerium für Bildung und Forschung unter den Förderkennzeichen 01FP1611 bis 01FP1615 gefördert wird, konzipiert und mehrmals durchgeführt.

Literaturverzeichnis

- [An08] Anderson, N.; Lankshear, C.; Timms, C.; Courtney, L.: "Because it's boring, irrelevant and I don't like computers": Why high school girls avoid professionally-oriented ICT subjects. *Computers & Education*, 50(4):1304 – 1318, 2008.
- [CDT18] Chytas, C.; Diethelm, I.; Tsilingiris, A.: Learning programming through design: An analysis of parametric design projects in digital fabrication labs and an online makerspace. In: *IEEE Global Engineering Education Conference (EDUCON)*. S. 1978–1987, 2018.
- [Ch13] Cheryan, S.; Plaut, V. C.; Handron, C.; Hudson, L.: The Stereotypical Computer Scientist: Gendered Media Representations as a Barrier to Inclusion for Women. *Sex Roles*, 69(1):58–71, 2013.
- [CTD19] Chytas, C.; Tsilingiris, A.; Diethelm, I.: Exploring Computational Thinking Skills in 3D Printing: A Data Analysis of an Online Makerspace. In: *2019 IEEE Global Engineering Education Conference (EDUCON)*. IEEE, S. 1173–1179, 2019.

- [DKW11] Diethelm, I.; Koubek, J.; Witten, H.: IniK — Informatik im Kontext. LOG IN, 31(1):97–104, 2011.
- [DKW14] Dittert, N.; Katterfeldt, E.; Wilske, S.: Programming Jewelry: Revealing Models behind Digital Fabrication. In: FabLearn Europe. June 2014.
- [Eg16] Eguchi, A.: RoboCupJunior for Promoting STEM Education, 21st Century Skills, and Technological Advancement Through Robotics Competition. Robot. Auton. Syst., 75(PB):692–699, 2016.
- [FM02] Fisher, A.; Margolis, J.: Unlocking the Clubhouse: The Carnegie Mellon Experience. SIGCSE Bull., 34(2):79–83, 2002.
- [FMM97] Fisher, A.; Margolis, J.; Miller, F.: Undergraduate Women in Computer Science: Experience, Motivation and Culture. In: Proc. of the Twenty-eighth SIGCSE Technical Symposium on Computer Science Education. SIGCSE '97, ACM, New York, S. 106–110, 1997.
- [JB13] Jacobs, J.; Buechley, L.: Codeable objects: computational design and digital fabrication for novice programmers. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, S. 1589–1598, 2013.
- [KKR17] Krisch, O.; Kastl, P.; Romeike, R.: 3D-Druck als Motivation für die Einführung in die Programmierung in der Realschule. Informatische Bildung zum Verstehen und Gestalten der digitalen Welt, 2017.
- [KI16] Klan, F.: Informatik – reine Männersache? LOG IN, 36(1):35–38, 2016.
- [Ma17] Master, A.; Cheryan, S.; Moscatelli, A.; Meltzoff, A. N.: Programming experience promotes higher STEM motivation among first-grade girls. Journal of Experimental Child Psychology, 160:92 – 106, 2017.
- [MNS08] Mitnik, R.; Nussbaum, M.; Soto, A.: An autonomous educational mobile robot mediator. Autonomous Robots, 25(4):367–382, Nov 2008.
- [PD18] Pancratz, N.; Diethelm, I.: Including Part-Whole-Thinking in a Girls' Engineering Course through the Use of littleBits: A Practical Report on Including Part-Whole-Thinking into the Content of Computer Science Education. In: IEEE Global Engineering Education Conference (EDUCON). S. 1134–1138, 2018.
- [Se17] Sentance, S.; Waite, J.; Hodges, S.; MacLeod, E.; Yeomans, L. E.: “Creating Cool Stuff” - Pupils' experience of the BBC micro:bit. In: Proceedings of the 48th ACM Technical Symposium on Computer Science Education. 2017.
- [SM11] Stoilescu, D.; McDougall, D.: Gender digital divide and challenges in undergraduate computer science programs. Canadian Journal of Education, 34(1):308–333, 2011.
- [Wi16] Witherspoon, E. B.; Schunn, C. D.; Higashi, R. M.; Baehr, E. C.: Gender, interest, and prior experience shape opportunities to learn programming in robotics competitions. International Journal of STEM Education, 3(1):18, 2016.
- [YAM19] Yadav, A.; Alexander, V.; Mehta, S.: Case-based Instruction in Undergraduate Engineering: Does Student Confidence Predict Learning? International Journal of Engineering Education, 35(1):25–34, 2019.
- [ZS08] Zimmermann, L.; Sprung, G.: Technology is Female: How Girls Can Be Motivated to Learn Programming and Take up Technical Studies through Adaptations of the Curriculum, Changes in Didactics, and Optimized Interface Design. In: Proc. of ICEE 08, Pécs. 2008.

Einsatz von Skill Cards und Story Cards für einen kreativitätsfördernden Informatikunterricht auf der Sekundarstufe I

Thomas Schmalfeldt¹

Abstract: Um die Mittel der Informatik nicht nur gemäß vorgegebenen Routinen und Standardprozessen, sondern auch kreativ nutzen zu können, benötigen Schülerinnen und Schüler gewisse Grundlagen des Programmierens, die ihnen in einer Weise vermittelt werden sollten, die neben den informatischen Basiskompetenzen auch die Kreativität gezielt fördert. Im vorliegenden Beitrag wird im Kontext des Physical Computings mit dem Calliope mini der Kompetenzaufbau im Bereich „Informatik“ mithilfe des Einsatzes sogenannter „Skill Cards“ und „Story Cards“ aufgezeigt, die an der Pädagogischen Hochschule Zürich neu entwickelt wurden. Anhand der Skill Cards werden grundlegende Konzepte des Programmierens erarbeitet, welche die Schülerinnen und Schüler zur Umsetzung eigener Projektideen nutzen können. Der kreative Gestaltungsprozess wird parallel dazu durch eine reduzierte Form der Skill Cards – die Story Cards – unterstützt. Die im Beitrag vorgestellten Materialien sind Teil eines laufenden Forschungsprojekts, in dessen Rahmen das Ziel verfolgt wird, zukunftsweisende Lernmaterialien für den Informatikunterricht zu entwickeln und zu erproben.

Keywords: Kreativität, Physical Computing, Calliope mini, Einführung Programmieren, Sekundarstufe I, Skill Cards, Story Cards, Differenzieren im Informatikunterricht

1 Einleitung

Angesichts der fortschreitenden Digitalisierung und der damit einhergehenden zunehmenden Bedeutung informatischer Bildung sind innovative Lernumgebungen erforderlich, die es Kindern und Jugendlichen ermöglichen, sich entsprechende Kompetenzen bereits im Schulunterricht anzueignen. Vor diesen Hintergrund wurden die im vorliegenden Beitrag vorzustellenden Skill Cards und Story Cards entwickelt, mit deren Einsatz zwei Hauptziele verfolgt werden: Die Schülerinnen und Schüler sollen bei der Beschäftigung mit dem Calliope mini Programmierkompetenzen aufbauen und dabei gleichzeitig in ihrer Kreativität gefördert werden. Des Weiteren wurde bei der Konzeption der Karten darauf geachtet, dass sie einen differenzierenden Unterricht erlauben und sich fächerübergreifend einsetzen lassen. Die Arbeit mit einem Mikrocontroller wie dem Calliope mini ist in diesem Zusammenhang insofern zweckmäßig, als Mikrocontroller Bestandteil sehr vieler Alltagsgegenstände sind und deshalb auch im täglichen Leben von Relevanz sind. Zudem konnte nachgewiesen werden, dass Physical Computing einen positiven Einfluss auf die Motivation von Schülerinnen und Schülern haben kann [PR18].

¹ Pädagogische Hochschule Zürich, Mathematik und Informatik Sekundarstufe I, Lagerstrasse 2, CH-8090 Zürich, thomas.schmalfeldt@phzh.ch

Die angestrebte Förderung der Kreativität wiederum lässt sich mit einer These hinsichtlich der „21st-century skills“ begründen, der zufolge sich zukünftige Probleme im Alltag und im Berufsleben nicht mehr nur durch eingeübte Standardverfahren lösen lassen. Vielmehr werden zu deren Lösung zunehmend auch unkonventionelle Ansätze gefordert sein [TF09]. Dies bedeutet beispielsweise, dass Jugendlichen lernen müssen, größere Projekte gemeinsam anzugehen und dabei ihre jeweiligen Stärken in die Gruppe einzubringen. Ob die Arbeit mit Skill Cards und Story Cards, zum Beispiel in einem Design-Thinking-Prozess, einen Beitrag zur Umsetzung dieses Ziels leisten kann, soll im Rahmen eines laufenden Forschungsprojekts geklärt werden.

2 Materialset für den kreativitätsfördernden Informatikunterricht

Die Skill Cards und die Story Cards haben einen klaren Bezug zur Arbeit mit dem Calliope mini und bilden die Basis für einen kreativitätsfördernden Informatikunterricht. Das vollständige Materialset besteht aus den folgenden vier Bestandteilen:

- *Skill Cards*: Aufbau von Fertigkeiten und Kompetenzen für die Arbeit mit dem Calliope mini. Grundlage für das Erlangen der entsprechenden Story Card.
- *Story Cards*: Grundlage für die Ideenfindung zum Verfassen einer eigenen Geschichte und für die Konzeptionierungsphase der Umsetzung dieser Idee mit dem Calliope mini; gewissermaßen eine reduzierte Variante der Skill Cards.
- *Zubehör-Boxen*: Enthalten zusätzliche Komponenten wie zum Beispiel Sensoren und Aktoren, die am Calliope mini angeschlossen werden können und dessen Funktion erweitern.
- *Challenge Cards*: Offene Problemstellungen und Ausgangspunkte für das Erfinden von Geschichten, die sich auch mit anderen Mikrocontrollern umsetzen lassen.

Auf den Story Cards und den Zubehör-Boxen ist jeweils ein Icon² angebracht, das sich auch auf den Skill Cards wiederfindet. Dieses optische Element unterstützt die Orientierung und erlaubt eine einfache Handhabung der einzelnen Bestandteile des Materialsets. Diese werden nachfolgend kurz beschrieben.

2.1 Skill Cards

Die Skill Cards bilden die Grundlage des Kompetenzaufbauprozesses. Auf diesen Karten wird aufgezeigt, wie gewisse Hardware-Komponenten an den Calliope mini angeschlossen werden und wie deren Steuerung programmiert wird. Sie sind immer gleich aufgebaut und beginnen mit einer „Ich möchte ...“-Formulierung, wie zum Beispiel „Ich möchte einen Motor anschließen und diesen mit dem Calliope steuern“. Auf der

² Quelle: <https://thenounproject.com>

Vorderseite wird gegebenenfalls benötigtes Zubehör aufgeführt. Außerdem wird beschrieben, was auf der Rückseite erklärt wird, und es finden sich allgemeine Hinweise zur verwendeten Hardware. Auf der Rückseite wird bildlich und sprachlich dargestellt, wie das erforderliche Zubehör angeschlossen werden muss. Des Weiteren werden die dazugehörigen Codeblöcke vorgestellt und erklärt (vgl. Abb. 1). Für die Skill Cards, und somit auch für den Unterricht, wurde die blockbasierte Umgebung MakeCode³ verwendet.

Diejenige Version der Skill Cards, die im vorliegenden Beitrag beschrieben wird, zeigt vor allem den Umgang mit Sensoren und Aktoren auf. Die Erläuterungen beziehen sich zum einen auf die bereits auf dem Calliope mini befindlichen Bauteile, zum anderen aber auch auf zusätzliche Komponenten wie Grove-Module, Taster, Regler, LEDs und andere Bauteile. In der ersten Erprobungsphase wurden bereits Skill Cards zu Programmierkonzepten wie Bedingungen und Schleifen erstellt, die in der zweiten Erprobungsphase von Beginn an verwendet werden sollen.

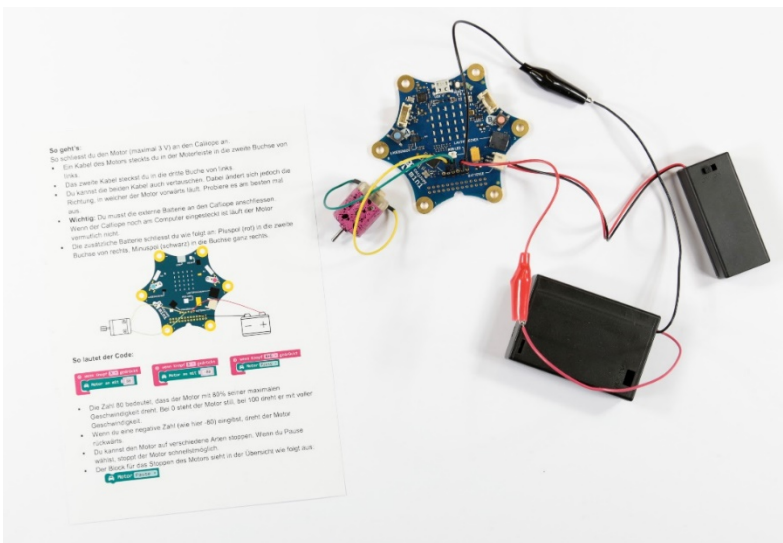


Abb. 1: Rückseite der Skill Card für die Steuerung eines Motors

2.2 Story Cards

Die Story Cards stellen in gewisser Weise die Kurzform der Skill Cards dar und bilden den Ausgangspunkt für den Kreativprozess. Wenn eine Schülerin oder ein Schüler die auf einer Skill Card beschriebenen Fertigkeiten aufgebaut hat, erhält sie bzw. er die entsprechende Story Card. Auf der Vorderseite befinden sich das jeweilige Icon (welches sich auch auf der dazugehörigen Skill Card befindet) sowie ein Schlagwort. Auf der

³ <https://makecode.calliope.cc>

Rückseite sind ein bis zwei QR-Codes aufgeführt, mit denen Erklärvideos abgerufen werden können. Diese Erklärvideos sind als Ergänzung zu den Skill Cards gedacht und ermöglichen es zum einen, die Story Cards kompakt zu halten, während zum anderen zugleich die Option besteht, bei Bedarf eine Repetition der entsprechenden Fertigkeiten anzubieten. Die Verwendung der QR-Codes ist vor allem dann sinnvoll, wenn die Schülerinnen und Schüler die Videos während des Unterrichts mit einem Smartphone oder einem Tablet direkt abrufen dürfen. Sollte dies nicht möglich oder erlaubt sein, können die Story Cards auch ohne QR-Codes ausgedruckt werden. In diesem Fall können die Jugendlichen auf die betreffenden Skill Cards zurückgreifen, wenn sie die erforderlichen Fertigkeiten nicht mehr präsent haben.

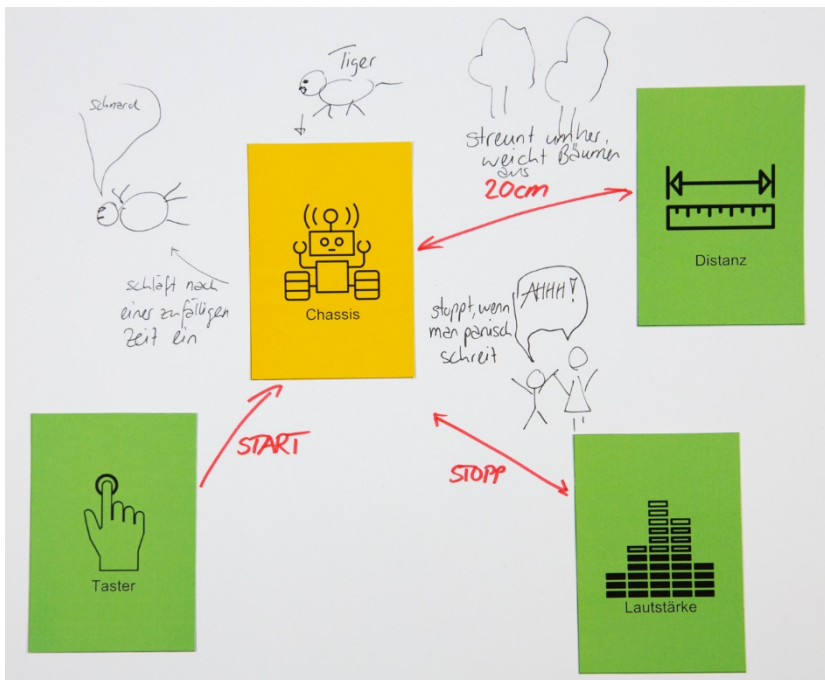


Abb. 2: Anhand von Story Cards erzählte Geschichte, inklusive erster technischer Kommentare (reproduziert nach einer Schülerinnenlösung)

Mithilfe der Story Cards sollen sich die Schülerinnen und Schüler ihren Fähigkeiten und Fertigkeiten entsprechende Geschichten ausdenken. Die Karten können ausgelegt, mit Pfeilen verbunden und kommentiert werden (vgl. Abb. 2). Das dadurch entstandene Netz dient in einem weiteren Schritt als Unterstützung bei der technischen Umsetzung mit dem Calliope mini. Zu diesem Zweck kann das Netz durch programmierspezifische und technische Bemerkungen ergänzt werden. Je mehr Story Cards die Schülerinnen und Schüler sammeln, desto mehr Möglichkeiten stehen ihnen für das Erfinden ihrer Geschichten und den Bau der darin vorkommenden Charaktere offen. Dieses Konzept

kann die Jugendlichen dazu motivieren, möglichst viele Kompetenzen aufzubauen, damit sie die persönliche Sammlung von Story Cards vergrößern können.

Obwohl die Story Cards einen direkten Bezug zu den Skill Cards aufweisen, lassen sie sich auch unabhängig davon verwenden. Dadurch können die den Story Cards zugrunde liegenden Fertigkeiten auch mithilfe anderer Methoden und Unterrichtsformen aufgebaut werden oder es kann direkt mit den Videos gearbeitet werden, die wie bereits ausgeführt mittels der QR-Codes abrufbar sind. Darüber hinaus eignen sich die Story Cards auch sehr gut für einen interdisziplinären Einsatz. So können die selbst erfundenen Geschichten zum Beispiel im Sprachunterricht in einem Aufsatz ausformuliert werden, während im Werkunterricht eine gestalterische Umsetzung der Charaktere der Geschichte stattfinden kann. Auf diese Weise wird aus einem Radroboter wie in Abb. 2 illustriert ein im Wald umherstreunender Tiger, der geschmeidig den einzelnen Bäumen ausweicht und so seinen Weg findet. Wenn der Tiger auf Menschen trifft und diese panisch schreien, bleibt er stehen, faucht und wartet, bis es wieder ruhig geworden ist. Da Tiger wie alle Katzenartigen viel schlafen, bleibt er regelmäßig (und zufällig) stehen und macht eine Schlafpause, um anschließend wieder herumzuströmen.

2.3 Zubehör-Boxen

Sind auf den entsprechenden Skill Cards zusätzliche Komponenten aufgeführt, so finden die Schülerinnen und Schüler diese in Zubehör-Boxen mit den entsprechenden Icons. Sofern dies möglich war, wurden Blanks-Streichholzschachteln verwendet (vgl. Abb. 3). Bei den Grove-Modulen wurden die jeweiligen Kabel beigelegt, an gewisse Bauteile wurden Kabel angelötet. In anderen Fällen muss gegebenenfalls auf zusätzliche Kabel mit Krokodil- und Pinanschlüssen zurückgegriffen werden.



Abb. 3: Zubehör-Boxen mit Materialien für die Eingabe (grün: Taster und Ultraschalldistanzmesser) und Ausgabe (gelb: DC-Motor und Hubmagnet)

2.4 Challenge Cards

Den Challenge Cards kommt eine andere Funktion zu als den zuvor beschriebenen Materialien. Sie lassen sich unabhängig von den Skill Cards und den Story Cards nutzen und können auch bei der Arbeit mit anderen Mikrocontrollern eingesetzt werden. Aus diesem Grund wird dieser Bestandteil des Materialsets im vorliegenden Beitrag nur am Rande erwähnt. Ganz allgemein lässt sich jedoch festhalten, dass es sich dabei um eine Sammlung von möglichen Problemstellungen handelt, welche die Schülerinnen und Schüler mithilfe ihrer bis zum gegebenen Zeitpunkt aufgebauten Kompetenzen bearbeiten können (vgl. Abb. 4). Sie bilden den Ausgangspunkt für die zu erfindenden Geschichten.

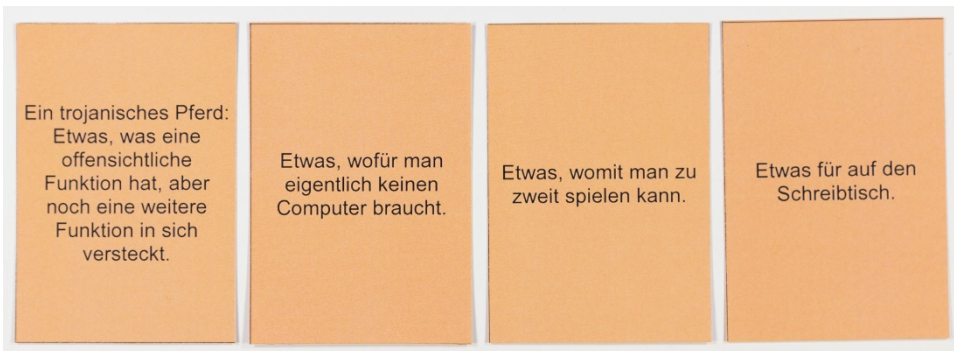


Abb. 4: Eine Auswahl an Challenge Cards

Der Einsatz der Challenge Cards lässt sich verschiedenartig gestalten. So kann eine bestimmte Karte die Problemstellung für die ganze Klasse bzw. eine Gruppe vorgeben oder die einzelnen Schülerinnen und Schüler ziehen individuell eine Karte. Des Weiteren kann die Arbeit in Projektteams, in Partner- oder in Einzelarbeit stattfinden. Die Sammlung der Challenge Cards lässt sich durch die Lehrpersonen, aber auch durch die Schülerinnen und Schüler selbst jederzeit erweitern.

3 Konzeption der Kartensets

Die hinter der Entwicklung der Skill Cards und der Story Cards stehende Idee grenzt sich von Konzepten, die in Büchern wie „Das Calliope-Buch“ [Be17] oder „Calliope mini: coden, basteln, entdecken“ [Ki18] vorgeschlagen werden, insofern ab, als diese stark auf Projekten basieren, bei denen die Schülerinnen und Schüler die vorgegebene Lösung schrittweise nachprogrammieren können. Die grundlegenden Konzepte des Programmierens werden bei diesem Ansatz im Zusammenhang mit den jeweiligen Projekten erläutert. Die „Scratch Coding Cards“ [Ru16] wiederum greifen solche Basiskonzepte, die sich die Schülerinnen und Schüler in kurzen Programmsequenzen aneignen sollen, auf einzelnen Karten auf. Diese Sequenzen werden aber ebenfalls an konkreten Beispielen festgemacht. So wird beispielsweise das Konzept der zufälligen

Zahlen anhand eines zufällig generierten Songs aufgezeigt. Da am Ende jeweils ein funktionsfähiges Produkt vorliegt, führen solche projektbasierten Ansätze zwar fortlaufend zu Erfolgserlebnissen; es besteht jedoch die Gefahr, dass die Schülerinnen und Schüler die einzelnen Projekte der Reihe nach abarbeiten, ohne danach in der Lage zu sein, ihre Fertigkeiten später auf einen anderen Kontext zu übertragen.

Im Gegensatz dazu verfolgen die Skill Cards zunächst einen technischen Ansatz. So wird beispielsweise das Erzeugen von zufälligen Zahlen ohne die Vorgabe einer konkreten Umsetzung vorgestellt. Daher ist darauf zu achten, dass die Schülerinnen und Schüler immer wieder die Gelegenheit erhalten, ihre neu erworbenen Fertigkeiten und Kompetenzen in von der Lehrperson gezielt gewählten Projekten anzuwenden. In den ersten Unterrichtssequenzen können diese Projekte, wie dies in den beiden oben erwähnten Büchern vorgeschlagen wird, eher eng geführt sein, sodass es mit Sicherheit zu Erfolgserlebnissen kommt. Dennoch ist von entscheidender Bedeutung, dass auch die Story Cards bereits in der Anfangsphase eingesetzt werden, weil die kreative Arbeit damit ebenfalls erlernt werden muss. Im weiteren Verlauf des Schuljahres können die Schülerinnen und Schüler mit zunehmender Selbstständigkeit individuelle Projekte verfolgen. Die Entwicklung der Story Cards wurde durch die „Tiles IoT Cards“ [MGD17] inspiriert, wobei Letztere primär den Ideenfindungsprozess fokussieren, während die konkrete Umsetzung dieser Ideen nicht im Zentrum steht. In dieser Hinsicht unterscheiden sich die „Tiles IoT Cards“ denn auch von den neu entwickelten Kartensets, welche durch die Kombination von Skill Cards und Story Cards die praktische Umsetzung der eigenen Projekte nicht nur ermöglichen, sondern dies auch explizit zum Ziel haben.

Der Einsatz der beiden Kartensets erfolgt in verschiedenen Phasen des Lernprozesses: Mittels der Skill Cards werden wie erläutert die technischen Grundlagen aufgebaut, während die Story Cards die Funktion haben, die Schülerinnen und Schüler zum Erzählen einer Geschichte anzuregen. Es geht beispielsweise nicht darum, dass die Jugendlichen mit einem Distanzmesser eine LED zum Blinken und eine Melodie zum Erklingen bringen, sondern vielmehr darum, dass sie die Geschichte von einem Wichtel erzählen, der mit seinen Augen funkelt und zu pfeifen beginnt, wenn man in seine Nähe kommt. Oder die Jugendlichen erfinden eine digitale Schatztruhe, in der sie ein Geheimnis aufbewahren können, das nur an spezielle Freundinnen und Freunde weitergegeben wird. Während mit den Skill Cards auf der mittleren Taxonomiestufe nach Anderson und Krathwohl [AK01], das heißt auf der Stufe des Anwendens, gearbeitet werden kann, eignen sich die Story Cards aufgrund des Anspruchs einer kreativen Umsetzung für die Arbeit auf der höchsten Taxonomiestufe, derjenigen des Erschaffens oder Kreierens [Ro08]. Wie die Karten in einer Testklasse probeweise konkret eingesetzt wurden, wird nachfolgend im Sinne eines Praxisberichts kurz dargelegt.

4 Einsatz im Unterricht

In der Deutschschweiz wird Informatik im neu erarbeiteten gemeinsamen Lehrplan 21 in Kombination mit Medienbildung im 5. bis 7. sowie im 9. Schuljahr mit je einer Wochenlektion aufgeführt. Die Einführung des Lehrplans 21 findet in den verschiedenen Kantonen gestaffelt statt. Im Kanton der Testklasse wird das Schulfach „Medien und Informatik“ auf der Sekundarstufe I (7. bis 9. Klasse) mit Beginn des Schuljahres 2019/2020 eingeführt. Die nachstehenden Ausführungen beziehen sich auf die erste praktische Erprobung der Unterrichtsmaterialien in einer Klasse des 8. Schuljahrs. Im Anschluss daran werden Beobachtungen aus dem Unterricht und erste Rückmeldungen der Schülerinnen und Schüler berichtet.

4.1 Durchführung der Unterrichtssequenzen

Der Informatikunterricht in der Testklasse hatte im Sommer 2018 begonnen und erstreckte sich über das ganze Schuljahr. Die 12 Schülerinnen und Schüler (je zur Hälfte Mädchen und Jungen) besaßen zu jenem Zeitpunkt noch keine Informatikvorkenntnisse. Die im Folgenden berichteten Beobachtungen stammen aus dem ersten Schulhalbjahr, in welchem insgesamt 20 Informatik-Unterrichtssequenzen stattfanden. Der Autor des vorliegenden Beitrags fungierte zugleich als unterrichtende Fachlehrperson.

Den Schülerinnen und Schülern standen während des ganzen Schuljahrs sowohl ein persönlicher Laptop als auch je ein Calliope mini zur Verfügung. Obwohl der Calliope mini ursprünglich für die frühen Schuljahre konzipiert wurde, fiel die Wahl auf diesen Mikrocontroller, da er einen auf Physical Computing basierenden Unterricht ermöglicht und verschiedene Programmierumgebungen verfügbar sind. Die elementaren Programmierkonzepte wie Schleifen, bedingte Anweisungen oder Variablen wurden vornehmlich im Klassenunterricht, ergänzt durch Erklärvideos, aufgebaut. Zusätzlich wurden in der ersten Unterrichtssequenz zwei Skill Cards eingeführt: Die erste Skill Card beschrieb das Darstellen von Zahlen, Buchstaben und Symbolen auf der LED-Matrix des Calliope mini, mit der zweiten Karte konnte die Verwendung der beiden auf der Platine verbauten Taster erlernt werden. Dabei zeigte sich bereits eine erste Herausforderung bei der Arbeit mit den Skill Cards: Blockbasierte Programmierumgebungen wie MakeCode bieten jeweils alle verwendbaren Blöcke an, während auf den Skill Cards wegen ihrer beschränkten Größe und zur Erhöhung der Verständlichkeit nur der Umgang mit einer Auswahl der verfügbaren Blöcke vorgezeigt wird. Dies bereitete den Schülerinnen und Schülern jedoch wenig Mühe. Die meisten konzentrierten sich zwar hauptsächlich auf die konkret beschriebenen Blöcke, aber es gab auch solche, die andere Blöcke ausprobierten.

Neben der Arbeit mit neuen Skill Cards wurde im Klassenverband auch viel Zeit dafür aufgewendet, Wege aufzuzeigen, wie kreative Geschichten erzählt werden können, ohne dass man sich technisch überfordert. Mithilfe verschiedener Methoden, unter anderem aus dem Design Thinking [ID03], wurde mit den Schülerinnen und Schülern die

Herangehensweise im Kreativprozess thematisiert und vielseitig umgesetzt. Sie lernten dabei, ein Projekt zu planen, in Gruppen zusammenzuarbeiten, Prototypen herzustellen, zu testen und zu erweitern, den Prozess zu dokumentieren und ein finales Produkt herzustellen, das anschließend in der Klasse präsentiert wurde.

4.2 Beobachtungen und erste Rückmeldungen zum Unterrichtskonzept

Wie in Abschnitt 4.1. bereits erwähnt wurde, hatten die Schülerinnen und Schüler zuvor noch keine Programmiererfahrungen gesammelt. In einer informellen Befragung am Anfang des Schuljahres, in der nach den Einstellungen zur Informatik gefragt wurde, reichten die Antworten über das ganze Spektrum hinweg: Ein Schüler hielt fest, dass er beabsichtige, später eine Lehre als Informatiker zu beginnen, während ein anderer Schüler bereits in der ersten Lektion klarstellte, dass er Informatik „doof“ finde. Obwohl die Einstellungen zur Informatik zu Beginn der Unterrichtsreihe somit sehr unterschiedlich ausgefallen waren, arbeiteten die Schülerinnen und Schüler im Unterricht engagiert mit. Sie waren schnell in der Lage, selbst erste Programme zu erstellen und diese auf dem Calliope mini laufen zu lassen. Entsprechend erhielten die Jugendlichen schon in der zweiten Veranstaltung den Auftrag, eine eigene Projektidee zu entwickeln.

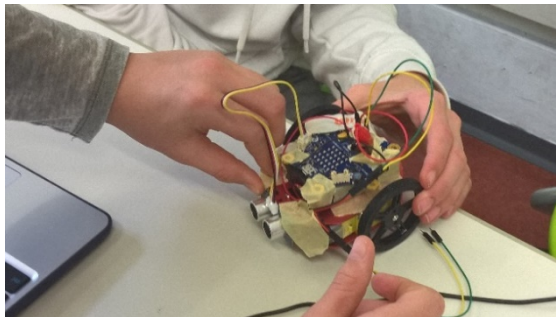


Abb. 5: Prototyp eines mithilfe des Calliope mini gesteuerten Radroboters

Da die Schülerinnen und Schüler im Laufe des ersten Schulhalbjahres unterschiedlich viele Story Cards erworben hatten, differenzierten sich die Niveaus im Unterricht fortlaufend aus. Dies ermöglichte es der Lehrperson, kleine Projektteams zusammenzustellen, in denen sich die einzelnen Schülerinnen und Schüler gut ergänzten. Die Jugendlichen organisierten sich aber auch selbst. So berichtete ein Schüler, dass er sehr gern mit einem anderen Schüler zusammenarbeite, weil dieser ein Experte im Umgang mit Hardware sei, während seine eigene Stärke im Programmieren liege (womit er das Entwickeln von geeigneten Algorithmen meinte). Ein anderer Schüler hatte beobachtet, dass eine Mitschülerin für ihr Projekt einen Entfernungsmesser nutzte (vgl. Abb. 5). Er bat sie um Hilfe, konnte sein eigenes Projekt entsprechend erweitern und erhielt nach einem kurzen Gespräch mit der Lehrperson selbst die Story Card mit dem Entfernungsmesser.

5 Fazit und Ausblick

Die im vorliegenden Beitrag vorgestellten Skill Cards und Story Cards wurden probenhalber bereits über ein halbes Schuljahr hinweg in einer Klasse des 8. Schuljahrs eingesetzt und dabei fortlaufend weiterentwickelt. Die verschiedenen Materialien haben sich bewährt. Während sich die Skill Cards erwartungsgemäß unkompliziert einsetzen ließen, musste für die Arbeit mit den Story Cards zu Beginn viel Zeit aufgewendet werden. Wie formative Leistungsdiagnosen während des Unterrichts ergaben, scheinen der angestrebte Kompetenzaufbau im Programmieren wie auch die Ziele der Kreativitätsförderung mehrheitlich erreicht worden zu sein. Darüber hinaus konnte eine Verbesserung der Zusammenarbeit in Teams beobachtet werden. Empirisch aussagekräftige Resultate werden jedoch erst am Ende des Forschungsprojekts vorliegen. Für das Schuljahr 2019/2020 ist eine erweiterte Erprobung in etwa drei bis fünf Klassen vorgesehen. Im Schuljahr 2020/2021 werden die Skill Cards und die Story Cards mit einem Mixed-Method-Ansatz umfassend evaluiert. Für die größer angelegte Evaluierungsphase ab 2020 wird das zurzeit noch zweckmäßig gehaltene Erscheinungsbild der Karten vereinheitlicht und ansprechender gestaltet. Sämtliche Materialien werden ab diesem Zeitpunkt öffentlich verfügbar sein.

Literaturverzeichnis

- [AK01] Anderson, L.; Krathwol, D.: A Taxonomy of Learning, Teaching, and Assessing, Longman, New York, 2001.
- [Be17] Bergner, N. et al.: Das Calliope-Buch: Spannende Bastelprojekte mit dem Calliope-Mini-Board, dpunkt.verlag, Heidelberg, 2017.
- [ID03] IDEO: IDEO Method Cards: 51 Ways to Inspire Design. William Stout Architectural Books, San Francisco, 2003.
- [Ki18] Kiefer, Ph.: Calliope mini: coden, basteln, entdecken, Vierfarben, Bonn, 2018.
- [MGD17] Mora, S.; Gianni, F.; Divitini, M.: Tiles: A Card-based Ideation Toolkit for the Internet of Things. In: Proceedings of the 2017 Conference on designing interactive systems. ACM, New York, S. 587-598, 2017.
- [PR18] Przybylla, M.; Romeike, R.: Impact of Physical Computing on Learner Motivation. In: Proceedings of the 18th Koli Calling International Conference on Computing Education Research (Koli Calling '18). ACM, New York, Article No. 9, 2018.
- [Ro08] Romeike, R.: Kreativität im Informatikunterricht, Dissertation. Universität Potsdam, Potsdam, 2008.
- [Ru16] Rusk, N.: Scratch Coding Cards, No Starch Press, 2016.
- [TF09] Trilling, B., Fadel, C.: 21st century skills: learning for life in our times, Jossey-Bass, San Francisco, 2009.

Das RetiBNE Café

Ein Konzept für Repair Cafés in der informatischen Bildung

Ann-Katrin Schmidt,¹ Carsten Schulte¹

Abstract: „Reparieren statt wegwerfen“ – eine Einstellung, die sich wachsender Beliebtheit erfreut. In vielen Städten weltweit lassen sich Repair Cafés finden, die der Öffentlichkeit einen Anlaufpunkt für die Reparatur technischer Geräte, wie z. B. Fahrräder, elektronische Haushaltsgeräte etc., bieten. In diesem Artikel beschreiben wir einen Transfer der Idee in die informatische Bildung. Dabei ergänzen wir das ursprüngliche Konzept um zwei Innovationen: einerseits wird der Reparaturprozess durch integrierte Reflexionen begleitet und andererseits liegt bei uns der Schwerpunkt auf Software-Reparatur (statt Hardware-Reparatur). Das Konzept ist im Zusammenhang mit der Reparatur von Smartphones von Studierenden entwickelt und mit Schulklassen erprobt worden. Daher werden erste Erfahrungen zur Wirkung und Durchführung des Konzepts vorgestellt und ergänzend Möglichkeiten beschrieben, wie das Konzept in schulischen und außerschulischen Bildungskontexten als Teil einer informatischen Bildung integriert werden kann.

Keywords: Repair Café; Reparieren; Bildung für nachhaltige Entwicklung (BNE); Software-Reparatur; Software-Upcycling

1 Einleitung

Am 25. September 2015 hat die Generalversammlung der Vereinten Nationen eine Resolution mit dem Titel „Transformation unserer Welt: die Agenda 2030 für nachhaltige Entwicklung“ verabschiedet. Dort werden 17 übergeordnete Ziele für eine nachhaltige Entwicklung formuliert, für dessen Erreichung sich die Gemeinschaft global einsetzt. Ziel 4 stellt dabei den Bereich „Bildung“ in den Mittelpunkt [UN15].

„Bildung für nachhaltige Entwicklung (BNE) ermöglicht es dem Individuum, aktiv an der Analyse und Bewertung von nicht nachhaltigen Entwicklungsprozessen teilzunehmen, sich an Kriterien der Nachhaltigkeit im eigenen Leben zu orientieren und nachhaltige Entwicklungsprozesse gemeinsam mit anderen lokal wie global in Gang zu setzen“ [DH08]. Um dies zu erreichen dient BNE der Entwicklung der sogenannten Gestaltungskompetenz. Sie umfasst unterschiedliche Teilkompetenzen, die für die Beurteilung von Entwicklungsprozessen als wichtig erachtet und bei [Pr08] und [DH08] näher erläutert werden. Dazu gehören u. a. die Fähigkeiten, interdisziplinär neues Wissen aufzubauen, Risiken und Gefahren zu erkennen und sich selbst aktiv in Entscheidungsprozesse einzubringen.

¹ Universität Paderborn, Didaktik der Informatik, Fürstenallee 11, 33102 Paderborn, Deutschland, vorname.nachname@uni-paderborn.de

Für einen Großteil der Fächer in Schulen bietet der „Orientierungsrahmen für den Lernbereich Globale Entwicklung“ Themen und Vorschläge, die die Integrierung von BNE erleichtern und unterstützen sollen [KB16]. Das Fach Informatik findet in diesem Rahmen jedoch keine Berücksichtigung.

Die schnelle technische Entwicklung und Verbreitung digitaler Artefakte im Alltag, macht es nach unserer Ansicht wichtig, sich auch in der informatischen Bildung mit nachhaltigen Entwicklungsprozessen auseinanderzusetzen. Das RetiBNE Projekt der Universität Oldenburg versucht, diese Lücke zu schließen, indem es das Ziel verfolgt, „Reparaturwissen und -können als Element einer technischen und informatischen Bildung für nachhaltige Entwicklung“ (kurz RetiBNE) zu vermitteln [Ar19]. Als Kooperationspartner bei diesem Projekt haben wir ein Konzept entwickelt, welches die Grundidee sogenannter Repair Cafés aufgreift und in die informatische Bildung integrieren will.

2 Konzeptentwicklung und Kontext

Unser Konzept stellt einen neuartigen Ansatz dar, der – soweit wir wissen – bisher nicht im Rahmen von Allgemeinbildung in den entwickelten Ländern getestet worden ist. Es gibt ähnliche Ideen, z.B. im Kontext der Entwicklungsländer [JPK11, AJR15] oder der Informatik im Allgemeinen [ARE15, Ho16]. Unser Ansatz sieht vor, Interessierte aktiv in die Reparatur defekter digitaler Artefakte einzubeziehen und die Geräte auf diese Weise besser zu verstehen. Dafür sind drei verschiedene Quellen zu einem stimmigen Konzept zusammengesetzt worden: Dualitätsrekonstruktion [Sc08], Ideen aus der Bildung für nachhaltige Entwicklung [DH08] und das Grundkonzept der Repair Cafés.

Umfragen zeigen, dass Nachhaltigkeit für 64% der deutschen Teenager ein wichtiges Thema ist, jedoch meist in deutlichem Zusammenhang mit Klimawandel, Armut und Welthunger betrachtet wird. Der Einsatz technischer Geräte in der eigenen Alltagsrealität spielt hingegen nur eine Nebenrolle [Bo09]. Die fehlende technischen Perspektive in der BNE greift das Projekt RetiBNE der Universität Oldenburg auf. In dessen Rahmen sind gemeinsam mit Kooperationspartnern eine Vielzahl von Unterrichtseinheiten entwickelt worden, die den Aufbau von Reparaturwissen und -können unterstützen.²

Im Mittelpunkt unseres Konzepts steht die Idee der Repair Cafés: Die meisten Initiativen in Deutschland sind Teil des „Netzwerk Reparatur-Initiativen“, das niederländische Netzwerk „Stichting Repair Café“ ist auch international tätig.³ In der Regel werden die Repair Cafés von Freiwilligen konzipiert, vorbereitet und betreut und stellen neben technischen und ressourcenschützenden Aspekten auch soziale Treffpunkte für Menschen aus den unterschiedlichsten Kontexten dar.

Im Wintersemester 2017/2018 haben sich Lehramtsstudierende der Universität Paderborn intensiv mit dem RetiBNE Projekt beschäftigt. Im Seminar in der Didaktik der Informatik

² Zugriff auf Materialien und weitere Informationen zum Projekt finden sich unter <http://www.retibne.de/>

³ Für weitere Informationen siehe <https://repaircafe.org/de/>

werden zu ausgewählten Themen Konzepte für Schulklassen erarbeitet, die einen Zyklus aus praktischer Erprobung, Reflexion und Überarbeitung durchlaufen. Zusätzlich werden Forschungsfragen im Zusammenhang mit dem Projekt bearbeitet. Damit folgt das Seminar den in [HS11] entwickelten Ideen zur Verbesserung der Lehrerbildung.

Für das Konzept ist die Unterrichtsreihe „Smartphone – Herstellung, Nutzung und Entsorgung“ aus dem RetiBNE Projekt genutzt und von den Studierenden für ihre Ziele angepasst und ergänzt worden. Die entstandene 3-stündige Lehr-Lerneinheit behandelt Themen wie den persönlichen Handykonsum, Aufbau und enthaltene Rohstoffe sowie Herstellung und Produktion von Smartphones, die im Hinblick auf Nachhaltigkeit, (sparsamen) Ressourcenverbrauch und den ökologischen Fußabdruck diskutiert werden. Der inhaltliche Rahmen wird durch einen Praxisteil ergänzt, der die Reparatur von defekten oder alten Handys und Zubehör thematisiert. Neben der üblichen Hardware-Reparatur in Repair Cafés, verfolgt unser RetiBNE Café zusätzlich die Reparatur von veralteter Software und dessen Anpassung an die eigenen Bedürfnisse. Teilnehmende sollen nicht nur Reparaturwissen erlangen und praktisch anwenden, sondern auch die duale Natur digitaler Artefakte kennenlernen. Wir vermuten, dass tiefere Kenntnisse über die duale Natur und Zusammenhänge im digitalen Artefakt zu einem besseren Verständnis beitragen und auf zukünftige Reparaturprobleme vorbereiten können.

3 Konzept des RetiBNE Cafés im Detail

Nachfolgend wird das Konzept und die Inhalte der einzelnen Phasen vorgestellt.

- 15 min Einführung: persönliche Handynutzung und -konsum, Aufbau eines Handys
- 15 min Gruppenarbeit: Rohstoffe im Handy, Herstellung des Handys
- 15 min Präsentation und Reflexion: Goldwert in Handys
- 90 min offene Reparaturphase an Stationen
- 10 min Reflexion der Reparaturphase
- 25 min Selbsttest und Gruppendiskussion: ökologischer Fußabdruck und Nachhaltigkeit im Alltag
- 10 min abschließende Reflexion inkl. Blitzlicht

Einführungsphase

Die Einführungsphase beginnt mit der Frage nach der Anzahl der Handys im Besitz der Teilnehmenden (aktuelle und nicht mehr genutzte Geräte). Die Ergebnisse werden offen an der Tafel gesammelt, sodass sich jeder ein eigenes Urteil über den individuellen Handykonsum bilden kann. Zusätzlich wird der durchschnittliche Besitz an Handys für die Gruppe berechnet.

Als nächstes sollen sich die Teilnehmenden über Materialien in ihren Smartphones informieren. Dazu erfolgt eine Gruppenarbeit, in der zu einigen Metallen und Erzen sowie Seltenen Erden gezielt Informationen gesammelt werden. Recherchiert wird dabei zum Wert des Rohstoffs, dem Herkunftsland, den Abbauverfahren, möglichen Folgen für die Region sowie zur Wiederverwendbarkeit und zum Recycling der Materialien. Anschließend werden die

Ergebnisse im Plenum vorgestellt.

Darüber hinaus erfolgt eine Berechnung des Goldwerts: Unter Annahme des durchschnittlichen Handybesitzes der Gruppe wird eine Hochrechnung auf die Schule und die entsprechende Region vorgenommen. Das Ergebnis dient als Motivation, um über das Potenzial der Einsparung von Ressourcen durch Recycling zu diskutieren. Diese Aktion stellt sicher, dass der eigene Handykonsum als Teil der wirtschaftlichen Ressourcennutzung berücksichtigt und realisiert wird.

Reparaturphase

Die Reparaturphase erfolgt selbstbestimmt durch die Teilnehmer, sodass sie entsprechend ihrer Interessen frei wählen können, ob sie in Einzelarbeit oder Kleingruppen reparieren und an welchen der drei Stationen sie arbeiten möchten. Aktuell stehen dafür die Stationen „Kabelbruch“, „Displaybruch“ und „Basteln“ zur Auswahl. Während der gesamten Phase werden die Teilnehmenden von der Hauptfrage „Warum funktioniert diese Funktion oder das digitale Artefakt nicht?“ geleitet.

Für die einzelnen Stationen stehen Ansprechpartner zur Verfügung, die als Mentoren nach dem Prinzip der minimalen Hilfe [Ae83] agieren. Damit wird das Ziel verfolgt, die Teilnehmenden möglichst autonom handeln und lernen zu lassen und ihnen bei Bedarf durch entsprechenden Input Ideen für die Weiterarbeit mitzugeben. Die Hilfestellungen erfolgen dabei mit Blick auf die Dualitätsrekonstruktion [Sc08], die eine differenziertere Formulierung der Hauptfrage „Warum funktioniert es nicht“ anregen soll. Durch dieses Vorgehen können die Teilnehmenden die beiden Seiten des digitalen Artefakts, das Architekturverständnis und das Bedeutungsverständnis, selbstständig entdecken⁴.

Darüber hinaus haben wir festgestellt, dass der konkretere und differenziertere Ansatz zum Verständnis des Problems (und damit des Artefakts) der Zerlegung des Warum durch Tinbergen [Ti63] in der Verhaltensbiologie ähnelt. Tinbergen hat vermutet, dass ein integratives Verständnis von Verhalten eine differenzierte Art der komplexen „Warum“-Frage benötigt. Daher haben wir vier Aspekte des Warum angepasst, um digitale Artefakte (oder deren Dysfunktion) in ihrer Gesamtheit zu erfassen. Um die Bedeutung des Artefakts zu verstehen, können wir fragen „Was ist der Zweck des digitalen Artefakts?“ und „Welche soziokulturellen Einflüsse führen zu diesem Artefakt?“. Mit Blick auf die Architektur können wir fragen: „Durch welche Bestandteile wird das Artefakt ermöglicht?“ und „Wie arbeiten diese Bestandteile zusammen?“. Diese differenzierten Aspekte ermöglichen die Kategorisierung und Analyse jedes einzelnen Schritts eines freien und selbstbestimmten Reparaturprozesses, der in der Regel zu komplex ist, um ihn zu erfassen.

Die Stationen „Displaybruch“ und „Kabelbruch“ laden die Teilnehmenden ein, Handys oder Teile davon in langsamen Schritten zu entdecken. Eine besondere Rolle in unserem Konzept spielt die Station „Basteln“, die keinen klaren Reparaturauftrag besitzt. Einerseits können die Teilnehmenden unterschiedliche alte Mobiltelefone vergleichen und mit Hilfe der adaptierten Formen des „Warum“ den architektonischen Entwicklungspfad als Teil

⁴ Wir haben uns dafür entschieden die Begriffe Architektur (statt Struktur) und Bedeutung (statt Funktion) zu verwenden, da diese den dualen Charakter nach unserer Erfahrung leichter zu verstehen und zu verwenden macht.

der Dualitätsrekonstruktion entdecken. Andererseits bietet sich die Möglichkeit, über die Begriffe „defekt“ und „reparieren“ in Bezug auf Software und damit über Software-Upcycling oder die alternative Nutzung technischer Artefakte nachzudenken. Da diese beiden Konzepte neu und wichtig für unser Konzept sind, stellen wir sie an dieser Stelle vor:

Unter *Software-Reparatur* verstehen wir die Aktualisierung älterer Smartphones, die nicht mehr vom Hersteller unterstützt werden, auf aktuellere Betriebssysteme. Aus praktischen Gründen liegt hierbei der Schwerpunkt auf Android-Smartphones. Geplante Obsoleszenz basiert oft auf Softwareproblemen, genauer gesagt auf dem Betriebssystem, und nicht auf Hardware-Gründen. Dies ist auf eine schlechte Update-Politik der Hersteller von Android-Smartphones zurückzuführen. So liefert beispielsweise einer der größten Hersteller Updates für sein Flaggschiffmodell im besten Fall für 2 Jahre, bei Midrange-Smartphones noch kürzer oder gar nicht. Einerseits ist der Sicherheitsaspekt ein Problem, andererseits bedeutet veraltete Software, dass neue Apps irgendwann nicht mehr funktionieren und damit das Smartphone obsolet machen.

Unter *Software-Upcycling* verstehen wir den Einsatz alter Hardware in anderen Anwendungsbereichen, z.B. die Verwendung eines alten Smartphones als Fahrradcomputer oder Überwachungskamera. In Gesprächen mit Lehrkräften ist die Rolle von Smartphones als Statussymbol als noch wichtiger identifiziert worden, als die Verwendung modernster Software. Deshalb bieten wir die Möglichkeit eigene ROMs zu verwenden, die ständig neue Versionen liefern und eine enorme Vielfalt an Individualität bieten. Nicht nur ältere Geräte können auf diese Weise eine Art zweites Leben erfahren, die durch die Nutzung in neuen Kontexten ermöglicht wird.

Reflexionsphase

In der abschließenden Reflexionsphase werden im Plenum die individuellen Erfahrungen aus der Reparaturphase gesammelt. Die Teilnehmenden sollen dabei auch diskutieren, wie viele ihrer eigenen Handys repariert oder anderweitig wiederverwendet werden könnten. An dieser Stelle soll auch die Software-Reparatur und das Software-Upcycling nochmal in den Blick genommen werden. Durch die Frage, was eine selbstständige Anpassung eines digitalen Artefakts bedeuten könnte, soll über die beiden Konzepte nachgedacht und diskutiert werden.

Diese Phase dient darüber hinaus auch dazu, den Handykonsum und das Reparieren mit dem Themenkomplex der Nachhaltigkeit zu verknüpfen. Die Teilnehmenden lernen den ökologischen Fußabdruck kennen und ermitteln mit Hilfe eines Online-Fragebogens ihren persönlichen Fußabdruck. Eine Eigenreflexion soll hier durch die Bekanntgabe des nachhaltigen Fußabdrucks und des deutschen Durchschnitts angeregt werden. Damit verfolgen wir das Ziel, dass Fragen der Nachhaltigkeit im Alltag bewusster wahrgenommen werden und beispielsweise die Einsparung von Ressourcen durch längere Nutzung von digitalen Artefakten und Maßnahmen wie Reparaturen und Upcycling erreicht werden kann.

4 Erste Ergebnisse und Reflexionen zum Konzept

Das im vorherigen Abschnitt vorgestellte Konzept zu unserem RetiBNE Café ist bei Schulklassenbesuchen erprobt und reflektiert worden. Dazu sind mehrere Beobachter anwesend gewesen, sodass allgemeine Eindrücke und Feedback aus freien Beobachtungen gesammelt werden konnten. Ergänzend haben die verantwortlichen Studierenden zu ausgewählten Aspekten Bewertungsinstrumente vorbereitet und ausgewertet. Die hier vorgestellten Ergebnisse basieren auf Besuchen von drei Klassen von drei verschiedenen Schulen und Schultypen. Insgesamt haben 67 Schüler teilgenommen (26, 23, 18).

Zu Beginn präsentieren wir allgemeine Ergebnisse, die auf den Beobachtungen der Studierenden, der begleitenden Lehrkräfte, des Universitätspersonals und Ergebnissen aus den Durchführungen basieren. Der Gesamteindruck war dabei wie folgt: Alle Kurse sind von allen Teilnehmenden gut angenommen worden (keine Polarisierung). Wir haben „typisch positive“ Phänomene von Exkursionen/Schülerlabortagen erlebt: die ansonsten ruhigen oder gar negativ auffallenden Schüler haben nicht nur im Hinblick auf ihr gewohntes Verhalten, sondern auch im Verhältnis zum Rest des Kurses einen guten bis sehr guten Beitrag geleistet. Alle Teilnehmer sind sehr konzentriert oder zumindest ruhig gewesen, sodass es zu keinen Störungen kam. Häufige Aussagen von Schülern während der jeweiligen Repair Cafés waren: „[Tag] war cool“, „Interessant, das Handy von innen zu sehen“, „cool, dass man auch seine eigenen Sachen mitbringen kann [defekte oder alte Geräte]“, „man konnte sich gut versuchen, da man keine Angst hatte, etwas zu beschädigen, weil alles schon kaputt war“.

Beobachtungen von Studierenden und Lehrkräften: Die Mini-Statistik über den eigenen Handykonsum faszinierte Schüler und Lehrer gleichermaßen und könnte ein guter Ausgangspunkt für weitere interessante Unterrichtsstunden sein.

Die Suche nach grundlegenden Informationen über die Reparatur von Handys ist mittels YouTube sehr einfach gewesen, als altersgerecht angesehen und daher gut angenommen worden. Die offene Reparaturphase ist als „didaktisch aufgeschlossen und methodenfrei“ angesehen worden, was die Schüler sehr geschätzt und mit großer Motivation genutzt haben. Diese offene Phase ist mit den „Warum“-Fragen sinnvoll angereichert worden und hat zum weiteren Nachdenken angeregt.

Neben der allgemeinen Beobachtung sind systematische Daten zu drei verschiedenen Fragen gesammelt worden. Uns hat u.a. interessiert, welche der Stationen „Kabelbruch“, „Displaybruch“ und „Basteln“ die Schüler am meisten fasziniert hat. Es hat sich herausgestellt, dass alle Stationen als ähnlich faszinierend angesehen worden sind. Wenn die Schüler mehrere besucht haben, scheint es keine Tendenz zu geben, welche sie bevorzugen.

Ein weiterer Aspekt unserer Beobachtungen ist der Einfluss von Gruppenkonstellationen auf den Reparaturprozess. Aufgrund des relativ hohen Aufwandes einer Beobachtungsstudie haben wir uns auf eine exemplarische Beobachtung von Gruppen konzentriert. Für die Bewertung der Gruppenkonstellationen ist ein Beobachtungsbogen erstellt und verwendet worden. Dieser beinhaltet insgesamt neun Verhaltensmuster, die sich in fünf kooperative

und vier nicht kooperative Verhaltensweisen unterteilen. In zwei der drei Klassenbesuche konnten wir vier klar wahrnehmbare Gruppen an insgesamt drei Stationen beobachten. Was das kooperative Verhalten betrifft, so lässt unsere Datenerhebung vermuten, dass zwei bis drei Personen eine vorteilhafte Gruppengröße darstellen. In einer Gruppe mit vier Personen hat mindestens eine Person ein „unbeteiligtes Zuhören“ gezeigt und insgesamt ist eine geringere aktive Teilnahme an der Aufgabe wahrgenommen worden. Zusätzlich gehen wir davon aus, dass eine ausgewogene Zusammensetzung aus „Experten“ und „Neulingen“ dazu beiträgt, alle Beteiligten persönlich einzubeziehen.

Bei den nicht kooperativen Aktivitäten haben wir festgestellt, dass 2 von 4 Gruppen zwischenzeitlich „nicht aufgabenbezogenes Arbeiten“ gezeigt haben. Weitere nicht kooperative Verhaltensweisen sind von uns nicht beobachtet worden.

Schließlich hat uns interessiert, inwieweit sich die Teilnehmer nach dem Besuch im RetiBNE Café noch mit dem Thema Reparatur beschäftigt haben. Dazu haben die Teilnehmenden zwei Wochen nach dem Besuch unseres Repair Cafés einen Fragebogen mit Aussagen erhalten, die sie bewerten sollten. 49 von 56 ausgefüllten Fragebögen waren gültig und in die Auswertung einbezogen.

Wir haben die Elemente des Fragebogens in drei „Aktionsphasen“ zusammengefasst: Experimentieren, Reflexion sowie erfolgreicher Abschluss. Wir erachten für einen erfolgreichen Abschluss ein vorheriges Experimentieren und Recherchieren zur Reparatur für notwendig, sodass die Befragten mehreren Aussagen zustimmen konnten. Daher ist es uns möglich gewesen, den Einfluss des RetiBNE Cafés zu bewerten:

51,0% der Teilnehmer haben das Thema nicht weiter betrachtet (n=25). Insgesamt 37 mal stimmen die Befragten Aussagen zu, die sich auf experimentelle Gedanken und Einflüsse beziehen. Des Weiteren stehen 25 Antworten mit Reflexionsaktivitäten in Zusammenhang. Zusätzlich geben 10 von 49 Personen eine erfolgreiche Reparatur eines Handys/Kabels an. Insgesamt haben damit mindestens 15% aller 67 Teilnehmenden erfolgreich ein defektes Kabel oder Handy nach dem Besuch im RetiBNE Café repariert.

5 Diskussion des Konzepts im Kontext informatischer Bildung

Nach Präsentation des Konzepts und den gewonnenen Ergebnissen werden wir das Konzept im Kontext der informatischen Bildung diskutieren. Wir glauben, dass das Konzept bekannte und etablierte Ziele der Informatik, wie z. B. Problemlösen, Verständnis der Funktionsweisen digitaler Artefakte und das Bewusstsein für gesellschaftliche Implikationen, anspricht. Darüber hinaus trägt es dazu bei, ein besseres Verständnis und eine detailliertere theoretische Untermauerung dieser allgemeinen Ziele in der informatischen Bildung zu entwickeln.

Die informatische Bildung und Informatik-Lehrpläne behaupten im Allgemeinen, einen wichtigen Aspekt für die Entwicklung von Fähigkeiten für das Leben in einer technologiebetriebenen Welt, für die Digitalisierung und zukünftige Entwicklungen beizutragen. Bell und Duncan erwähnen zum Beispiel die folgenden Ziele [BD18]:

1. „Verständnis dafür wie alles funktioniert“
2. „sie befähigen, Schöpfer [nicht nur] Verbraucher zu sein“
3. (verstehen, dass) „die gleiche Hardware für die Arbeit mit einer Vielzahl von Informationen verwendet werden kann“
4. „Themen in den Kontext stellen“
5. „werden sich bewusst, dass, wenn die Schüler etwas über Programmierung lernen, sie immer den Menschen im Auge behalten müssen“
6. „lernen, sich darauf zu konzentrieren, die Bedürfnisse des Endverbrauchers über die des Programmierer zu stellen“

Im Hinblick auf das RetiBNE Café ist es interessant, dass die Studierenden und Teilnehmenden oft zum ersten Mal in ihrem Leben ein Smartphone geöffnet und darüber nachgedacht haben, wie es funktioniert (1). Die Reparatur und das Upcycling unterscheiden sich vom täglichen Gebrauch als Nutzer und haben es ihnen ermöglicht, Schöpfer zu sein (2). Gleichzeitig trägt dies zur Idee bei, dass Geräte für etwas anderes verwendet werden können (3), z. B. kann ein altes Telefon als Fahrrad-Tacho wiederverwendet werden, ... Interessant ist auch, dass sich der Blick auf die Technologie (Smartphone) und die Nutzungsmuster der Involvierten geändert haben: Sie haben begonnen, ihre Gewohnheiten zu reflektieren und den Kauf eines neuen Telefons in einem breiteren ökologischen Kontext zu sehen (4) und ein Bewusstsein dafür entwickelt, dass es nicht nur eine Sache der Nutzung ist, sondern auch Menschen außerhalb des eigenen Umfelds betrifft (5,6). In diesem Sinne ist es eine Intention gewesen, durch die Anpassung, Reparatur oder das Upcycling eines Gerätes „Verantwortung zu übernehmen“. Anfangs wird das Reparieren eines alten Smartphones als nicht wertvoll betrachtet, da diese eine Art Statussymbol sind – je neuer, desto besser. Aber durch das Upcycling wird das Handy einzigartig, angepasst an die eigenen Handlungen und Ideen – und dann ist es wieder ein wertvolles Statussymbol, auf das die Lernenden stolz sind.

Es wäre interessant, in weiteren Studien herauszufinden, ob insbesondere die Station „Basteln“ für Hobby-Bastler von großem Wert sein könnte, wenn es um kollaborative Aspekte der gemeinsamen Reparatur nach den von Houston et al. [Ho16] aufgezeigten Aspekten geht.

49% der deutschen Jugendlichen geben deutlich an, dass nicht nur Regierungen, Unternehmen und große Organisationen ihr Verhalten ändern müssen, sondern auch die Bevölkerung [Bo09]. Obwohl der Kontext der Frage in unserem Repair Café für nachhaltiges Verhalten ganz anders ist, haben auch hier 49% der Befragten angegeben, sich nach dem Besuch mit dem Thema beschäftigt zu haben. Darüber hinaus ist das Upcycling in den Kontext der „Wegwerfgesellschaft“ gestellt worden: die Schüler aus unseren Klassenbesuchen haben im Durchschnitt mindestens drei Handys und bis zu 8 Geräte besessen. Hier sind die zwei Wochen später gemessenen Effekte interessant. Nach unseren Erfahrungen und auf Grundlage von Gesprächen mit Anbietern außerschulischer Aktivitäten sind längerfristige Effekte von einmaligen Angeboten schwer zu erreichen. Die Ergebnisse deuten also auf eine hohe Wirkung des RetiBNE Café hin.

6 Mögliche Integration in Bildungskontexte

Nach unseren gewonnen Erkenntnissen sehen wir das RetiBNE Café als gute Möglichkeit an, die informatische Bildung nicht nur in schulischen Bildungskontexten, sondern auch in außerschulischen Bereichen zu stärken. Daher möchten wir unser RetiBNE Café für Schulklassen, Gruppen und interessierte Einzelpersonen öffnen und ihnen so die Möglichkeit geben, Reparaturwissen und -können zu erwerben und im Kontext der Nachhaltigkeit zu sehen. Wir sehen das RetiBNE Café daher auch als Beitrag zur Verbraucherbildung und wollen eine reflektierte Urteils- und Handlungsfähigkeit fördern.

Unser Konzept soll daher auf zwei Arten weiter umgesetzt werden: Einerseits gibt es die Möglichkeit an offenen Repair Café Sitzungen teilzunehmen und eigene defekte Kleingeräte wie Handys, Tablets, Laptops o. ä. mitzubringen und mit Unterstützung von erfahrenen Ansprechpartnern zu reparieren. Auch hier soll das Konzept der Reparaturphase mit dem Grundgedanken der Dualität als Leitlinie dienen und den Teilnehmenden ermöglichen, ihr technisches Gerät nicht nur zu reparieren, sondern auch insgesamt besser zu verstehen. Andererseits wird die Möglichkeit bestehen, den 3-stündigen Kurs in der hier vorgestellten Form mit Gruppen zu besuchen.

Bei der im August 2017 in Kraft getretenen Rahmenvorgabe Verbraucherbildung in Schulen in Nordrhein-Westfalen handelt es sich um eine fächerübergreifende Vorgabe, zu deren Umsetzung Schulen in ihren internen Lehrplänen verpflichtet sind. Ziel von Verbraucherbildung ist die Entwicklung eines verantwortungsbewussten Verhaltens als Konsument, welches reflektierte und informierte Entscheidungen beinhaltet. In der Rahmenvorgabe wird auch auf die Agenda 2030 der UN verwiesen. Sie sieht eine Entwicklung von konsumbezogenen Kompetenzen vor, u. a. Sachkompetenz und Handlungskompetenz, in denen wir Parallelen zu Teilen der Gestaltungskompetenz erkennen.

In unseren Augen kann das RetiBNE Café wichtige Beiträge zur Entwicklung dieser Kompetenzen beitragen und diese um Kenntnisse und Fähigkeiten zur Reparatur von technischen Geräten erweitern. Denkbar wäre etwa, Teile des Konzepts in den Schulunterricht zu verlagern und sich bereits vorher mit Nachhaltigkeit, Rohstoffgewinnung und Produktion technischer Kleingeräte auseinanderzusetzen. Im Besuch des RetiBNE Cafés kann dann eine Vertiefung und Erweiterung um den Reparaturgedanken stattfinden und gleichzeitig praktisch von Schülern erprobt werden. Im Anschluss an den Besuch bieten die neuen Erkenntnisse Möglichkeiten zur Weiterarbeit im Unterricht, etwa über eine Diskussion der technischen und gesellschaftlichen Wechselwirkungen und Interessen.

7 Danksagung

Unser Dank geht an die Studierenden Dominik Brett, Jonas Laukötter, Ann-Katrin Schmidt und Dimitri Schneider. Sie haben die Schüler während der drei Repair Café Sitzungen hochmotiviert betreut und die hier vorgestellten Auswertungen durchgeführt. Gleichzeitig bedanken wir uns herzlich bei Jessica Krüger und Andreas Gödecke, die den Studierenden bei der Entwicklung des Konzepts mit Rat und Tat zur Seite standen.

Literaturverzeichnis

- [Ae83] Aebli, Hans: Zwölf Grundformen des Lehrens. Klett Cotta, Stuttgart, 1983.
- [AJR15] Ahmed, Syed Ishtiaque; Jackson, Steven J.; Rifat, Md. Rashidujjaman: Learning to Fix: Knowledge, Collaboration and Mobile Phone Repair in Dhaka, Bangladesh. In: Proceedings of the Seventh International Conference on Information and Communication Technologies and Development. ICTD '15, ACM, NY, USA, S. 4:1–4:10, 2015.
- [Ar19] Arbeitsgruppe Technische Bildung/Carl von Ossietzky Universität Oldenburg: , RETIBNE, 2019. Abrufbar über <http://retibne.de/ueber>.
- [ARE15] Ames, Morgan G.; Rosner, Daniela K.; Erickson, Ingrid: Worship, Faith, and Evangelism: Religion As an Ideological Lens for Engineering Worlds. In: Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing. CSCW '15, ACM, New York, NY, USA, S. 69–81, 2015.
- [BD18] Bell, Tim; Duncan, Caitlin: Teaching Computing in Primary Schools. In (Sentance, Sue; Barendsen, Erik; Schulte, Carsten, Hrsg.): Computer Science Education Perspectives on Teaching and Learning in School. Bloomsbury Academic, S. 132 – 150, 2018.
- [Bo09] Boecker, Malte Christopher: Ergebnisse einer repräsentativen Umfrage in Deutschland und Österreich „Jugend und Nachhaltigkeit“. Aug 2009.
- [DH08] De Haan, Gerhard: Gestaltungskompetenz als Kompetenzkonzept der Bildung für nachhaltige Entwicklung. In: Kompetenzen der Bildung für nachhaltige Entwicklung. Springer, S. 23–43, 2008.
- [Ho16] Houston, Lara; Jackson, Steven J.; Rosner, Daniela K.; Ahmed, Syed Ishtiaque; Young, Meg; Kang, Laewoo: Values in Repair. In: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems. CHI '16, ACM, NY, USA, S. 1403–1414, 2016.
- [HS11] Hornung, Malte; Schulte, Carsten: ProspectiveTeachers@Research: CS Teacher Education Revised. In: Proceedings of the 11th Koli Calling International Conference on Computing Education Research. Koli Calling '11, ACM, New York, NY, USA, S. 138–143, 2011.
- [JPK11] Jackson, Steven J.; Pompe, Alex; Krieshok, Gabriel: Things Fall Apart: Maintenance, Repair, and Technology for Education Initiatives in Rural Namibia. In: Proceedings of the 2011 iConference. iConference '11, ACM, New York, NY, USA, S. 83–90, 2011.
- [KB16] Kultusministerkonferenz; Bundesministerium für wirtschaftliche Zusammenarbeit: Orientierungsrahmen für den Lernbereich Globale Entwicklung. Cornelsen, Bonn, 2016.
- [Pr08] Programm Transfer-21: , Lernen für die Zukunft - Definition von Gestaltungskompetenz und ihrer Teilkompetenzen, Jul 2008. Abrufbar über <http://www.transfer-21.de/indexb4c1.html?p=222>.
- [Sc08] Schulte, Carsten: Duality Reconstruction - Teaching Digital Artifacts from a Socio-technical Perspective. In: Proceedings of the 3rd international conference on Informatics in Secondary Schools - Evolution and Perspectives: Informatics Education - Supporting Computational Thinking. Springer-Verlag, S. 110–121, 2008.
- [Ti63] Tinbergen, Nikolaas: On aims and methods of ethology. Zeitschrift für Tierpsychologie, 20(4):410–433, 1963.
- [UN15] Resolution 70/1 – Transformation unserer Welt: die Agenda 2030 für nachhaltige Entwicklung. Vereinte Nationen, New York, 2015.

AI Unplugged – Wir ziehen Künstlicher Intelligenz den Stecker

Stefan Seegerer¹, Annabel Lindner¹, Ralf Romeike²

Abstract: Aufgrund seiner großen Bedeutung in Medien, StartUp-Welt und der Digitalstrategie der Bundesregierung wird das Thema Künstliche Intelligenz (KI) auch für die Schule zunehmend relevant. Bisher sind Ansätze, die KI abseits von Programmierung für Schülerinnen und Schüler erlebbar machen, rar. Um dem zu begegnen, wurde eine Sammlung verschiedener Unplugged Aktivitäten rund um KI entwickelt, die das Thema greifbar macht und auch als Unterrichtssequenz verwendet werden kann. Unplugged stellt Aktivitäten bereit, die Ideen und Konzepte der Informatik ohne Computer ergründen. Die Aktivitäten beleuchten wichtige Konzepte des Themas KI und ermöglichen, der Künstlichen Intelligenz zugrunde liegende Ideen zielgruppengerecht zu vermitteln. Zudem bieten sie Ansatzpunkte, um gesellschaftliche Fragen der KI zu diskutieren. Dieser Beitrag beschreibt die Aktivitäten und deren theoretische Hintergründe, skizziert einen möglichen Unterrichtsverlauf und schildert konkrete Praxiserfahrungen von AI Unplugged.

Keywords: Künstliche Intelligenz; CS Unplugged; Maschinelles Lernen; Unterrichtsaktivitäten

1 Einleitung

Vermutlich bestimmt kein anderes informatisches Thema die aktuelle Berichterstattung so stark wie die Künstliche Intelligenz (KI, engl. AI). KI findet in vielen Bereichen Anwendung: Wir sprechen mit Künstlichen Intelligenzen in Form von Siri, Cortana oder Alexa, erhalten „intelligente“ Produktempfehlungen beim Online-Shopping oder lesen computergenerierte Texte. Immer mehr Softwareprodukte werden als KI-gestützt beworben. Auch die Bundesregierung widmet sich in einem aktuellen Strategiepapier dem Thema KI [Bu18]. Diese gesellschaftliche Bedeutung macht das Thema zukünftig auch für die Schule relevant, gerade vor dem Hintergrund, dass laut einer aktuellen Umfrage 50 % der Deutschen keine genaue Vorstellung von KI haben [Pr18]. Aufgrund der Komplexität des Themas ist ein werkzeuggestützter Zugang jedoch erst, wenn überhaupt, für höhere Altersstufen geeignet. Unplugged Aktivitäten können die Beschäftigung mit dem Thema Künstliche Intelligenz erleichtern und von Lehrerinnen und Lehrern ebenso wie von Schülerinnen und Schülern, auch bei geringerem Alter der Lernenden, als Einstieg in die Thematik verwendet werden.

¹ Friedrich-Alexander-Universität Erlangen-Nürnberg, Didaktik der Informatik, Martensstraße 3, 91058 Erlangen, stefan.seegerer@fau.de, annabel.lindner@fau.de

² Freie Universität Berlin, Didaktik der Informatik, Königin-Luise-Str. 24-26, 14195 Berlin, ralf.romeike@fu-berlin.de

2 Künstliche Intelligenz

Künstliche Intelligenz existiert als Teilgebiet der Informatik bereits seit 1956 [Mc06], praktische Relevanz hat das Thema jedoch erst durch die Verfügbarkeit entsprechender Rechenkapazitäten bekommen. Der Begriff Künstliche Intelligenz stellt vor allem eine Sammelbezeichnung für verschiedene Technologien und Verfahren dar, wobei hier klassische Themen der KI und „neuere“ Ansätze unterschieden werden können. Zwischen diesen beiden Richtungen steht insbesondere im Teilbereich des Maschinellen Lernens ein (Paradigmen)Wechsel von symbolischer Repräsentation, wie sie in Regelbasierten Systemen zur Anwendung kommt, hin zu sub-symbolischer Wissensdarstellung, wie sie Neuronale Netze nutzen und welche keine explizite Darstellung der erlernten Lösung erlaubt [La11].

Der Begriff des Maschinellen Lernens steht dabei fast schon synonym für diese „neueren“ Ansätze. Er beschreibt die Fähigkeit künstlicher Systeme, aus großen Datenmengen Muster und zugrundeliegende Regeln abzuleiten. Ergebnis dieses Lernprozesses ist dann ein Modell, das für die erfolgreiche Bearbeitung unbekannter Daten oder Probleme eingesetzt werden kann. Eine Technologie, die derartige Lernprozesse erlaubt, sind künstliche neuronale Netze. Hierbei handelt es sich um Netzwerke aus künstlichen Neuronen, die in ihrer Funktion den biologischen Neuronen des menschlichen Gehirns nachempfunden sind. Die künstlichen Neuronen sind innerhalb des Netzwerks miteinander über Kanten verbunden. Sie nehmen Information auf, verarbeiten diese und geben sie anschließend innerhalb des Netzes weiter.

In Bezug auf den Lernprozess künstlicher Systeme lassen sich drei grundlegende Typen des Lernens unterscheiden, die zum Trainieren eines Modells eingesetzt werden. Einerseits das Überwachte Lernen (Supervised Learning), bei dem bereits bekannt ist, welches Ergebnis von einem Computer auf eine bestimmte Eingabe hin erwartet wird. Die tatsächliche Ausgabe des Computers wird mit dieser Erwartung verglichen und es werden Rückschlüsse gezogen, wie das Modell modifiziert werden muss, um Erwartung und Ausgabe anzugleichen. Im Gegensatz dazu steht Unüberwachtes Lernen (Unsupervised Learning), bei dem sich das Modell basierend auf der Ähnlichkeit von Eingaben selbständig verändert. Diese Art des Lernens kommt insbesondere dann zum Einsatz, wenn keine bereits klassifizierten Daten, die zum Training des KI-Systems verwendet werden können, vorhanden sind. Das Verstärkende Lernen (Reinforcement Learning) stellt den dritten grundlegenden Lerntyp dar. Hierbei werden die Reaktionen eines lernenden Agenten auf bestimmte Eingabedaten bewertet. Basierend auf der erhaltenen Bewertung passt das System seine Reaktionen an. In modernen KI-Systemen nutzen diese Ansätze häufig eine sub-symbolische Abbildung von Wissen, d. h., dass das vom System erworbene Wissen nur implizit repräsentiert wird, z. B. durch unterschiedliche Kantengewichte in einem Neuronalen Netz, und konkrete Lösungsmuster nicht ersichtlich sind. Demgegenüber steht die symbolische KI mit einer expliziten Darstellung des Wissens und der Anwendung von Logik als zentralem Prinzip. Typische Themenbereiche dieser klassischen KI sind beispielsweise Suchverfahren, Planen, Wissensrepräsentation (u. a. mit Entscheidungsbäumen) und Inferenz unter der Verwendung von Logik. [Er13]

3 Künstliche Intelligenz als Thema informatischer Bildung

Mit der steigenden gesellschaftlichen Relevanz des Themas Künstliche Intelligenz, welche vor allem mit großen Fortschritten im Bereich Maschinelles Lernen zusammenhängt, wird KI nun auch vermehrt in Bildungskontexten diskutiert und thematisiert, etwa in CS4All Kursen an Hochschulen [SR18]. Zunehmend gewinnt das Thema aber auch bei der Gestaltung von Schulcurricula an Bedeutung (z.B. [CS17] oder [In18]), so hat beispielsweise China KI landesweit zum Unterrichtsinhalt in allgemeinbildenden Schulen gemacht [YC18]. Um eine Vereinheitlichung der curricularen Inhalte zu erreichen, wurden zudem in Anlehnung an die Big Ideas of K12 Computing [Be18] bereits Vorschläge für „Big Ideas“ der Künstlichen Intelligenz gemacht [To19]. Auch ein Konzept von Kandlhofer et al., das mit der Vermittlung von KI Grundlagen bereits im Kindergarten beginnt, zielt auf die zugrundeliegenden Konzepte und eine *AI Literacy* ab [Ka16].

Gerade zu Themen der klassischen KI existiert bereits eine Reihe von Lernmaterialien, beispielsweise bietet Informatik im Kontext (IniK) eine Unterrichtsreihe zum Thema Chatbots, welche mit Weizenbaums ELIZA arbeitet [WH08]. Aber auch sub-symbolischen Ansätze werden in verschiedenen Projekten für den Unterricht aufbereitet. Kahn et al. präsentieren z. B. ein Konzept, das es Schülerinnen und Schülern erlaubt, verschiedene KI-Dienste in der Programmierumgebung Snap! zu verwenden [Ka18], und Google stellt eine Sammlung von Experimenten bereit, mit denen Lernende KI ausprobieren können³. Auch *Machine Learning for Kids* bietet Online-Demos, in denen Schülerinnen und Schüler Klassifikationsmodelle trainieren und diese im Anschluss in Scratch verwenden können⁴. Daneben existieren Ansätze, die Roboter zur Vermittlung von KI-Themen einsetzen⁵.

Derartige Angebote decken häufig jedoch nicht das ganze Feld der Künstlichen Intelligenz ab, sondern fokussieren einzelne Teilbereiche, sodass es bisher nur wenige Ansätze gibt, wie das Feld der Künstlichen Intelligenz in seiner Breite und mit einem umfassenden Unterrichtskonzept schüler nah vermittelt werden kann. Zudem wird in vielen Projekten vor allem eine anwendungsorientierte Perspektive eingenommen, insbesondere wenn moderne Ansätze, wie Neuronale Netze thematisiert werden. Die zugrundeliegenden Konzepte der Künstlichen Intelligenz sind in der reinen Anwendungssituation allerdings schwer zu erfassen, sodass KI-Systeme bei derartigen Vermittlungskonzepten eine Blackbox bleiben. Um auch die technologische und gesellschaftlich-kulturelle Perspektive einzunehmen, ist ein Blick auf zugrundeliegende Ideen und Konzepte notwendig (vgl. [Br16]).

Die entwickelten Unplugged Aktivitäten zielen aus diesem Grund darauf ab, die der Künstlichen Intelligenz zugrundeliegenden Konzepte zugänglich zu machen, ohne dabei die Breite des Feldes zu vernachlässigen. Gleichzeitig wird auf eine stark formalisierte, mathematische Darstellung verzichtet, welche den Zugang für Schülerinnen und Schüler maßgeblich erschweren würde.

³ <https://experiments.withgoogle.com/>

⁴ <https://machinelearningforkids.co.uk/>

⁵ z.B. Cozmo Link (<https://www.anki.com/de-de/cozmo>)

4 AI Unplugged Aktivitäten

CS Unplugged stellt verschiedene Aktivitäten bereit, die Lernenden jeden Alters Ideen und Konzepte der Informatik näher bringen. Es geht darum, Konzepte der Informatik enaktiv zu begreifen. CS Unplugged Aktivitäten werden seit mittlerweile gut 30 Jahren im Kontext informatischer Bildung verwendet [BRC12], der erfolgreiche Einsatz von Unplugged-Aktivitäten ist dabei in allen Altersstufen möglich. So werden entsprechende Aktivitäten etwa in einer Sequenz für die Einführung in die Programmierung in der Grundschule [GFH17], in außerschulischen Lernlaboren [GGH15] oder in der Erwachsenenbildung eingesetzt [GHS12]. Im Bereich KI gibt es dazu bisher wenig Material, obwohl sich Unplugged Aktivitäten für diesen Bereich aufgrund seiner Komplexität und Vielseitigkeit gut eignen. Im Folgenden werden Ablauf und theoretischer Hintergrund von fünf AI Unplugged Aktivitäten vorgestellt, welche verschiedene elementare Konzepte hinter KI-Systemen verdeutlichen und sich an den Kriterien von Nishida et al. [Ni09] orientieren.



Abb. 1: Beispiel einer Aktivität zur Veranschaulichung von Klassifikationsstrategien

Aktivität 1: Klassifikation mit Entscheidungsbäumen – Das Gute-Äffchen-Böse-Äffchen-Spiel. Wie trifft ein Computer selbstständig Entscheidungen? Wie entscheidet ein Computer, ob eine Person sportlich ist, einen Kredit erhalten sollte usw.? Solche Klassifikationen sind eine häufige Anwendung von KI. In dieser Aktivität haben Schülerinnen und Schüler die Möglichkeit, selbst ein Klassifikationsmodell mithilfe eines Entscheidungsbaumes zu erstellen (siehe Abb. 1). Die Schülerinnen und Schüler untersuchen, wie die Zugehörigkeit von Beispielen zu einer Kategorie zustande gekommen ist. Dazu entwickeln Sie in Paaren Kriterien, mit denen neue Elemente klassifiziert werden können. Im Anschluss werden die entstandenen Modelle mit neuen Beispielen getestet und die Genauigkeit der Vorhersage bestimmt. Aus den verschiedenen Modellen wird dann das beste Modell ausgewählt.

Aktivität 2: #deeplearning – Erkennen von Bildern mit Neuronalen Netzen. Wie kann ein Computer eigentlich Dinge erkennen? Wie entscheidet ein Computer, ob auf einem Foto ein Hund, eine Katze oder eine Maus abgebildet ist? Wie kann er Gebäude von Menschen unterscheiden? Gegenstände oder Objekte anhand ihrer Form bzw. ihres Äußeren

zu erkennen ist für Menschen sehr leicht. Für den Computer, der beispielsweise in einem selbstfahrenden Auto die Objekte in seiner Umgebung erkennen muss, stellt dies jedoch eine komplexe Aufgabe dar. In dieser Aktivität haben Kinder und Jugendliche die Möglichkeit, selbst nachzuvollziehen, wie Computer den Inhalt von Bildern erkennen. Zudem wird der Begriff Deep Learning, der den Einsatz von vielschichtigen, komplexen Neuronalen Netzen in KI-Systemen bezeichnet, aufgegriffen.

Aktivität 3: Reinforcement Learning – Schlag das Krokodil. Computer, die Schach spielen und auch Menschen schlagen können, sind mittlerweile bekannt. Das chinesische Brettspiel Go galt hingegen lange Zeit als so komplex, dass nur Menschen es meistern können – so lange, bis Google mit AlphaGo menschliche Profi-Spieler das Fürchten lehrte. In dieser Aktivität wird beleuchtet, wie Computer selbst Strategien für Spiele entwickeln, obwohl sie nur die Spielregeln und somit nur die Liste möglicher Züge kennen. Je zwei Schülerinnen bzw. Schüler spielen eine Partie “Mini-Schach” gegeneinander. Dabei übernimmt eine Schülerin bzw. ein Schüler die Rolle eines “Papier“-Computers. Der Computer wählt seine Züge zunächst zufällig, lernt aber nach und nach mit einem Token-System, welche Züge ihm zum Sieg verhelfen und welche in einer Niederlage enden. Mit Hilfe der Strategie, die sich auf diese Weise entwickelt, wird der Computer mit der Zeit immer besser.

Aktivität 4: Back to the Roots – Krokodil-Schach und klassische KI. Die bisherigen Übungen beschäftigen sich intensiv mit lernenden KI-Systemen. Das ist jedoch noch nicht alles, was die KI zu bieten hat: Die Ursprünge der KI liegen in der Logik und der Idee, Wissen mathematisch zu formalisieren und es auf diese Weise für Maschinen zugänglich zu machen. Welche Unterschiede es zwischen lernender KI und klassischen Ansätzen gibt und wo die Grenzen der Systeme liegen, wird in dieser Aktivität aufgezeigt. Dazu wird die vorangehende Reinforcement Learning Aktivität mit einem Expertensystem umgesetzt und so die stark unterschiedlichen Herangehensweisen verdeutlicht. Wie in Aktivität 3 spielen je zwei Schülerinnen bzw. Schüler eine Partie “Mini-Schach” gegeneinander. Dabei übernimmt eine Schülerin bzw. ein Schüler die Rolle eines “Papier“-Computers. Im Gegensatz zur vorherigen Aktivität lernt der Computer hier sein Verhalten jedoch nicht, sondern arbeitet nach Regeln der Logik. Wird die Aktivität direkt mit der vorangehenden Reinforcement Learning-Version kombiniert, bietet es sich an, dass ein Rollentausch stattfindet.

Aktivität 5: “Ach wie gut, dass niemand weiß, dass ich ein Computer bin!” - Der Turing-Test. Welches Verhalten muss eine Maschine zeigen, um als intelligent zu gelten? Was bedeutet Künstliche Intelligenz genau? Diese Fragen beschäftigen Forscher seit den Anfängen der Künstlichen Intelligenz. Alan Turing hat 1950 mit dem Turing-Test ein Verfahren entwickelt, wie man feststellen könnte, ob eine Maschine intelligent ist. Diese Aktivität, entlehnt aus den ursprünglichen Unplugged Materialien [BWF98], stellt mit Schülerinnen und Schülern den Turing-Test nach und soll zur Diskussion anregen, ob

Computer tatsächlich so etwas wie Intelligenz zeigen können. Zudem wird deutlich, wie leicht man durch sorgfältig gewählte Beispiele von “Intelligenz” von einer Maschine in die Irre geführt werden kann.

5 Einbindung in den Unterricht

Ziel des Unterrichts ist das Entdecken und Ergründen der zugrundeliegenden Konzepte im Bereich Künstlicher Intelligenz. Dabei explorieren die Schülerinnen und Schüler spielerisch verschiedene Phänomene rund um das Thema. Denkbar ist sowohl die Integration mehrerer Aktivitäten in eine Einheit als auch einzelne Unterrichtsstunden zu ausgewählten Aspekten. Im Folgenden wird ein möglicher Ablauf der Sequenz grob skizziert, anschließend eine Stunde beispielhaft genauer ausgeführt. Zielgruppe sind insbesondere Schülerinnen und Schüler der Sekundarstufe 1 und 2, obwohl einige Aktivitäten auch (eventuell angepasst) in anderen Lernszenarien eingesetzt werden können.

5.1 Unterrichtssequenz

Die im Folgenden beschriebene Sequenz adressiert mehrere Kompetenzbereiche, insbesondere aus den Inhaltsbereichen Informatiksysteme und Informatik, Mensch und Gesellschaft, sowie den Prozessbereichen Begründen und Bewerten und Modellieren und Implementieren, allerdings kommen auch die anderen Inhalts- und Prozessbereich der Bildungsstandards Informatik in der Sequenz in Ansätzen zum Tragen [Pu08]. Die in Tabelle 1 überblicksartig dargestellte Sequenz verdeutlicht, dass Künstliche Intelligenz vielfältige Formen annehmen kann und aktuelle KI-Systeme auf stark spezialisierte Anwendungen begrenzt sind. Zudem werden gesellschaftlich-soziale Auswirkungen von KI-Systemen thematisiert und illustriert, dass Künstliche Intelligenz weniger „magisch“ ist, als man vielleicht zunächst annehmen würde.

Zu Beginn der Sequenz wird das Thema KI anhand der Klassifikation von Objekten eingeführt, welche etwa bei selbstfahrenden Autos von Bedeutung ist. Nach dem Einstieg dient die Bilderkennung als Kontext für Maschinelles Lernen mit neuronalen Netzen, ehe diese bisher thematisierten, neueren Ansätze klassischen Ansätzen gegenübergestellt werden. Dazu wird anhand eines Spiels der Unterschied zwischen Lernverfahren und Regelbasierten Systemen herausgestellt und die Breite des Themenfeldes Künstliche Intelligenz angedeutet. Zum Abschluss wird das Thema reflektiert und die Frage in den Mittelpunkt gestellt, inwieweit Computer wirklich intelligent sein können.

5.2 Unterrichtsstunde

Exemplarisch wird die erste Stunde der Sequenz (45 Minuten) geschildert, diese kann auch eigenständig zum Einsatz kommen. Die Aktivität *Klassifikation mit Entscheidungsbäumen*:

Stunde	Stundenthema	Beschreibung	AI Unplugged
1	Einstieg in den Themenbereich KI	Die Schülerinnen und Schüler explorieren das Konzept Maschinellen Lernens und beispielhafte Auswirkungen auf die Gesellschaft.	Aktivität 1
2-3	Neuronale Netze und Deep Learning	Anhand der Bildklassifizierung mit Neuronalen Netzen ergründen die Schülerinnen und Schüler ein weiteres bekanntes Gebiet der KI und analysieren die Funktionsweise Neuronaler Netze. Sie lernen außerdem, was hinter dem Begriff Deep Learning steckt.	Aktivität 2
4-5	Maschinelles Lernen und klassische KI	Anhand von zwei Strategien für dasselbe Spiel vergleichen die Schülerinnen und Schüler „neue“ und klassische Ansätze der KI.	Aktivität 3 und 4
6	Was ist künstliche Intelligenz?	Mithilfe des Turing-Tests gehen Schülerinnen und Schüler der Frage nach, ob Computer wirklich Intelligenz zeigen können und reflektieren ethische Aspekte des Themas KI.	Aktivität 5

Tab. 1: Exemplarischer Unterrichtsverlauf

Das Gute-Äffchen-Böse-Äffchen Spiel, die in dieser Stunde zum Einsatz kommt, erlaubt das Erstellen von Klassifikationsmodellen unterschiedlicher Komplexität. Schülerinnen und Schüler können nach dieser Stunde zum einen erklären, wie ein Computer anhand einer bereits klassifizierten Beispielmenge lernt, Elemente erfolgreich zu klassifizieren, zum anderen können sie den Trainingsvorgang eines Klassifikationssystems beschreiben und die Bedeutung von Trainings- und Testdaten herausstellen. Des Weiteren können reale Situationen beschrieben werden, in welchen Entscheidungen von Künstlicher Intelligenz problematisch sein können. Die Stunde stellt grundlegende Aspekte Maschinellen Lernens vor und beleuchtet gleichzeitig gesellschaftliche Problemstellungen, die aus der Verwendung von KI-Systemen für Klassifikationsaufgaben entstehen können. Folgende Gliederung der Unterrichtsstunde ist dabei möglich:

- Zum Einstieg spielen die Schülerinnen und Schüler zunächst eine einfache Version des Äffchen-Spiels, um das Spielprinzip zu verdeutlichen.
- Die Ergebnisse des ersten Durchgangs werden ausgewertet und das beste Klassifikationsmodell ausgewählt.
- In einer weiteren Spielrunde kommt nun eine komplexere Spielversion zum Einsatz.
- Auch dieser Durchgang wird ausgewertet und das beste Klassifikationsmodell bestimmt.
- Mithilfe einer Lernaufgabe, die den Abstraktionsprozess vom konkreten Spiel hin

zum allgemeinen maschinellen Lernprozess ermöglicht, wird die Aktivität dekontextualisiert.

- Abschließend werden die Ergebnisse der Lernaufgabe diskutiert und auf das Beispiel der Teachable Machine⁶ übertragen.

Ebenso wie diese Aktivität eignen sich auch die anderen Aktivitäten der Sequenz, etwa *Reinforcement Learning: Schlag das Krokodil* und *Back to the roots: Krokodil-Schach und klassische KI* für den isolierten Einsatz in einzelnen Unterrichtsstunden.

6 Resümee und Ausblick

AI Unplugged wurde in einem Schulversuch mit einer neunten und einer zehnten Klasse der Realschule erprobt. Obgleich die Schülerinnen und Schüler eine rudimentäre Vorstellung des Begriffs Künstliche Intelligenz hatten, verfügten sie doch über keine inhaltlichen Vorkenntnisse in diesem Themenbereich. So verbinden viele Schülerinnen und Schüler den Begriff vor allem mit selbstständig agierenden, lernfähigen Robotern.

Bei der Durchführung erfassten die Schülerinnen und Schüler die Funktionsprinzipien der Aktivitäten schnell und umfassend, bei der Arbeit in Teams waren die Schülerinnen und Schüler konzentriert und mit Spaß bei der Sache. Als es beispielsweise galt, Regeln für die Klassifikation von Äffchen aufzustellen (Aktivität 1), entwickelte sich ein Wettbewerb darum, das beste Klassifikationsmodell zu erstellen. Zudem wurde rege diskutiert, wie fair ein derartiges Klassifikationsmodell sein kann. Bei Fragen und Problemen besprachen sich die Schülerinnen und Schüler mit ihren Mitschülerinnen und Mitschülern und erarbeiteten Lösungen im Team ohne die Hilfe der Lehrkraft zu beanspruchen. Auch Schülerinnen und Schüler, die dem Konzept „Informatik ohne Computer“ am Anfang skeptisch gegenüber standen, konnten durch die Aktivitäten sehr gut motiviert werden und zeigten anschließend großes Interesse am Thema.

Im Rahmen der anschließenden Dekontextualisierung war bei den Schülerinnen und Schülern ein deutlicher Erkenntniszuwachs feststellbar: Nach der Durchführung von AI Unplugged Aktivitäten waren die Schülerinnen und Schüler in der Lage, den Begriff des Maschinellen Lernens zu erläutern und konnten auf dieser Basis zudem gesellschaftliche Implikationen Künstlicher Intelligenz diskutieren. Sie erkannten die Zusammenhänge zwischen individuellen Aktivitäten und den zugrundeliegenden Konzepten der Künstlichen Intelligenz sehr schnell und konnten Parallelen zu Phänomenen, die sie aus dem Alltag kannten (z. B. Captchas, Google Bildersuche etc.), ziehen. Dies sorgte für Aha-Effekte, teilweise sogar schon während der Aktivitäten, und ermöglichte den Schülerinnen und Schülern die Grenzen solcher Systeme zu erkennen.

⁶ <https://teachablemachine.withgoogle.com/>

Es zeigt sich, dass die AI Unplugged Aktivitäten gut geeignet sind, um das komplexe Thema Künstliche Intelligenz umfassend darzustellen. Sie erlauben den handlungsorientierten Zugang zum Thema KI, ohne jedoch auf einen technischen Rahmen angewiesen zu sein. Das Thema kann damit in reduzierter Komplexität vermittelt werden und bleibt gleichzeitig anschlussfähig für weitergehende Vertiefungen, die sich unterschiedlichster Softwarewerkzeuge bedienen können. Zudem bieten die Unplugged Aktivitäten auch Lehrkräften, die KI als Unterrichtsthema aufgrund der thematischen Komplexität und der technischen Hürden bei der Einrichtung von Programmbibliotheken usw. bisher kritisch gegenüber standen, eine gute Möglichkeit, das Thema in den schulischen Unterricht zu integrieren. Die Konzeption der Aktivitäten ermöglicht dabei sowohl den Einsatz als zusammenhängende Unterrichtssequenz zu Künstlicher Intelligenz als auch den individuellen Einsatz „zwischen durch“ oder als Ergänzung zu anderen Ansätzen und Zugängen zu KI.

Die Materialien sind unter Creative Commons Lizenz online abrufbar.⁷ Im Onlineangebot finden sich zudem weitere Materialien und Links zum Thema, die ebenfalls im Unterricht zum Einsatz kommen können.

Literaturverzeichnis

- [Be18] Bell, Tim; Tyman, Paul; Yehudai, Amiram et al.: The Big Ideas in Computer Science for K-12 Curricula. *Bulletin of EATCS*, 1(124), 2018.
- [Br16] Brinda, Torsten; Diethelm, Ira; Gemulla, Rainer; Romeike, Ralf; Schöning, Johannes; Schulte, Carsten: Bildung in der digitalen vernetzten Welt–Dagstuhl-Erklärung. URL: <https://www.gi.de/aktuelles/meldungen/detailansicht/article/dagstuhl-erklaerung-bildung-in-der-digitalen-ernetzten-welt.html> [15.01.2019], 2016.
- [BRC12] Bell, Tim; Rosamond, Frances; Casey, Nancy: Computer Science Unplugged and Related Projects in Math and Computer Science Popularization. In: *The Multivariate Algorithmic Revolution and Beyond*. Springer, Berlin, Heidelberg, S. 398–456, 2012.
- [Bu18] Bundesregierung: Strategie Künstliche Intelligenz der Bundesregierung. URL: https://www.bmbf.de/files/Nationale_KI-Strategie.pdf [30.1.2019], 2018.
- [BWF98] Bell, Tim; Witten, Ian; Fellows, Mike: *Computer Science Unplugged: Off-line activities and games for all ages*. Citeseer, 1998.
- [CS17] CSTA: About the CSTA K-12 Computer Science Standards. URL: <https://www.csteachers.org/page/standards> [15.01.2019], 2017.
- [Er13] Ertel, Wolfgang: *Grundkurs Künstliche Intelligenz*. Springer Fachmedien Wiesbaden, 2013.
- [GFH17] Geldreich, Katharina; Funke, Alexandra; Hubwieser, Peter: Willkommen im Programmierzirkus - Ein Programmierkurs für Grundschulen. In: *Informatische Bildung zum Verstehen und Gestalten der digitalen Welt, Proceedings der 17. GI-Fachtagung Informatik und Schule, INFOS 2017*, 13.-15. September 2017, Oldenburg. S. 327–334, 2017.

⁷ <https://ddi.cs.fau.de/schule/ai-unplugged>

- [GGH15] Gallenbacher, Jens; Gose, Karola; Heun, Dominik: Gestrandet auf der Schatzinsel - Schätze heben mit Informatik in der Grundschule. In: Informatik allgemeinbildend begreifen, INFOS 2015, 16. GI-Fachtagung Informatik und Schule, 20.-23. September 2015, Darmstadt, Germany. S. 101–109, 2015.
- [GHS12] Garcia, Daniel; Harvey, Brian; Segars, Luke: CS principles pilot at University of California, Berkeley. *ACM Inroads*, 3(2):58–60, 2012.
- [In18] International Society for Technology in Education (ISTE): Bold New Program Helps Teachers and Students Explore the Power of AI. URL: <https://www.iste.org/explore/articleDetail?articleid=2229> [15.01. 2019], 2018.
- [Ka16] Kandhofer, Martin; Steinbauer, Gerald; Hirschmugl-Gaisch, Sabine; Huber, Petra: Artificial intelligence and computer science in education: From kindergarten to university. In: 2016 IEEE Frontiers in Education Conference (FIE). IEEE, S. 1–9, 2016.
- [Ka18] Kahn, Ken; Megasari, Rani; Piantari, Erna; Junaeti, Enjun: AI Programming by Children using Snap! Block Programming in a Developing Country. 2018.
- [La11] Langley, Pat: The changing science of machine learning. *Machine Learning*, 82(3):275–279, Feb 2011.
- [Mc06] McCarthy, John; Minsky, Marvin; Rochester, Nathaniel; Shannon, Claude: A proposal for the dartmouth summer research project on artificial intelligence, August 31, 1955. *AI magazine*, 27(4):12, 2006.
- [Ni09] Nishida, Tomohiro; Kanemune, Susumu; Idosaka, Yukio; Namiki, Mitaro; Bell, Tim; Kuno, Yasushi: A CS unplugged design pattern. *ACM SIGCSE Bulletin*, 41(1):231–235, 2009.
- [Pr18] Preuss, Susanne: Die Hälfte der Deutschen weiß nicht, was KI ist. URL: <https://www.faz.net/aktuell/wirtschaft/kuenstliche-intelligenz/die-haelfte-der-deutschen-weiss-nicht-was-ki-ist-15898006.html> [15.01. 2019], 2018.
- [Pu08] Puhlmann, Hermann; Brinda, Torsten; Fothe, Michael; Friedrich, Steffen; Koerber, Bernhard; Röhner, Gerhard; Schulte, Carsten: Grundsätze und Standards für die Informatik in der Schule. *Bildungsstandards Informatik für die Sekundarstufe I. Beilage zu LOG IN*, (150/151), 2008.
- [SR18] Seegerer, Stefan; Romeike, Ralf: Was jeder über Informatik lernen sollte – Eine Analyse von Hochschulkursen für Studierende anderer Fachrichtungen. In: HDI 2018. Potsdam, S. 13–28, 2018.
- [To19] Touretzky, David; Gardner-McCune, Christina; Martin, Fred; Seehorn, Deborah: Envisioning AI for K-12: What should every child know about AI? In: „Blue sky talk“ at the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19). 2019.
- [WH08] Witten, Helmut; Hornung, Malte: Chatbots Teil 1: Einführung in eine Unterrichtsreihe zu Informatik im Kontext (IniK). *LOG IN*, 28(154/155):51–60, 2008.
- [YC18] Yu, Yanfang; Chen, Yuan: Design and Development of High School Artificial Intelligence Textbook Based on Computational Thinking. *Open Access Library Journal*, 5(09):1, 2018.

Eine Unterrichtssequenz zum Einstieg in Konzepte des maschinellen Lernens

Kerstin Strecker¹, Eckart Modrow²

Abstract: In diesem Beitrag soll ein Weg aufgezeigt werden, um Grundprinzipien maschinellen Lernens so zu reduzieren, dass sie in der Schule praktisch umzusetzen sind. Dafür werden LEGO-NXT-Roboter so programmiert, dass sie „aus Erfahrungen lernen“.

Keywords: LEGO-Roboter, maschinelles Lernen, Entmystifizierung des „schlau“ Computers, Konditionierung, Hebb-Regel

1 Entmystifizierung des „schlau“ Computers

Grundschüler³, denen ein Computer Rechenaufgaben stellt und ggf. ein positives Feedback gibt, wurden gefragt, weshalb der Computer immer die korrekte Lösung kennt oder ob er sich auch irren kann. Die einvernehmliche Antwort war, dass Computer sich nicht irren, weil sie „schlau“ sind, zumindest schlauer als die Schülerinnen und Schüler. Erfahren sie später in der Sekundarstufe, wenn sie z.B. mit digitaler Elektronik einen Parallel-Addierer im Informatikunterricht gebaut haben, dass sich ein Computer deshalb nicht „verrechnen“ kann, weil die Leitungen starr gesteckt sind und deterministisch immer derselben Funktionalität folgen, dann erscheint ihnen diese Maschine sehr viel weniger „schlau“. Stattdessen könnten sie jetzt einen Schachcomputer für „intelligent“ halten, bis sie lernen, dass in einem Entscheidungsbaum einfach sehr viele Möglichkeiten durchprobiert werden und dass eine der möglichen Aktionen gewählt wird, die zu einem positiven Ergebnis (Gewinn des Spiels) führt. Auch hier folgt der Rechner einem deterministischen Algorithmus. In beiden Beispielen erscheinen die durch einfache Algorithmen gesteuerten Systeme nur deshalb „intelligent“, weil die Nutzer von deren Funktionsweise nichts wissen (vgl. [Br93]). Allerdings können beide Systeme noch nichts „dazulernen“.

Viele Anwendungen auf dem Smartphone der Lernenden scheinen aber zu lernen. Die Wortergänzung passt sich immer mehr den speziellen Ausdrücken der Schülerinnen und Schüler an, ebenso die Autovervollständigung und weitere. Erst wenn Schülerinnen und Schüler im Informatikunterricht mit Datenbanken arbeiten, verstehen sie, dass es sich hier um einfache Erweiterungen der Datenbasis handelt, also um INSERT-Befehle unter

¹ Universität Göttingen, Didaktik der Informatik, Goldschmidtstr. 7, 37077 Göttingen, kerstin.strecker@informatik.uni-goettingen.de

² Universität Göttingen, Didaktik der Informatik, Goldschmidtstr. 7, 37077 Göttingen, emodrow@informatik.uni-goettingen.de

³ Es wurde eine zweite Klasse mit 21 Grundschulern befragt. Aus dem Ergebnis kann keine allgemeingültige Aussage abgeleitet werden.

bestimmten Bedingungen. Der scheinbare Wissenserwerb wird darauf reduziert, dass sich eine Datenbasis durch das Feedback der Benutzer erweitert.

Im Informatikunterricht lassen sich also viele scheinbar „intelligente“ Systeme entmystifizieren. Das gilt auch für die künstliche Intelligenz. Dieser viel diskutierte Bereich der Informatik wird über kurz oder lang die Schulinformatik erreichen. Er erscheint sehr viel weniger mystisch, wenn gezeigt wird, dass die aktuellen Feed-Forward-Systeme im Wesentlichen Funktionen anwenden, deren Parameter durch viele Beispiele angepasst werden. Wir beschreiben hier einige Beispiele, in denen die Schülerinnen und Schüler lernen, dass auch „neuronal“ Systeme einfach nur deterministischen, vorgegebenen Regeln folgen.

2 Bestärkendes Lernen durch Zuordnung Wahrnehmung - Aktion

Bei dem Ansatz des bestärkenden Lernens „lernt“ ein System in einer Lernphase oder Trainingsphase anhand von Beispielen. Dazu ist ein Feedback in der Lernphase notwendig. Wir nutzen folgendes Beispiel dieses Ansatzes dazu, um die Begriffe der Wahrnehmung und Aktion einzuführen.

Wir betrachten ein System, welches mit Hilfe von Sensoren Wahrnehmungen als Eingabe erhält. Dabei sei eine Wahrnehmung eine Kombination der aktuellen Werte aller Sensoren. Gibt es beispielsweise einen Sensor, der hell und dunkel unterscheiden kann und einen Sensor, der in drei Abstufungen Lautstärke unterscheiden kann (leise, mittlere_Lautstärke, laut), dann verfügt das System über sechs verschiedene Wahrnehmungen (hell-leise, hell-mittlere_Lautstärke, hell-laut, dunkel-leise, dunkel-mittlere_Lautstärke, dunkel-laut).

In unserem Programmierbeispiel betrachten wir einen kleinen schwarzen Ball, der sich in einem farbigen Gebiet befindet (Abbildung 1). Dabei kann der Benutzer den grünen Balken mit der Maus von links nach rechts bewegen, ähnlich einem Schläger im Spiel „Pong“.

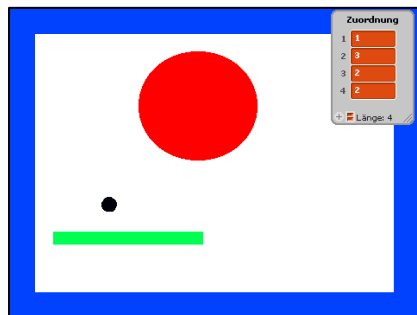


Abbildung 1 : Ball in farbigem Gebiet

Unser Ball soll vier Wahrnehmungen unterscheiden können, nämlich ob er die Farben Rot (1), Blau (2), Grün (3) oder nur Weiß (4) berührt. Weiterhin kann dieser Ball drei verschiedene Aktionen ausführen: er kann langsam vorwärtsfliegen (1), abprallen (2) oder schnell vorwärtsfliegen (3). Eine Zuordnungsliste zwischen Wahrnehmung (1,2,3,4) und Aktion (1,2,3) ist anfangs leer. Der Ball prüft ständig seine Wahrnehmung und führt die Aktion aus, die er in der Zuordnungsliste für diese Wahrnehmung findet. Findet er keine Zuordnung, führt der Ball Aktion 1 aus und „fragt“ anschließend, ob das richtig war. Verneint der Benutzer, macht der Ball die Aktion rückgängig und führt Aktion 2 aus. Auch jetzt fordert er das Feedback des Benutzers. Wir gehen davon aus, dass eine der ausgeführten Aktionen ein positives Feedback bekommt. Diese Aktion wird in der Zuordnungsliste gespeichert. Am Ende der Trainingsphase hat der Roboterball jeder Wahrnehmung eine Aktion zugeordnet, die er fortan ausführt. Damit ist das Spiel ganz unterschiedlich konfigurierbar. Der Ball könnte bei Grün (am Schläger) abprallen und bei Blau (am Rand) und bei Rot seine Geschwindigkeit erhöhen. Er könnte aber auch so trainiert werden, dass er bei Rot abprallt und bei Grün seine Geschwindigkeit erhöht, usw.

3 Systeme mit „Neuron“

In dem obigen Beispiel „lernte“ unser Ball sein Verhalten durch Feedback von außen. Jetzt wollen wir Systeme betrachten, die ihre Konfiguration selbst so verändern, dass sie in ihrer Umwelt optimal (re-)agieren (vgl. [MP47]). Unser Beispiel entspricht der klassischen Konditionierung des Pawlowschen Hundes [PH19].

Wir modellieren ein Neuronales Netz als gerichteten Graphen, wobei die Knoten Zellkörper repräsentieren und die Kanten die Verbindungen zwischen den Neuronen. Im Inneren eines Knotens wird ein Schwellenwert notiert. Die Kanten sind gewichtet. Es existieren außerdem Neuronen, die Signale aus der Außenwelt erhalten. Wir verwenden nur binäre Signale (Null oder Eins). Die Ausgabefunktion eines Neurons ist folgendermaßen definiert: Ist die Summe aller Eingaben multipliziert mit den zugehörigen Kantengewichten größer oder gleich dem Schwellenwert, so ist die Ausgabe Eins, sonst Null. Abbildung 2-4 veranschaulicht die Modellierung.

Im Folgenden betrachten wir stets nur ein Neuron. Wir können das logische ODER bereits als erste Programmierübung auf dem NXT-Roboter implementieren. In der Schule hilft dies, um sich die Funktionsweise langsam zu erarbeiten, hier können wir den Code aus Platzgründen nicht wiedergeben, er ähnelt aber dem Code im nächsten Kapitel, nur ohne Änderung der Gewichte.

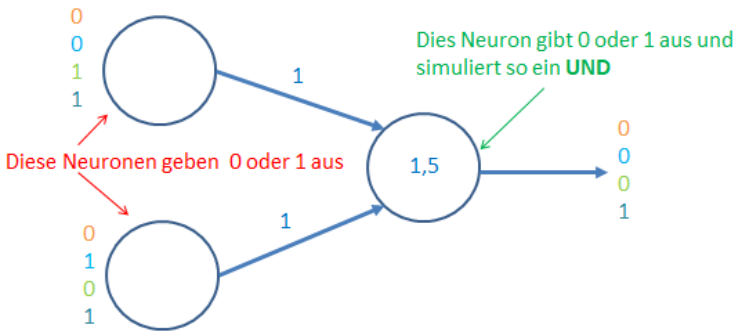


Abbildung 2: Beispiel UND

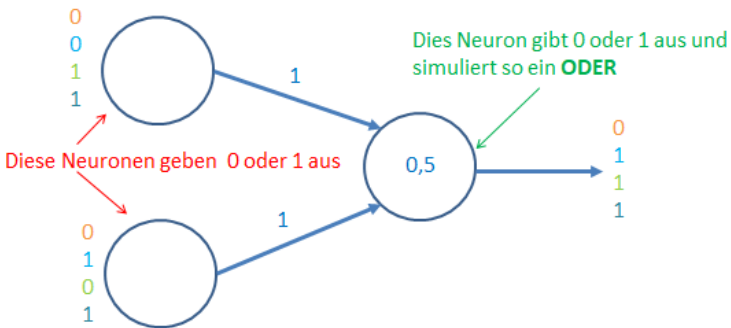


Abbildung 3 : Beispiel ODER

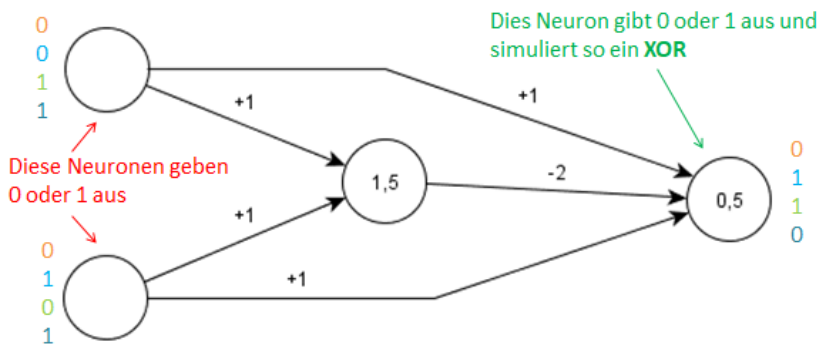


Abbildung 4 : Beispiel XOR

4 Der Roboter „lernt“

Unser Roboter soll lernen, aber wie lernt ein Neuronales Netz? Prinzipiell durch die Entwicklung neuer Verbindungen, Löschen bestehender Verbindungen, Ändern der Gewichtung (der Gewichte von Neuron i zu Neuron j), Anpassen der Schwellenwerte der Neuronen, Hinzufügen von Neuronen oder Löschen von Neuronen (vgl. [Wö06]). In unserem reduzierten Modell betrachten wir im Folgenden nur die Möglichkeit der Änderung der Kantengewichte. Dafür gibt es viele verschiedene Lernalgorithmen. Wir verwenden hier die Hebb-Regel, die besagt, dass das Gewicht zwischen zwei Neuronen dann erhöht wird, wenn beide Neuronen gleichzeitig aktiv sind, also beide Neuronen eine Eins am Ausgang liefern (vgl. [Ro96]). Die Erhöhung der Gewichte zwischen den Knoten i und j wird mit einem $+$ dargestellt.

Aktivität Knoten i	Aktivität Knoten j	Hebb-Regel
0	0	=
0	1	=
1	0	=
1	1	+

Abbildung 5 : Hebb-Regel

Betrachten wir unseren NXT-Roboter mit einem „Neuron“.

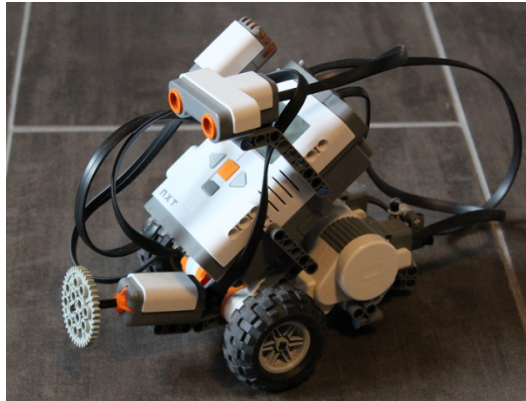


Abbildung 6 : unser Roboter mit Sensoren und Aktoren

Der Roboter besitzt zwei Motoren, mit deren Hilfe er fahren kann. Die beiden möglichen Aktionen sind „vorwärts“ oder „umdrehen“:

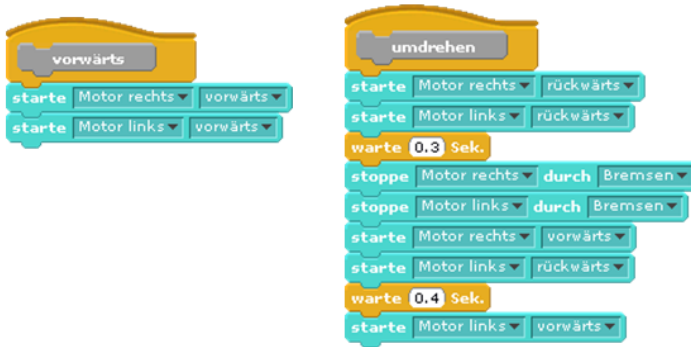


Abbildung 7 : Roboteraktionen

Unser Roboter fährt geradeaus. Liefert sein Neuron eine Eins, so dreht er um. Das Neuron hat nun drei Eingänge. Eingang 1 liefert Eins, wenn der Geräuschsensor laut registriert, Null sonst. Eingang 2 liefert Eins, wenn der Ultraschallsensor nah registriert, sonst Null. Eingang 3 ist Eins, wenn der Berührungssensor gedrückt ist und Null sonst.



Abbildung 8 : Wahrnehmungen des Roboters

Mit unserer Lernregel nach Hebb gilt: liefert eine der Eingaben den Wert Eins (ist also aktiv) und das Neuron liefert ebenfalls den Wert Eins, so erhöht sich das Kantengewicht zwischen Eingabe und Neuron. Für unser Neuron gelten anfangs folgende Kantengewichte:

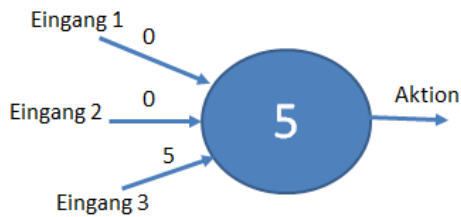


Abbildung 9 : Neuron des Roboters

Ist Eingang 3 aktiv (Berührsensor gedrückt), wird bereits der Schwellenwert erreicht und das Neuron feuert, der Roboter dreht um. Das „angeborene“ Verhalten ist also ein Umdrehen, wenn der Roboter z. B. gegen eine Wand fährt. Die anderen Sensoren beeinflussen die Aktivität des Neurons bisher nicht. Setzen wir nun folgenden Algorithmus um:

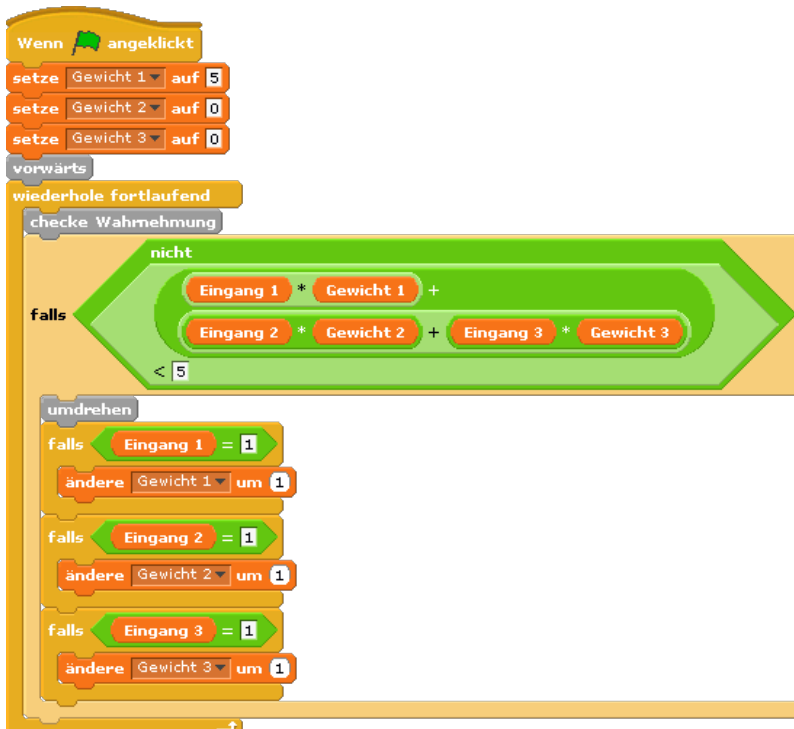


Abbildung 10 : Lernalgorithmus

Man kann Folgendes beobachten: Der Roboter fährt geradeaus. Wenn er gegen eine Wand fährt (Berührsensor löst aus), dreht er um und fährt wieder geradeaus. Das Ganze passiert mehrmals hintereinander. Nach einigen Kollisionen scheint der Roboter etwas „gelernt“ zu haben, denn er dreht schon um, bevor er die Wand berührt. Jedes Mal, wenn das Neuron aktiv ist, also bedingt durch den Berührsensor umdreht, liefert Eingang 2 auch eine Eins, weil der Ultraschallsensor registriert, dass etwas nah ist. Nach der Lernregel wird dadurch die Gewichtung zwischen Eingang 2 und dem Neuron erhöht. Irgendwann erreicht das Gewicht so den Wert Fünf. Jetzt dreht der Roboter bereits um, wenn Eingang 2 eine Eins liefert (eine Wand nah ist), weil die Summe den Schwellenwert erreicht und das Neuron feuert. Mit den Worten eines Schülers: Nachdem sich der Roboter ein paar Mal die Nase gestoßen hat, fährt er nicht mehr gegen eine Wand, sondern dreht vorher um⁴. Der Roboter ist konditioniert wie der Pawlowsche Hund.

5 Der „intelligente“ Ball

Steht kein Roboter zur Verfügung, kann man auch andere Systeme zum „Lernen“ bringen. Unser Beispiel ist das Spiel „Pong“ mit einem „intelligenten“ Ball. Unser Ball kann zwei verschiedene Aktionen ausführen: fliegen oder abprallen (Der Code ist aus Platzgründen weggelassen, ist aber wenig komplex).

Außerdem hat sein Neuron zwei Eingänge. Eingang 1 liefert nur dann eine Eins, wenn die Farbe Schwarz berührt wird, Eingang 2 nur dann eine Eins, wenn die Farbe Rot berührt wird. Auch hier lassen wir aus Platzgründen den Code weg, er ist aber ähnlich zu dem Befehl „checke Wahrnehmung“ des Roboters.

Unser Schläger hat drei verschiedene „Kostüme“. Er kann schwarz sein, rot oder rot-schwarz gestreift.



Abbildung 11 : drei verschiedene Kostüme

Wir implementieren analog zu unserem Roboter folgende Funktionalität (siehe Abbildung 11).

⁴ Natürlich wachsen die Gewichte immer weiter, was unnötig ist, wenn der Wert Fünf überschritten ist. Auch kann der Roboter nicht wieder „verlernen“. Hier finden die Schüler aber aller Erfahrung nach selbstständig individuelle Lösungen.

Was passiert? Ist der Schläger schwarz, kann man damit das ganz normale Pong spielen. Ist der Schläger nur rot, fällt der Ball hindurch und prallt nicht am Schläger ab. Wählt der Benutzer den rot-schwarz-gestreiften Schläger, so prallt der Ball ab (der schwarzen Farbe wegen). Er berührt dabei aber gleichzeitig auch rot. Der Ball „lernt“, auch bei Rot abzuprallen, denn wenn der zweifarbige Schläger eine längere Zeit verwendet wird und danach der nur rote Schläger, prallt der Ball auch an diesem Schläger ab.



Abbildung 12 : Lernalgorithmus

6 Fazit

In diesem Beitrag haben wir eine mehrfach erprobte Unterrichtssequenz aufgezeigt, die didaktisch sehr reduziert die Grundprinzipien maschinellen Lernens verdeutlicht und die praktisch umsetzbar ist. Dabei wurden NXT-Roboter so programmiert, dass sie scheinbar „aus Erfahrungen“ lernen. Diese Sequenz soll einen ersten Einblick in die Grundprinzipien neuronaler Netze ermöglichen. Hinzu kommt die Erkenntnis der Schülerinnen und Schüler, dass auch diese einfach nur deterministischen Regeln folgen und Algorithmen abarbeiten. Sehr viel größere Netze, die heute durch die hohe Rechenleistung der Rechner zu realisieren sind, verarbeiten auch sehr viel komplexere Situationen, arbeiten aber nicht prinzipiell anders (vgl. [Go16]). Die Frage, ob das

menschliche Gehirn prinzipiell mehr leisten kann oder nicht, bleibt (natürlich) unbeantwortet, erschließt sich den Schülerinnen und Schülern aber als Fragestellung.

Literaturverzeichnis

- [Br93] Valentin Braitenberg: "Vehikel – Experimente mit kybernetischen Wesen" Rowohlt-Verlag, 1993
- [Go16] Ian Goodfellow, Yoshua Bengio, Aaron Courville : "Deep Learning", MIT Press, 2016
- [MP47] W. McCulloch, W. Pitts: „How We Know Universals: the Perception of Auditory and Visual Forms”, Bulletin of Mathematical Biophysics, Vol. 9, 1947
- [PH19] https://de.wikipedia.org/wiki/Pawlowscher_Hund, Zugriff: 30.1.19
- [Ro96] Raul Rojas: "Neural Networks. A Systematic Introduction", Springer-Verlag, 1996
- [Wö06] P. Manoonpong, T. Gend, F. Wörgötter: „Exploring the dynamic walking range of the biped robot RinBot with an active upper-body component”, IEEE-RAS International Conference of humanoid robots, 2006

Kurzbeiträge

Was haben Staubsaugerroboter, Spielekonsolen und Smartphones gemeinsam?

Schülervorstellungen von Teil-Ganzes-Beziehungen in Informatiksystemen

Nils Pancratz¹, Ira Diethelm¹

Abstract: In diesem Beitrag wird eine Untersuchung vorgestellt, in der der Frage nachgegangen wird, welche mentalen Teil-Ganzes-Denkstrukturen Schülerinnen und Schüler bei der Erklärung von Informatiksystemen anwenden. Dazu wird Concept Mapping in leitfadengestützte Interviews, die mittels qualitativer Inhaltsanalyse ausgewertet werden, integriert. Die Ergebnisse einer ersten, pilotierenden Durchführung mit acht Achtklässlerinnen zeigen, dass entsprechende Systeme unter ihrer Oberfläche häufig als Blackboxes begriffen und zu ihrer Erklärung hauptsächlich sicht- und interagierbare Teile herangezogen werden.

Keywords: Denken in Teilen von Ganzen; mentale Ordnungssysteme; Wissensrepräsentationen; Schülervorstellungen; Didaktische Rekonstruktion; Concept Mapping

1 Motivation und theoretische Grundlagen

Kognitionswissenschaftliche Untersuchungen belegen², dass wir – oft unbewusst – *mentale Ordnungssysteme* anlegen, um den über unsere Sinnesorgane wahrgenommenen Informationsstrom auf uns möglichst effektiv zu verarbeiten. Die bei der Zuordnung zu mentalen Kategoriensystemen (*Taxonomien*) angewandten Kriterien implizieren dabei, dass wir Objekte unterbewusst danach kategorisieren, aus welchen einzelnen Teilen sie aufgebaut sind [Ro78, TH84]. Die Identifikation einzelner Teile, die ein ganzheitliches System ausmachen, und ihrer wechselseitigen Beziehungen zueinander ist somit als entscheidende Komponente menschlicher Kognition zu sehen; entsprechend sind die dadurch angelegten *Partonomien* (Teil-Ganzes-Hierarchien) zentraler Bestandteil menschlicher Wissensorganisation [Bi98, 654]. Zweifelsfrei ist die *Partitionierung* – also die strukturelle Zerlegung zusammengesetzter Systeme – essentiell für ihr allgemeines Verständnis [TZH08]. Die zugehörige Fähigkeit bezeichnen wir in Hinblick auf ihre Relevanz für die Informatikdidaktik als *Denken in Teilen von Ganzen*. Im Sinne jeder *konstruktivistischen Auffassung von Lernen* ist es unerlässlich, im Unterricht an die Vorerfahrungen der Schülerinnen und Schüler (SuS) anzuknüpfen [Du07]. Entsprechend wird die hohe Bedeutung der Erforschung von Schülervorstellungen

¹ Carl von Ossietzky Universität Oldenburg, Didaktik der Informatik, Uhlhornsweg 84, 26129 Oldenburg, vorname.nachname@uni-oldenburg.de

² Ein umfassender, übersichtlicher Überblick über entsprechende Untersuchungen wird in [La87, 12-57] gegeben.

für die Didaktische Strukturierung von Informatikunterricht (IU) im Modell der *Didaktischen Rekonstruktion* (DR) expliziert [Di11]. Da die mentalen Ordnungssysteme, die wir bereits – individuell über unsere Wahrnehmung und Kognition oder kulturell bspw. über die Sprache – gebildet haben, von sich aus bereits einen enormen Einfluss auf die Wahrnehmung neuer Situationen und Eigenschaften haben [Ro78, 29] und dadurch auch auf die Art und Weise, wie wir neue, uns unbekannte Objekte verstehen und begreifen, sind auch Lernvoraussetzungsanalysen, die sie zum Forschungsgegenstand nehmen, im Sinne der DR von Bedeutung für die Informatikdidaktik³. Besonders stark zum Tragen kommen die Aspekte des Denkens in Teilen von Ganzen in den curricularen Vorgaben zu IU z. B. beim Thema *Informatiksysteme* (IS), denn da zu deren Nutzung „ein grundlegendes Verständnis ihres Aufbaus und ihrer Funktionsweise notwendig [ist]“ [GI08, 37] fanden entsprechende Kompetenzanforderungen über die Bildungsstandards der Gesellschaft für Informatik [GI08] auch Einzug in die Curricula der einzelnen Länder. Im Sinne einer konstruktivistischen Auffassung von Lernen (s. o.) ist es somit unerlässlich, entsprechende Schülervorstellungen zu erheben. So soll über das in diesem *Work-in-Progress*-Beitrag vorgestellte Forschungsvorhaben den folgenden Forschungsfragen explorativ am Beispiel der drei IS *Staubsaugerroboter* (SR), *Spielekonsole* (SK) und *Smartphone* (SP) nachgegangen werden: Welche Vorstellungen haben und entwickeln SuS von der Funktionsweise und von der Struktur von zusammengesetzten IS? Welche Teile und welche Ganze identifizieren sie dabei? Welche Beziehungen der Komponenten untereinander identifizieren sie dabei?

2 Methodischer Ansatz und erste Ergebnisse

Um die internalisierten, kognitiven Wissensrepräsentationen zu externalisieren, wird in diesem Untersuchungssetting *Concept Mapping* in leitfadengestützte Interviews integriert. Hierzu wurde ein Leitfaden von zirkulärer Struktur entwickelt, über den in jedem Interview SP, SK und SR miteinander verglichen werden sollen und die Vorstellungen der Befragten zum Aufbau der Systeme gemeinsam mit dem Interviewer auf Basis der Schülerinnen-Aussagen in einer (für die drei IS) gemeinsamen *Concept Map* externalisiert werden. Das Sample der hier vorgestellten Datenerhebung setzt sich aus acht Teilnehmerinnen (B1 bis B8) an einem Förderprojekt zur Steigerung des Anteils an Mädchen und Frauen in informatiknahen Berufen zusammen, die die 8. Klasse einer nds. Oberschule (Zusammenfassung von Haupt- und Realschule) besuchten, bis dato keinen formalen IU hatten und zum Zeitpunkt der Erhebung 13 bis 14 Jahre alt waren. Die Wahl fiel auf vier Einzelinterviews und zwei Partnerinterviews (B3/B4 und B5/B6) von 14 bis 33 Minuten Länge. Zur Auswertung wurden die Interviews transkribiert und anschließend einer *inhaltlich-strukturierenden qualitativen Inhaltsanalyse* [Ma02] unterzogen, wobei die Kategorienbildung deduktiv-induktiv erfolgte. Da die Anfertigung der Concept Maps während der Interviews zweifelsohne im „Grenzbereich zwischen Erhebung und Auswertung“ [Ma02, 86] liegt und die in ihnen enthaltenen Informationen größtenteils redundant mit den Transkriptinhalten sind, rückten sie

³ So befassen sich aktuell bspw. [Br19] oder [RP18] mit der Frage danach, welche Kategorien (*Taxonomien*) Lernende in ihren mentalen Repräsentationssystemen von digitalen Artefakten bilden.

während der Inhaltsanalyse in den Hintergrund. Im Folgenden werden Alltagsvorstellungen, wie sie während der Interviews zum Vorschein kamen, anhand expl. Aussagen dargelegt:

Bezogen auf den **SR** geht B1 davon aus, dass dieser seinen Weg einprogrammiert bekommen muss (B1:195⁴) und „wie so ein Hund“ über einen „Spürsinn“ (B1:187) den Dreck zunächst gezielt aufspürt und dann aufsaugt (B1:191). Andere Schülerinnen hingegen verstehen ihn als System, das den Weg automatisch erkennt (B2:24). Die dazu benötigte Komponente wird von Zweien als *Sensor* identifiziert (B3:27, B6:37); eine weitere verwendet zwar nicht diese Bezeichnung, expliziert aber, dass SR „da vorne [etwas] haben“ (B2:12), worüber sie an die nötigen Informationen – also „wo er langfahren muss“ (B2:10) – gelangen. Das Umfeld wird nach Ansicht von B1 nicht abgespeichert, weil der SR schließlich auch funktioniert, wenn man ihn in einer anderen Umgebung platziert (B1:320). Zwar ist der Ansicht einer weiteren Schülerin nach in SR auch eine *Festplatte* enthalten, jedoch ist ihre Aufgabe dort lediglich „Signale zu empfangen“ (B5:215). In **SK** hingegen ist auf den Komponenten, die sie als *Festplatten* (also „diesen grünen Platten, wo man was anlötet“, B5:101) bezeichnet, „alles drauf gespeichert“ (B5:130). Ihre Beschreibung dieser Komponente ähnelt der, die B2 mit dem Begriff des *Laufwerks* beschreibt (B2:46): Während diese Komponente, deren Rolle mit der eines „Gehirns“ (B2:100) verglichen wird, im SR dafür verantwortlich gemacht wird, zu steuern, wann sich welche Räder drehen sollen, „damit die den Dreck auffegen“ (B2:46), ist Aufgabe des *Laufwerks* im **SP**, dass es „an bleibt [...] und dass alles angezeigt wird und so“ (B2:158). In SK wiederum – in denen das *Laufwerk* aus Platzgründen ebenso wie in SP übrigens nicht so groß ist wie in SR (B2:148) – kann es „auch Sachen in sich speichern“ (B2:150); in SP übernehmen *SIM-* bzw. *SD-Karten* (B2:130) – zwei Begriffe, die von B3 während des Interviews auffallend synonym verwendet werden – bzw. „so etwas wie eine Festplatte“ (B3:124) diese Aufgabe. Während unter den Interviewteilernehmerinnen Einigkeit darüber herrscht, dass *Speicherplatz* eine nötige Voraussetzung dafür ist, „Sachen“ herunterzuladen und speichern zu können (bspw. B1:161, B4:118), herrscht Verwirrung darüber, welche Daten überhaupt gespeichert werden müssen („der [SR] kriegt ja Sachen programmiert [...] und muss man das nicht speichern?“, B1:314-316) und wo sie herkommen („Vom Strom? [...] Ne, ähm, naja irgendwo muss es ja herkommen.“, B3:82), wo die Speicherung gewisser Daten physisch zu verorten ist (lokal oder auf „Plattformen“ oder „Apps“ wie bspw. Google, „[das] im Internet gespeichert ist“, B1:307) und ob überhaupt lokal-physische Speicherkomponenten benötigt werden (B8:16) oder Speicherplatz ausschließlich als – vermeintlich web-basierter – *Zustand* verstanden wird. In Analogie dazu erwecken die Beschreibungen einiger Schülerinnen zum *Akku* den Eindruck, dass sie diesen Begriff eher als *Ladezustand* („Akku ist ja halt Akku, dann hast du das halt nicht mehr“, B3:154), der bspw. dem SP ermöglicht zu „leben“ (B3:157), begreifen statt als *physische* Komponente (bspw. B1:69) in Form eines *Akkumulators*. So erklärt B4, dass „[Akkus] nicht mehr bei so vielen Handys [drinstecken]“ (B4:126) und „es [das] ja auch noch irgendwie anders [gibt]“ (B4:126). Schließlich würden SP „ja auch immer dünner [werden] und dann passen da ja nicht mehr so fette dicke Akkus rein“ (B4:128), weswegen es jetzt „nicht mehr so viele Handys [gäbe], in denen Akkus drin sind“ (B4:128).

⁴ Notation: *Befragte:Zeilennummer*; die Transkripte werden bei Interesse gerne zur Verfügung gestellt.

3 Fazit und Ausblick

Im Rahmen der bisher durchgeführten Interviews konnte eine Reihe an *Alltagsvorstellungen* zum Aufbau von IS herausgearbeitet werden, die vor allem über die Verbindungen einzelner Komponenten untereinander, ablaufende Datenströme sowie Informationsflüsse und die Rollen einzelner Komponenten als *Fehlvorstellungen* aufzufassen sind. Hinsichtlich der Tatsache, dass die bislang befragten Schülerinnen IS unter ihrer Oberfläche häufig als Blackbox behandeln und zur Erklärung ihrer Wirkprinzipien hauptsächlich sicht- und interagierbare Teile identifizieren, kam es dabei über die Interviews hinweg zu einer Stabilisierung. Um entsprechende Schülervorstellungen von Teil-Ganzes-Beziehungen in IS im Sinne der DR für den IU nutzen zu können, sind jedoch zusätzliche, tiefgreifendere Studien nötig, die auch über weitere methodische Ansätze (bspw. das Anfertigen von Zeichnungen) angestrebt werden. In folgenden Interview-Durchläufen nach dem in diesem Beitrag vorgestellten Vorgehens sollen Teil-Ganzes-Denkstrukturen dabei u. A. über stärkeres Einfordern passender Formulierungen („ist enthalten in“, „ist Teil von“, ...) stringenter hinterfragt werden.

Literaturverzeichnis

- [Bi98] Billman, D.: Representations. In (Bechtel, W.; Graham, G., Hrsg.): A Companion to Cognitive Science, Kapitel 51, S. 649–659. Blackwell, 1998.
- [Br19] Brinda, T.; Napierala, S.; Tobinski, D.; Diethelm, I.: Student strategies for categorizing IT-related terms. In: Education and Information Technologies. 2019.
- [Di11] Diethelm, I.; Dörge, C.; Mesaros, A.; Dünnebier, M.: Die Didaktische Rekonstruktion für den Informatikunterricht. In (Thomas, M., Hrsg.): Informatik in Bildung und Beruf. Jgg. P-189 in LNI. GI, S. 77–86, 2011.
- [Du07] Duit, R.: Science Education Research Internationally: Conceptions, Research Methods, Domains of Research. Eurasia Journ. of Math., Science and Tech. Ed., 3(1):3–15, 2007.
- [GI08] GI, Gesellschaft für Informatik e.V.: Grundsätze und Standards für Informatik in der Schule: Bildungsstandards Informatik für die Sek. I (Beilage zu LOG IN, Heft Nr. 150/151). 2008.
- [La87] Lakoff, G.: Women, Fire, and Dangerous Things. Univ. of Chicago Pr., Chicago, 1987.
- [Ma02] Mayring, P.: Einführung in die qualitative Sozialforschung: Eine Anleitung zum qualitativen Denken. Beltz Verlag, Weinheim und Basel, 6. Auflage, 2002.
- [Ro78] Rosch, E.: Principles of categorization. In (Rosch, E.; Lloyd, B. B., Hrsg.): Cognition and Categorisation, S. 27–48. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1978.
- [RP18] Rücker, M. T.; Pinkwart, N.: The things that belong: a grounded theory study of student categorizations of complex technical artifacts. International Journal of Technology and Design Education, 28(3):701–720, 2018.
- [TH84] Tversky, B.; Hemenway, K.: Objects, Parts, and Categories. In: Journal of Experimental Psychology: General. Jgg. 113. American Psychological Assoc., Inc., S. 169–193, 1984.
- [TZH08] Tversky, B.; Zacks, J.; Hard, B.: The Structure of Experience. In: Understanding Events: From Perception to Action, S. 436–464. Shipley, T. F. and Zacks, J. M., 2008.

Zur Beliebtheit des Schulfachs Informatik in der Oberstufe nordrhein-westfälischer Schulen

Ausgewählte Ergebnisse einer Studie zu Einflüssen auf die Wahl von Informatik in der Oberstufe

Niklas Klinge¹, Marco Thomas²

Abstract: Im Rahmen des Projekts „Kriterien zum Informatikunterricht von Schülerinnen und Schülern“ (KISS) wurden fast 5000 Schüler in Informatikkursen an gut 150 Schulen vorwiegend quantitativ zur An- und Abwahl, zu Vorerfahrungen, Erwartungen und zur Abiturplanung befragt. Es lassen sich Zusammenhänge zwischen der Belegung von Informatik in der Sekundarstufe I und der Anwahl von Informatik in der Oberstufe aufzeigen. Insbesondere ist eher ein Desinteresse am Informatikunterricht als die Konkurrenz zu anderen Fächern oder die Lehrerkompetenz ein Grund für die Abwahl von Informatik, wobei Unterschiede zwischen Gymnasium und Berufskolleg deutlich werden.

Keywords: Informatik, Unterricht, Didaktik, Schule, Oberstufe, Bildung, Standards, MINT

1 Informatische Bildung in der GOST NRW

Seit fast 50 Jahren wird Informatik in der Sekundarstufe I (SI) in Nordrhein-Westfalen (NRW) nicht verpflichtend unterrichtet, was vermutlich Auswirkungen auf die Anwahl von Informatik in der Oberstufe (GOST) hat. Die fehlende Gleichberechtigung unter den MINT-Fächern wird als eine weitere Ursache für die Entscheidung von Schülern³, Informatik nicht (weiter-)zu belegen, angenommen. Ein hoher Prozentsatz von durchschnittlich 35 % der Schüler wählt Informatik für das zweite Jahr der Oberstufe wieder ab (in den Naturwissenschaften sind es nur 18 %, in Mathematik 22 %) [N18].

Es existieren nur vereinzelt Studien zur Situation des Informatikunterrichts in Deutschland, die das Wahlverhalten von Schülern untersuchen. Seit 2013 besteht das Ziel des Projekts KISS, diese empirische Lücke zu schließen. Nachdem wir uns im Projekt KISS zunächst auf die SI fokussiert hatten und Lehrer, Schüler sowie Schulleiter befragten, wurden in den Schuljahren 2017/18 und 2018/19 Schüler in der GOST befragt.

¹ Westfälische Wilhelms-Universität Münster, Didaktik der Informatik, Corrensstraße 80, 48149 Münster, niklas.klinge@uni-muenster.de

² Anschrift s. o., marco.thomas@uni-muenster.de

³ Aus Gründen der Lesbarkeit verzichten wir auf die Angabe weiterer Genderformen, soweit dies sachlich nicht erforderlich ist.

2 Untersuchungsmethodik

Die Datenerhebung erfolgte weitestgehend quantitativ mit einem standardisierten Online-Fragebogen. Der Online-Fragebogen umfasst zehn Blöcke zu den Themen „Allgemeine Angaben“, „Vertrautheit mit Anwendungen“, „Berührungen mit Informatik und Programmieren“, „Gründe für das Wahlverhalten“, „Abiturplanung“, „Erwartungen an einen Informatikunterricht“, „Gründe für eine Abwahl“. Die Auswahl der Items orientierte sich an Fragen, die wir bei unseren bisherigen Fragebögen in KISS erfolgreich eingesetzt haben, welche sich wiederum an bewährten Studien orientierten. Gegenstand der Untersuchung sind Schüler an Gesamtschulen, Gymnasien und Berufskollegs in NRW, die in den Schuljahren 2017/18 und 2018/19 einen Informatikkurs in der Oberstufe besuchten. Der Aufruf zur Umfrage wurde ab September 2017 in allen Bezirksregierungen NRWs gestartet. Im Schuljahr 2017/18 wurden insgesamt 903 Schüler an 52 Schulen befragt, im Schuljahr 2018/19 haben sich 4021 Schüler von 131 Schulen beteiligt.

Schulform	Berufskolleg	Gesamtschule	Gymnasium
Gesamt*	101	246	584
Rückmeldungen (in %)	26,73	5,70	24,14

Tabelle 1: Anzahl Schulen bzgl. intendierter Population nach Schulformen. *Quelle: [N18]

3 Ergebnisse und Auswertung

Es können an dieser Stelle aus Platzgründen nur einige Zusammenhänge und Interpretationen der Studie dargestellt werden. Die Zusammenhänge weisen alle eine hohe Signifikanz (Chi-Quadrat höher als 30 und einem p-Wert $< 0,005$) auf. Sprechen wir von wagen Vermutungen, aufgrund geringer Signifikanz oder schlechter Datenlage, werden diese gekennzeichnet.

3.1 Anwahl von Informatik

Wir haben die Schüler in einer Mehrfachauswahl von Möglichkeiten gefragt, warum sie Informatik gewählt haben (ergänzend konnten die Schüler in einem Freitext antworten). Über die Hälfte der Schüler gaben an, Informatik gewählt zu haben, weil sie es interessanter als andere naturwissenschaftliche Fächer finden. Für viele schien der Informatikkurs aber auch eine Alternative zu einer weiteren Fremdsprache zu sein. Zwischen 28 % und 38 % der Schüler hatten schon Vorkenntnisse im Programmieren und zur Informatik. Vergleichbare Aussagen finden sich in unseren Studien zur Anwahl von Informatik in der SI [JT15]. Im näheren Umfeld gab es bei der Hälfte der Schüler Personen, die einen „[...] Beruf mit Bezug zur Informatik ausüben? (z.B. Software-Entwicklung, System-Administration etc.)“. Gut 43 % der Schüler nannten als weiteren Grund, dass ihnen der Unterricht in der Unter- und Mittelstufe gefallen hat.

Gut ein Drittel der befragten Schüler hatte keinen Kontakt mit Informatik in der SI. Fast die Hälfte der Schüler gab an, Informatik als Wahlpflichtfach entweder als reines Fach oder als Kombinationskurs in der SI belegt zu haben. Bei einem Fünftel der Schülerschaft hatte auch die gesamte Klasse das Fach in der SI⁴. Nur etwa 35 % aller befragten Schüler gaben an, dass sie bei der Anwahl von Informatik für die GOST wussten, was sie erwartet.

Es lassen sich Tendenzen für einen positiven Einfluss der Vorkenntnisse aus der SI auf die spätere Anwahl von Informatik in GOST finden. 70% der Schüler, die einen Wahlpflichtkurs in der SI besucht hatten, gaben an, Informatik u. a. deswegen gewählt zu haben, weil sie es in der SI hatten und es ihnen gefallen hat. Besuchten die Schüler nur Ergänzungsstunden bzw. AGs lag deren Anteil bei obiger Aussage nur bei circa 26 %, wurde die gesamte Klasse in Informatik unterrichtet bei 40 %.

3.2 Abwahl von Informatik

Die Schüler wurden gefragt, ob sie Informatik am Ende der EF oder nach dem ersten Halbjahr abwählen wollen und ob sie Informatik sofort abwählen würden, wenn dies möglich wäre. Über alle Schulformen und -stufen mit GOST planen 75 % der Schüler Informatik nicht abzuwählen. Wenn die Abwahl sofort möglich wäre, würde ein Fünftel der Schüler Informatik abwählen. Warum die Schüler Informatik abwählen bzw. sofort abwählen wollen, kann verschiedene Gründe haben. Im Folgenden können wir nur zwei Zusammenhänge von erhobenen Parametern darstellen und ansatzweise diskutieren.

Interesse an Informatik und dem Programmieren

Es wird deutlich, dass bei Interesse an dem Fach Informatik, auch kein Abwahlwunsch besteht. Stimmen die Schüler Aussagen wie „Ich interessiere mich für Computer“ oder „Ich möchte programmieren“ zu, so wollen mehr als 75 % dieser Schüler den Kurs nicht abwählen. Häufig ist Desinteresse ein Grund gegen die Fortführung eines Informatikkurses: Schüler, die Informatik abwählen möchten, wollen nicht programmieren, interessieren sich nicht für Computer oder benötigen aus ihrer Sicht Informatik nicht für ihre Zukunft. Werden andere Wahlpflichtfächer von den Schülern als interessanter angegeben, so wollen über 40% dieser Schüler den Kurs abbrechen. Einigen Schülern sind die Informatik und das Programmieren aber auch zu schwer und möchten deswegen das Fach nicht mehr belegen.

Die Zufriedenheit der Schüler mit dem Informatikunterricht ist für die Abwahl von Informatik bedeutend. Je nachdem, ob die Schüler Informatik abwählen möchten oder nicht, stellen sie sich den Unterricht anders vor. Diejenigen, die Informatik nicht mehr belegen möchten, wollen weniger „mathematische Aufgaben lösen“, „Automaten analysieren und modellieren“ oder „über informatische Sachverhalte argumentieren“. Diese Schüler betonen, dass Inhalte langsam und verständlich erläutert werden sollen.

⁴ Z. B. im Rahmen eines schulinternen Lehrplans mit Pflichtstunden Informatik in der Unterstufe. Aktuell werden in NRW (Modell-)Versuche zur Pflichtstunden Informatik in der SI durchgeführt.

Schulform/-stufe

Die Beliebtheit des Informatikkurses scheint nach Schulform und -stufe zu variieren. Schüler vom Gymnasium führen am ehesten den Informatikkurs weiter: 8 von 10 Schülern geben an, den Kurs nicht abzuwählen. Nur etwa 7 % der Schüler wählen Informatik nach der EF oder zum Halbjahr ab. Beim Berufskolleg ist die Quote mit 15 % mehr als doppelt so hoch und nur 6 von 10 Schülern wollen den Kurs weiterhin belegen. Erklären lässt sich dies vermutlich dadurch, dass Schüler an den befragten Berufskollegs Wirtschaftsinformatik verpflichtend⁵ belegen müssen, sich aber für das Fach bzw. die Inhalte nicht interessieren.

4 Fazit und Ausblick

Die Mehrheit der Schüler in den Informatikkursen ist an Informatik interessiert, aber Desinteresse bedingt die Abwahl von Kursteilnehmern. Eine Anwahl von Informatik in der GOST wird durch Informatik in der Sekundarstufe I begünstigt. Die Untersuchung zeigt, dass eher die Schüler vom Gymnasium Informatik fortführen möchten als die vom Berufskolleg.

Literaturverzeichnis

- [JT15] Janzen, Irina; Thomas, Marco; Angelica Yomayuzza (2015): Wahlverhalten zum Schulfach Informatik in der SI. In: Jens Gallenbacher (Hg.): Informatik 2015. Informatik allgemeinbildend begreifen; 16. GI-Fachtagung Informatik und Schule ; Tagung vom 20.-23. September 2015. Bonn. S. 181–190.
- [N18] NRW: Das Schulwesen in NRW aus quantitativer Sicht. Amtliche Schuldaten zu den Schuljahren 2014-2018
<https://www.schulministerium.nrw.de/docs/bp/Ministerium/Service/Schulstatistik/Amtliche-Schuldaten/index.html>

⁵ <https://www.berufsbildung.nrw.de/cms/bildungsgaenge-bildungsplaene/berufliches-gymnasium-anlage-d/studentafeln/anlage-d-27.html> (14.01.2019)

Informatikwissen im Schulalltag sichtbar machen

„Wie funktioniert das Internet?“ in Kinderzeichnungen

Bettina Waldvogel¹

Abstract: Primarschulkinder gehen geschickt mit digitalen Geräten um und erwecken einen kompetenten Eindruck. Diese Anwenderfertigkeiten täuschen nicht selten über die Tatsache hinweg, dass die grundlegenden Konzepte falsch verstanden wurden oder gar nicht bekannt sind. Das stellt die Lehrpersonen vor die Schwierigkeit, auf welchem Vorwissen sie aufbauen können. Die Forschung kennt verschiedene Methoden zur Erhebung von Präkonzepten. Diese eignen sich aber meist nicht für den Schulalltag - sei es, weil die Auswertung der Daten zu aufwändig oder das Erhebungssetting im regulären Klassenunterricht nicht durchführbar ist. Im Unterricht muss die Erhebung von Vorwissen schnell gehen und vor allem muss die Lehrperson auf einen Blick erkennen können, wo Wissenslücken bestehen oder problematische Fehlkonzepte vorliegen. In diesem Beitrag wird gezeigt, wie Kinderzeichnungen im Unterricht genutzt werden können zur schnellen Erhebung der Schülervorstellungen anhand des Beispiels «Wie funktioniert das Internet?». Die im regulären Unterricht entstandenen Zeichnungen liefern gleichzeitig auch wertvolle Anknüpfungspunkte für die Forschung.

Keywords: Präkonzepte, Vorwissen, Kinderzeichnungen.

1 Einleitung

Im Rahmen des Moduls Medien- und Informatik im Lehrplan 21 [Le16] werden in der deutschsprachigen Schweiz eine Vielzahl von Kompetenzen unterrichtet. Es fehlen Methoden, die der Lehrperson in kurzer Zeit, im Rahmen des regulären Unterrichts das Wissen der Schülerinnen und Schüler sichtbar machen.

In der Wissenschaft gibt es viele Ansätze, wie Alltagswissen und Präkonzepte erhoben und visualisiert werden können. Eine anschauliche Gegenüberstellung von Erhebungsmethoden zum Alltagswissen in der Informatik findet man zum Beispiel bei Rücker [Rü17]. Die präsentierten Methoden basieren hauptsächlich auf Interviews, welche eine hochgradig individualisierte Erfassung des Wissens ermöglichen, deren Durchführung und Auswertung jedoch zeitaufwendig und im Rahmen des regulären Unterrichts kaum realisierbar ist.

Novak und Cañas [NC08] empfehlen Concept Maps anstelle von Interviews unter anderem wegen der effizienten, automatisierten Auswertung. Für die Schülerinnen und

¹ Pädagogische Hochschule Schaffhausen, Fachbereich Medien und Informatik, Ebnatstrasse 80, CH-8200 Schaffhausen, bettina.waldvogel@phsh.ch

Schüler, sowie für die Lehrpersonen stellt aber allein schon die Bedienung der Concept Mapping Software eine zeitraubende Hürde dar.

Mit Kinderzeichnungen als Erhebungsmethode, zeigt Neuss [Ne17] eine Möglichkeit, die ohne aufwändige Einführung angewendet werden kann, praktisch auf jeder Schulstufe.

Die vorliegende Arbeit beschreibt eine Vorstudie zur Klärung, inwiefern Kinderzeichnungen gleichzeitig für die schnelle, unkomplizierte Erfassung des Vorwissens bzw. zur Dokumentation von Lernfortschritten im Schulalltag, aber auch für weiterführende wissenschaftliche Studien genutzt werden können.

2 Vorgehen und Unterrichtssetting

In der folgenden Untersuchung wurden die Schülerinnen und Schüler zum ersten Mal mit dem Internet als eigentliches Unterrichtsthema konfrontiert. Das Internet haben alle schon in der Freizeit benutzt. Dabei haben einige Schülerinnen und Schüler lückenhafte oder falsche Vorstellungen entwickelt.

Der Auftrag an die Schülerinnen und Schüler lautete: „Wie funktioniert das Internet? – Zeichne alles, was Dir dazu einfällt. Einzelne Teile dürfen beschriftet werden.“ Um das Vorwissen und den anschliessenden Wissenserwerb zu visualisieren wurde folgendes Vorgehen gewählt:

1. Ersterhebung am Anfang der Lektion: Zeichnung, ca. 15 Minuten.
2. Erklärvideo vom Schweizer Schulfernsehen (SRF mySchool [SRF17]) schauen und diskutieren ca. 15 Minuten (Videodauer 5:41 Minuten).
3. Zweiterhebung am Ende der gleichen Lektion: Zeichnung, ca. 15 Minuten.
4. Dritterhebung einen Monat später: Zeichnung, ca. 15 Minuten.

Die Erst- und Zweiterhebung wurden mit insgesamt 42 Schülerinnen und Schülern aus zwei Klassen durchgeführt, die Dritterhebung nur an einer Klasse mit 22 Kindern.

3 Resultate

Im Erklärvideo wird das Internet als weltumspannendes Netzwerk gezeigt, sowie vier zentrale Elemente des Internets erklärt: Server, Router, Kabel und Endgeräte. Die Zeichnungen wurden in einer ersten rudimentären Auswertung in Hinblick auf das Vorhandensein dieser Elemente untersucht. Weiter wurde darauf geachtet, ob diese Elemente isoliert oder vernetzt dargestellt wurden. Besonders interessante Einblicke entstanden, inwiefern das Vorwissen aus der Ersterhebung in der Zweit- und Dritterhebung noch in Erscheinung trat.

In der Ersterhebung wurden im Durchschnitt 0.9 von den vier im Erklärvideo genannten Elementen genannt, meist die Endgeräte. Am Häufigsten wurden jedoch die Logos von Google (24 mal) und YouTube (23 mal) gezeichnet, gefolgt von Spielen und anderen Apps. In der Lebenswelt der Kinder wird das Internet als Dienstleistung rege genutzt. Das Internet als weltumspannende Infrastruktur ist jedoch weitgehend unbekannt.

In der Zweiterhebung (nach dem Erklärvideo) wurden im Durchschnitt 3.5 der vier im Erklärvideo genannten Elemente gezeichnet. Interessanterweise wurden Server, Router und Kabel oft gezeichnet und sogar miteinander vernetzt. Die Endgeräte sind dafür aber in Vergessenheit geraten (Abb. 1).

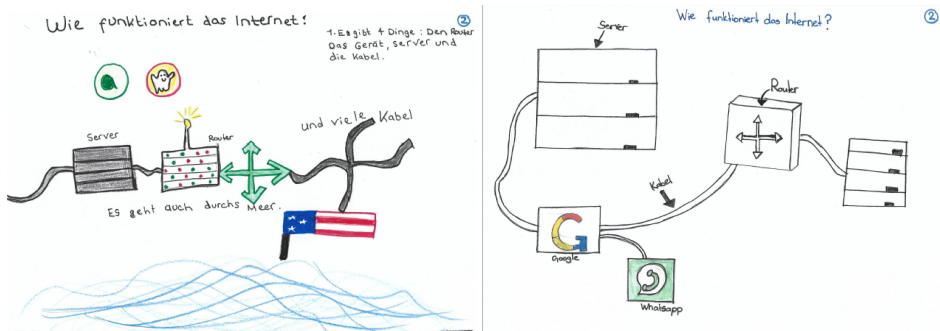


Abb. 1: In der Zweiterhebung dominieren drei der vier im Erklärvideo genannten Elemente des Internets: Kabel, Server, Router. Die Endgeräte wurden oft vergessen.

Hinweise darauf, ob es sich bei der Zweiterhebung nur um eine kurzfristige Konstruktion oder tatsächlich um einen Konzeptualisierungswechsel handelt, gibt eine Dritterhebung, welche einen Monat später stattfand. Im Durchschnitt wurden 3.1 der vier im Erklärvideo genannten Elemente gezeichnet. Also nur geringfügig weniger als in der Zweiterhebung aber deutlich mehr als in der Ersterhebung. Die Schülerinnen und Schüler haben also etwas gelernt. Viele haben ihre Konzeptualisierung nachhaltig verändert, weg vom Endgeräte- und App-zentrierter Sichtweise hin zu einem globalen, weltumspannenden Blick auf das Internet.

4 Schlussfolgerungen und Ausblick

Zeichnungen und Skizzen erlauben den Lehrpersonen schnell zu erfassen, welche Konzeptualisierungen vorhanden sind. Die oben skizzierte rudimentäre Auswertung eignet sich für den Unterrichtsalltag. Wichtig ist, die Zeichnungen zu diskutieren und reflektieren, damit sich die Kinder ihrer Vorstellungen bewusst werden [HM18, S.51].

Die Tatsache, dass die meisten Zeichnungen zwischen der Zweit- und Dritterhebung keine grossen Unterschiede aufweisen, legt nahe, dass mehr als nur eine spontane Ad-hoc Konstruktion, sondern ein nachhaltiges Umlernen stattgefunden hat.

Nicht allen Kindern fällt das Zeichnen leicht. Deshalb schlagen verschiedene Autoren begleitende Interviews vor [Ne17]. Es wäre auch denkbar, Zeichnungen vorzugeben und diese hinterfragen zu lassen im Sinne der von Weigend [2005] vorgeschlagenen Vorgehensweisen zur Erschliessung der intuitiven Modelle.

Die rudimentäre Vorstudie für den Unterrichtsalltag lässt grosses Potenzial in den Kinderzeichnungen vermuten. Eine Studie mit systematischer Auswertung ist in Planung.

Literaturverzeichnis

- [Le16] Lehrplan 21, Modul Medien und Informatik, Deutschschweizer Erziehungsdirektorenkonferenz, 2016.
- [Rü17] Rücker, M.T., Pancratz, N., Gold-Veerkamp, C., Pinkwart, N., Brinda, T., 2017: Alltagsvorstellungen in der Informatik: Erhebungsmethodik und Implikationen für den Unterricht, I.Diethelm (Hrsg.): Informatische Bildung zum Verstehen und Gestalten der digitalen Welt, Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2017, 393-400.
- [NC08] Novak, J., and A.J. Cañas, 2008: The Theory Underlying Concept Maps And How to Construct them. Technical Report IHMC CmapTools 2006-01 Rev 01-2008, Florida Institute for Human and Machine Cognition.
- [Ne17] Neuss, N., 2017: Kinderzeichnungen als Erhebungsmethode. In: Mikos L. und Wegener C. (Hrsg.), Qualitative Medienforschung. Ein Handbuch, 2. Auflage. UVK Verlagsgesellschaft, Konstanz und München. S. 380 – 388.
- [SRF17] Schwiezer Radio und Fernsehen, 2017: Wie funktioniert das Internet? Auf Youtube: <https://www.youtube.com/watch?v=e9FJpNFQWi8> (zuletzt besucht: 02.05.2019)
- [HM18] Hartinger, A., und L. Murmann, 2018: Schülervorstellungen erschliessen – Methoden, Analyse, Diagnose. In: Adamina M., M. Kübler, K. Kalcsics, S. Bietenhard, E. Engeli (Hrsg.), «Wie ich mir das denke und vorstelle...», Bad Heilbrunn Klinkhardt. S. 51-62.
- [We05] Weigend, M., 2005: Intuitive Modelle in der Informatik. In: Lecture Notes in Informatics. Unterrichtskonzepte für informatische Bildung. Proceedings zur 11. GI-Fachtagung Informatik und Schule. S. 275-284.

Poster

Algorithmen im Anfangsunterricht mit RunCode

Christoph Gräßl¹

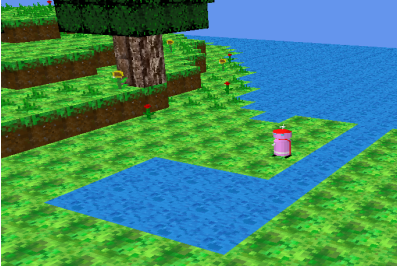
An bayerischen Gymnasien wird aktuell in den Jahrgangsstufen 6 und 7 einstündiger Anfangsunterricht in Informatik gehalten. Ein Schwerpunkt liegt im Bereich *Beschreibung von Abläufen durch Algorithmen*. In den Schulbüchern und Unterrichtsempfehlungen wird davon ausgegangen, dass der Zugang über altersgerechte Lernumgebungen erfolgt, da herkömmliche Programmiersprachen (wie Pascal, C oder Java) ungeeignet sind. Häufig wird hier eine Anwendung, basierend auf einer Arbeit von Richard E. Pattis, eingesetzt in der ein virtueller Roboter in einer schachbrettartig aufgebauten Welt programmiert werden kann. Durch diese Programme kann der Roboter Aufgaben, wie beispielsweise das Legen eines Quadrates mit Ziegelsteinen, erledigen. Insgesamt sind die Möglichkeiten einer abwechslungsreichen Aufgabenstellung recht beschränkt und die Grafik ist zwar zweckmäßig, aber dennoch wenig ansprechend und keineswegs zeitgemäß. Infolgedessen ist die Schülermotivation, mit diesem System zu arbeiten, nicht von Dauer.

RunCode kombiniert den bewährten Ansatz von Pattis mit einer ansprechenderen 3D-Umgebung und zusätzlichen Befehlen, die es erlauben, spannende Aufgaben zu stellen. Die Lernumgebung, dargestellt in Abbildung 1, benötigt minimale Einarbeitungszeit und ist intuitiv bedienbar.



Abb. 1: Das Benutzerinterface von *RunCode*

¹ Donau-Gymnasium Kelheim, Rennweg 61, 93309 Kelheim c.graessl@donau-gymnasium.de



```
wiederhole solange nicht betretbar
  wiederhole solange nicht betretbar
    hinlegen
  *wiederhole
  schritt
*wiederhole
```

Abb. 2: Links: Roboter hebt einen Teich aus. Rechts: Programm zum Erreichen einer Nachbarinsel.

Die grafische Benutzeroberfläche besitzt im linken Bereich einen Editor, in dem das Programm eingegeben werden kann. Abspeichern und Laden ist ebenfalls möglich. Unter dem Editor befinden sich Knöpfe zum Ausführen, Anhalten und Unterbrechen des Codes. Die Ausführungsgeschwindigkeit kann mit einem Regler eingestellt werden. Zur besseren Übersicht können benutzerdefinierte Anweisungen in eine Bibliothek verschoben werden und verschwinden somit aus dem Editor. Außerdem ist es möglich per Knopfdruck alle Befehle, Kontrollstrukturen und Bedingungen mit Beschreibung angezeigt zu bekommen. Während der Programmausführung wird die aktuelle Befehlszeile markiert. Sowohl der Programmcode, als auch die erzeugte *Welt* lassen sich abspeichern beziehungsweise laden. *RunCode* besitzt noch eine Reihe weiterer Eigenschaften, die aber aufgrund von Platzmangel in diesem Schriftstück nicht präsentiert werden können.

Es werden im Folgenden einige Aufgabenbeispiele, die auch bereits im Unterricht getestet wurden, kurz skizziert:

- Grabe, beginnend am Meer, einen Bachlauf der Länge 5 Schritt aus. Der Graben füllt sich dann automatisch mit Wasser. Wenn du das geschafft hast, dann schreibe das Programm so, dass am Ende des Grabens ein See entsteht (Abbildung 2 links).
- Grabe einen Tunnel durch den Berg.
- Baue eine Brücke bis zur nächsten Insel (mögliche Lösung ist in Abbildung 2 rechts).
- Grabe einen Tunnel unter dem Meer bis zur nächsten Insel. Achte darauf, mindestens 6 Schritt tief zu graben, damit kein Wasser in den Tunnel eindringt.
- Erstelle einen 6x6 Schritt breiten und 20 Schritt tiefen Schacht. Baue anschließend eine Treppe, die vom Grund des Schachts an die Oberfläche zurück führt.

Die in dieser Arbeit vorgestellte Lernumgebung wurde im Schuljahr 2018/19 in zwei Informatikklassen der 7. Jahrgangsstufe eingesetzt und evaluiert. Obwohl die verwendeten Computer (unter Windows 7) vom aktuellen Stand der Technik eher als leistungsschwach anzusehen sind, lief *RunCode* flüssig und ruckelfrei. Die Schüler fanden sich schnell zurecht und die Einarbeitungszeit war sehr gering. Der Lehrplanstoff im Bereich *Algorithmen* konnte mit der Software komplett umgesetzt werden. *RunCode* ist auf <http://klassenkarte.de> kostenlos erhältlich.

Identifizierung leistungsstarker Schülerinnen und Schüler in der Informatik

Claudia Hildebrandt¹, Matthias Matzner¹

Abstract: Die schulischen Entwicklungsmöglichkeiten leistungsstarker und potenziell besonders leistungsfähiger Schülerinnen und Schüler in der Schule - unabhängig von Herkunft, Geschlecht und sozialem Status - im Regelunterricht zu fördern, ist eines der Ziele des BMBF-Projekts LemaS (Leistung macht Schule) bzw. des Teilprojekts MINT-Informatik. Mithilfe einer Literaturrecherche wird ein erster theoretischer Merkmalskatalog aufgestellt, der zur möglichen Identifizierung leistungsstarker und potenziell besonders leistungsfähiger Schülerinnen und Schüler in der Sekundarstufe I beitragen soll.

Keywords: Begabung, Leistung, Informatik, Diagnose

1 Einleitung und das Projekt LemaS

Mehr Chancengerechtigkeit für alle, unabhängig von Herkunft, Geschlecht und sozialem Status, ist eines der bildungspolitischen Ziele der Initiative „Leistung macht Schule - LemaS“, die Bund und Länder gemeinsam unterstützen. Umgesetzt wird LemaS mithilfe von 22 Teilprojekten, von denen sich acht auf den MINT-Bereich beziehen. Deutschlandweit sind 300 Schulen aus dem Primar- und Sekundarbereich in die Umsetzung eingebunden. Der offizielle Start für die Bund-Länder-Initiative erfolgte am 30.01.2018 in Berlin [KB18].

Dem LemaS-Verbundprojekt liegt ein mehrdimensionaler Leistungsbegriff zugrunde, der Leistung einerseits als schulbezogene Leistung betrachtet, andererseits aber die Persönlichkeitsentwicklung und gesellschaftliche Verantwortung mit einschließt. Die Begriffe des Leistungspotenzials und der Begabung werden gleichgesetzt. Das Leistungspotenzial ergibt sich dabei als individuelle Konstellation aus Fähigkeiten und (dynamisch sich entwickelnden) Persönlichkeitsmerkmalen. Begabung ist eine Disposition von Leistung, die „durch langfristige systematische Anregung, Begleitung und Förderung das Individuum in die Lage versetzt, sinnorientiert und verantwortungsvoll zu handeln und auf Gebieten, die in der jeweiligen Kultur als wertvoll erachtet werden, anspruchsvolle Tätigkeiten auszuführen“ [Ip09, S. 16]. Für die Entwicklung eines theoretischen Merkmalskatalogs ist die Unterscheidung zwischen leistungsstarken und potenziell besonders leistungsfähigen Schülerinnen und Schülern noch nicht notwendig, so dass im Folgenden vereinfacht von leistungsstarken Schülerinnen und Schülern gesprochen wird.

¹ Carl von Ossietzky Universität Oldenburg, Didaktik der Informatik, Uhlhornsweg 84, 26129 Oldenburg, claudia.hildebrandt@uni-oldenburg.de

2 Das Konstrukt informatischer Leistungsstärke sowie Ausblick

Eine erste Annäherung an die informatikspezifischen Begabungsmerkmale für Kinder in der Sekundarstufe I, die das Fundament der informatischen Leistungsstärke sind, erfolgt mithilfe der Beachtung und Erweiterung der Ergebnisse von Kujath und Schwill [KS11], die die Problemlösefähigkeit von Hochleistern (Endrundenteilnehmer des Bundeswettbewerbs Informatik 2005) im Vergleich zu Niedrigleistern (leistungsschwächere Informatikstudenten vor dem Vordiplom) untersucht haben (siehe „Informatikspezifische Merkmale“, Abb. 1). Weiterhin ist anzunehmen, dass die begabungsstützenden Persönlichkeitseigenschaften, die im Modell der Entwicklung mathematischer Begabungen im Grundschulalter nach Käpnick und Fuchs [Kä13] zu finden sind, ebenfalls für leistungsstarke Schülerinnen und Schüler der Informatik zutreffen. Ergänzt werden sie durch die Kommunikationsfähigkeit sowie die Fähigkeit zum Perspektivwechsel (siehe „Stützende Persönlichkeitseigenschaften“, Abb. 1).



Abb. 1: Merkmalsystem angelehnt an [Kä13] sowie [KS11]

Die aufgestellten informatikspezifischen Leistungsmerkmale sind mithilfe geeigneter Indikatoraufgaben zu verifizieren oder zu widerlegen bzw. neue Merkmale müssen hinzugefügt werden. Kontraststudien mit Hoch- und Niedrigleistern werden folgen.

Literaturverzeichnis

- [Ip09] International Panel of Experts for Gifted Education. (2009). Professionelle Begabtenförderung. Empfehlungen zur Qualifizierung von Fachkräften in der Begabtenförderung. Salzburg: Österreichisches Zentrum für Begabtenförderung und Begabungsforschung (ÖZBF).
- [Kä13] Käpnick, F.: Theorieansätze zur Kennzeichnung des Konstruktes „Mathematische Begabung“ im Wandel der Zeit. In (Firtzlar, T.; Käpnick, F., Hrsg.): Mathematische Begabungen, 1. Aufl., WTM-Verlag, Münster, 2013, S. 9–40.
- [KB18] Käpnick, F.; Benölken, R.: „Leistung macht Schule“ (LemaS). Ein BMBF-Projekt zur Förderung leistungsstarker und potenziell besonders leistungsfähiger Schülerinnen und Schüler. Mitteilungen der GDM 105/2018, S. 27-28, 2018.
- [KS11] Kujath, B., Schwill, A.: Hochleister bei der Lösung informatischer Probleme - Was können Niedrigleister lernen. In (Bayrhuber, H.; Harms, U.; Muszynski, B.; Ralle, B.; Rothgangel, M.; Schön, L.-H.; Vollmer, H. J.; Weigand, H.-G., Hrsg.): Empirische Fundierung in den Fachdidaktiken, S. 147-158, 2011.

Gratulation! Sie haben die Prüfung bestanden.

Ergebnisse einer empirischen Umfrage zum ECDL an österreichischen Schulen.

Peter Micheuz¹

Abstract: Der ECDL hat seit mehr als 20 Jahren einen nicht zu vernachlässigenden Teil österreichischer Schulgeschichte im Kontext digitaler (Aus)Bildung geschrieben. Dieser empirisch angelegte Beitrag befasst sich mit den Ergebnissen einer repräsentativen Umfrage über dieses internationale Zertifikat unter österreichischen Informatiklehrenden.

Keywords: Zertifikat, Syllabus, Qualifikation, IT-Skills, Digitale Grundbildung

1 Einleitung und methodisches Vorgehen

Im Jahr 1997 kam der ECDL (European Computer Driving License) durch die Österreichische Computergesellschaft auch nach Österreich. Kurz darauf wurde der ECDL an österreichischen Schulen angeboten. Der Schwerpunkt des Beitrages liegt in der Darstellung der Ergebnisse einer im Jahr 2017 durchgeführten repräsentativen Online-Umfrage unter Experten und Lehrkräften. Er liefert ein differenziertes (Stimmungs)Bild der subjektiven Einstellungen und Meinungen zum Angebot des ECDL an österreichischen Schulen.

Im Juni 2017 wurde unter den registrierten Usern des österreichischen Informatik-Portals eine Online-Umfrage durchgeführt. Mit der Erhebung sollte eine Momentaufnahme von Expertenmeinungen einholt werden, um daraus eine Standortbestimmung zum ECDL an österreichischen Schulen ableiten zu können. 140 Rückmeldungen von Experten aus ganz Österreich und aus verschiedenen Schultypen wurden sowohl quantitativ als auch qualitativ ausgewertet.

2 Auszüge der Ergebnisse der Umfrage

Beispiele von Freiantworten:

- „IT/EDV/Office Wissen geht durch Handy & Co teilweise verloren. Daher ECDL nach wie vor wichtig! (Nach NMS in AHS/BMHS teilweise schon wieder grobe Lücken im Anwendungswissen)“
- „Nachweis von erworbenen Kompetenzen, administrativer Aufwand“

¹ Alpen-Adria-Universität Klagenfurt, Institut für Informatik und deren Didaktik, Universitätsstraße 65-67, 9020 Klagenfurt, peter.micheuz@aau.at

- „Produktschulung, microsoftlastig, keine informatischen Konzepte“
- „kein Thema / AHS: zu früh für's Berufen / Kompetenzen können lustvoller erworben werden“

Kumulierte exemplarische Ergebnisse von quantitativ erfassten Daten:

	ja/eher ja	eher nein/nein	k.A.
Der ECDL an den österreichischen Schulen ist eine Erfolgsgeschichte.	107	28	5
Der ECDL wird an den österr. Schulen in den kommenden Jahren an Bedeutung gewinnen.	55	78	7
Die Kosten für das ECDL-Zertifikat sind insgesamt angemessen.	86	51	3
Der Preis für die Skillscard (dzt. 46 Euro) ist zu teuer.	89	47	4
Die Prüfungskosten (pro Modul 14 Euro) sind zu hoch.	64	73	3
Es sollte keine Wahlmodule geben.	18	99	23
Der ECDL hat mit Informatik viel zu tun.	77	63	
Das dzt. Angebot an 9 ECDL-Modulen passt.	111	21	6
Das ECDL-Zertifikat ist für (fast) alle SEK I - Schüler/innen schaffbar.	76	58	6

Auszüge der SWOT-Analyse zu inhaltlichen Aspekten:

Kategorie	Stärken – Chancen	Schwächen - Bedrohungen
Inhaltliche Aspekte	<i>Klare Stoffabgrenzung, Basis-kompetenzen. Gute Unterlagen (Portal easy4me). 1x1 der Informatik. Wirtschaftszertifikat. Abdeckung der Office-Skills.</i>	<i>Inhalte z.T. lebensfremd. Nur bedingt für alle von Bedeutung. Informatik ist mehr als ECDL. Hat nichts mit Informatik zu tun. Informatik kommt zu kurz. Zuviel Produktschulung – keine Konzeptschulung. Zu einfach: Zuviel wird bei Prüfungen angegeben (z.B. Formeln)</i>

3 Schlussanmerkungen

Mit diesem Beitrag wurde versucht, auf Grundlage einer Befragung unter Experten ein möglichst objektives und nicht „schöngefärbtes“ (Stimmungs)Bild des ECDL an österreichischen Schulen zu zeichnen. Die Umfrage ergab eine starke Polarisierung, wobei sich die Befürworter und Gegner des ECDL ziemlich die Waage halten.

Die kürzlich ab dem Schuljahr 2018/2019 verordnete Einführung des Faches Digitale Grundbildung für die Sekundarstufe I (5.-8. Schulstufe) mit einem sehr detaillierten Lehrplan hat auch die Rahmenbedingungen für den ECDL geändert. Derzeit kann noch niemand sagen, wie sich das auswirken wird. Die Entwicklungen rund um den ECDL an österreichischen Schulen befinden sich auf jeden Fall in einer sehr interessanten Phase.

Freies Explorieren und Programmieren

Erkenntnisse über die Eignung didaktischer Materialien mit informatischem Fokus für offene Unterrichtskonzepte in der Grundschule

Philipp Straube¹, Martin Brämer² und Hilde Köster³

Abstract: Innerhalb eines offenen Lernsettings wurde das Interesse an unterschiedlichen Lernmedien mit informatischem Fokus in einer Grundschulklasse evaluiert. Es zeigten sich signifikante Genderdifferenzen.

Keywords: Sachunterricht, Computational Thinking, Grundschule

Die fortschreitende Digitalisierung der Lebenswelt führt zu Herausforderungen, die Kompetenzen erfordern, die über eine reine Medienbildung hinausgehen [Dö17]. Dementsprechend ist eine Förderung informatischer Kompetenzen auch in der Grundschule anzustreben [Ge19].

Didaktische Materialien mit informatischem Fokus, die potentiell auch für die Grundschule geeignet scheinen, zeigen zunehmende Verbreitung. Eine systematische Evaluation dieser Materialien stellt aktuell ein Desiderat dar. Diese Studie untersucht das Interesse von Kindern einer 1.-3. Klassenstufe in Bezug zu unterschiedlichen Lernmedien mit informatischem Fokus. Konkret soll untersucht werden, *ob sich Unterschiede im Interesse bei Mädchen und Jungen in Relation zu unterschiedlichen Lernmedien mit informatischem Fokus zeigen.*

Interesse wird hier verstanden als „[...] die Beziehung einer Person zu und die Auseinandersetzung mit erfahrbaren Ausschnitten ihrer Umwelt. Im Hintergrund steht eine Person-Gegenstands-Konzeption“ [Kr99]. Als ein Indikator für Interesse wird hier die maximale Verweildauer an den konkreten Stationen betrachtet.

Die hier betrachtete Lernumgebung orientiert sich an der Konzeption des Freien Explorierens und Experimentierens [Kö06] und besteht aus elf verschiedenen Angeboten mit informatischem Fokus, die teilweise hardwarebasiert (z.B. Roboter) und teilweise ausschließlich softwarebasiert (z.B. Online-Spiele) sind. Die Kinder sollten die Materialien selbstbestimmt und selbstorganisiert explorieren bzw. damit arbeiten.

¹ Freie Universität Berlin, FB Erziehungswissenschaften und Psychologie, AB Grundschulpädagogik / Sachunterricht, Habelschwerdter Allee 45, 14195 Berlin, philipp.straube@fu-berlin.de

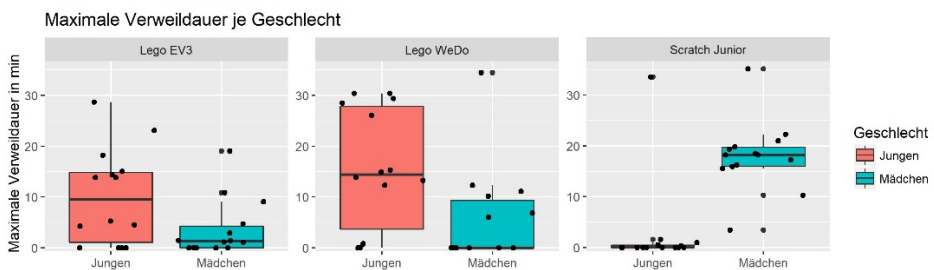
² siehe 1, martin.braemer@fu-berlin.de

³ siehe 1, hilde.koester@fu-berlin.de

Die Stichprobe besteht aus einer Berliner Grundschulklasse (N = 28), die jahrgangsübergreifend (Jül 1.-3. Klasse) lernt. Die Gruppe beinhaltet 14 Jungen und 14 Mädchen (1. Klasse: 8 Schüler*innen; 2. Klasse: 10 Schüler*innen; 3. Klasse: 10 Schüler*innen).

Die Lernumgebung wurde videographiert und die Daten quantitativ im Hinblick auf die maximale Verweildauer an den Stationen ausgewertet. Mögliche Unterschiede wurden mit dem Wilcoxon-Man-Whitney-U-Test geprüft.

Die Analysen zeigen für die drei hier betrachteten Materialien (Scratch Jr., Lego® EV3, Lego® WeDo) Genderdifferenzen in Bezug auf die maximale Verweildauer. Eine statistische Überprüfung zeigt nicht-signifikante Unterschiede an der Station Lego® WeDo ($U=17$, $p=0.050$, $d=0.63$). Ein signifikanter Effekt mittlerer Stärke zu Gunsten der Jungen zeigt sich an der Station Lego® EV3 ($U = 7$, $p = 0.041^*$, $d = 0.65$). Bei Scratch Junior konnte ein höchst signifikanter Effekt hoher Effektstärke zu Gunsten der Mädchen festgestellt werden. ($U = 102$, $p = 0.000^{***}$, $d = 1.01$).



Die Ergebnisse deuten darauf hin, dass Lego® EV3 eher Jungen zu interessieren scheint, wohingegen die Mädchen eher eine Präferenz für Scratch Junior zeigen. Die Differenz in der Verweildauer zu Gunsten der Jungen an der Station Lego® WeDo verfehlt die Signifikanz knapp. Folgende Untersuchungen müssen zeigen, inwiefern diese Trends weitere Evidenz finden.

Literaturverzeichnis

- [Dö17] Döbeli Honegger, B.: Mehr als 0 und 1. Schule in einer digitalisierten Welt. Hep Verlag, Bern, 2017.
- [Ge19] Gesellschaft für Informatik: Kompetenzen für informatische Bildung im Primarbereich. https://dl.gi.de/bitstream/handle/20.500.12116/20121/61-GI-Empfehlung_Kompetenzen_informatische_Bildung_Primarbereich.pdf, 14.02.2019.
- [Kö06] Köster, H.: Freies Explorieren und Experimentieren. Eine Untersuchung zur selbstbestimmten Gewinnung von Erfahrungen mit physikalischen Phänomenen im Sachunterricht. Logos-Verl., Berlin, 2006.
- [Kr99] Krapp, A.: Intrinsische Lernmotivation und Interesse. In Zeitschrift für Pädagogik, 1999, 45; S. 387–406.

Aufhören oder weitermachen?

Ursachen des Studienabbruchs im Fach Informatik

Juliane Wegner¹, Mona Arndt²

Abstract: Der Übergang von der Schule in die Hochschule gestaltet sich für Studenten der Informatik auf vielschichtige Art und Weise problematisch. Wesentliches Ziel der Studie ist die Identifikation von Herausforderungen und Spezifika des Fachbereichs Informatik in Hinblick auf Studierfähigkeit und Studienerfolg sowie die Markierung von Stellschrauben für einen höheren Studienerfolg. Erste Ergebnisse zeigen: Die Ursachen für einen Studienabbruch liegen v. a. in den Leistungsanforderungen und der zu bewältigende Stoffmenge.

Keywords: Ursachen Studienabbruch, Informatikstudium, Studierfähigkeit, Belastungen, Herausforderungen

1 Vorbemerkung

In Zeiten des Lehr- und Fachkräftemangels steht neben der schulischen auch die akademische Bildung im Fokus – nur ein geringer Teil der Informatikstudenten erlangt einen Studienabschluss. Laut Kultusministerkonferenz sind in den kommenden Jahren in den Lehrämtern Einstellungsbedarfe v. a. in den MINT-Fächern zu erwarten [KK18]. Aber auch in der IT-Branche gab es im vergangenen Jahr knapp 40.000 unbesetzte Stellen [AKP18]. Um dem Mangel an Lehr- und Fachkräften entgegenzuwirken, muss der Anteil der Absolventen gesteigert werden. Zur Entwicklung entsprechender Maßnahmen und zur effizienten Nutzung von Ressourcen sind gesicherte Kenntnisse über die Ursachen und Bedingungen des Studienabbruchs notwendig [He17]. Die Förderung der Studierfähigkeit scheint essentiell. Sie beschreibt den individuellen Umgang mit Herausforderungen des Studiums auf verschiedenen Ebenen. Bereits während der Schulzeit werden hierfür erste Grundsteine gelegt (Grundkompetenzen Verstehen und Schreiben) und im Rahmen des Studiums erfolgt die Weiterentwicklung (fachspezifische Inhalte und Transferleistungen) [Be16].

Ein Studienabbruch ist in der Regel auf eine Vielzahl von Faktoren zurückzuführen. Studienabbrecher scheitern am häufigsten an den hohen Studienanforderungen bzw. an den fehlenden fachlichen Voraussetzungen, aber auch die mangelnde Motivation ist zentral. Der dritthäufigste Abbruchgrund ist die Orientierung auf eine praktische Tätigkeit. Vermisst wird der Praxis- und Berufsfeldbezug im Studiengang. Eine Ausbildung erscheint als attraktive Alternative [He17].

¹ Universität Rostock, Zentrum für Lehrerbildung und Bildungsforschung, Doberaner Straße 115, 18057 Rostock, juliane.wegner@uni-rostock.de

² Universität Rostock, Zentrum für Lehrerbildung und Bildungsforschung, Doberaner Straße 115, 18057 Rostock, mona.amdt@uni-rostock.de

2 Untersuchung und Ergebnisse

Im Projektverlauf sollen die folgenden Fragen insbesondere für die Lehramtsstudenten beantwortet werden: Welche Ursachen und Bedingungen führen zum Studienabbruch? Inwieweit können aus diesen Erkenntnissen Maßnahmen abgeleitet werden, um den Studienerfolg im Fach Informatik zu erhöhen? In der Untersuchung werden Studenten des Faches Informatik der Universität Rostock befragt. Methodisch liegt dem Projekt eine schriftliche Datenerhebung mithilfe eines standardisierten Fragebogens zugrunde (Paper-Pencil-Befragung). Dieser beinhaltet sowohl Fragen zur Situation vor und während des Studiums als auch zur Wahrnehmung der Anforderungen, der Lehre und des eigenen Lern- und Studierverhaltens. Bisher gibt es drei Erhebungszeiträume, welche jeweils am Ende der Vorlesungszeit der einzelnen Semester liegen. Begonnen wurde im Wintersemester 2017/18 mit der Befragung der Erstsemesterstudierenden. Im Sommersemester 2018 erfolgte die zweite und im Wintersemester 2018/19 die dritte Erhebung.

Erste Auswertungen der Ergebnisse weisen darauf hin, dass etwa zwei Drittel der Studenten zumindest teilweise Sorge hat, das Studium erfolgreich abzuschließen. Die Leistungsanforderungen (45 %) und die Komplexität bzw. der Umfang der Lehrinhalte (44 %) wird von ihnen als stark belastend empfunden. Auch bevorstehende Prüfungen (36 %) können zur Überforderung führen. Weiter werden fehlende Kenntnisse hinsichtlich der universitären Organisation bzw. Strukturen (41 %) und mangelnde Studienmotivation (33 %) angeführt. Es zeigt sich, dass sowohl fachliche als auch überfachliche Kompetenzen und organisatorische Bedingungen des Studiengangs Informatik von den Studenten kritisch betrachtet werden. Die Gründe für einen Studienabbruch gestalten sich auch an der Universität Rostock mannigfaltig [He17]. Maßnahmen, die zur Verbesserung der Studiensituation beitragen, werden im Moment diskutiert. Hierzu zählen u. a. die Streckung eines Moduls von einem auf zwei Semester sowie eine bessere inhaltliche Abstimmung zwischen den Modulen. Nur durch verbesserte Studienbedingungen können die Ressourcen und Potentiale nutzbar gemacht werden. Nicht nur vor dem Hintergrund der zunehmenden Digitalisierung und den daraus resultierenden Veränderungen in den Bereichen Lernen, Leben und Arbeiten steht die universitäre Informatik-Ausbildung vor großen Herausforderungen.

3 Literaturverzeichnis

- [AKP18] Anger, C.; Koppel, O.; Plünnecke, A.: MINT-Frühjahrsreport 2018. MINT – Offenheit, Chancen, Innovationen. Köln, www.iwkoeln.de/fileadmin/user_upload/Studien/Gutachten/PDF/2018/MINT-Fr%C3%BChjahrsreport_2018neu.pdf, 07.02.2019.
- [Be16] Berk, I. et al.: Studierfähigkeit. Theoretische Erkenntnisse, empirische Befunde und praktische Perspektiven. Universitätskolleg-Schriften Band 15, Hamburg, 2016.
- [He17] Heublein, U. et al.: Zwischen Studiererwartungen und Studienwirklichkeit. Forum Hochschule, DZHW, www.dzhw.eu/pdf/pub_fh/fh-201701.pdf, 08.02.2019.
- [KK18] KK, Kultusministerkonferenz, Lehrereinstellungsbedarf und -angebot in der Bundesrepublik Deutschland 2018 – 2030. Zusammengefasste Modellrechnungen der Länder. www.kmk.org/fileadmin/Dateien/pdf/Statistik/Dokumentationen/Dok_216_Bericht_LEB_LEA_2018.pdf, 08.02.2019.

Workshops

Informatiksysteme für den Unterricht aufbereiten

Mike Barkmin,¹ Torsten Brinda²

Abstract: Dem Fach Informatik kommt eine immer größer werdende Bedeutung zu. Als Bezugswissenschaft für die Digitalisierung kann das Fach den Schüler*innen Kompetenzen vermitteln, um die von der Digitalisierung geprägten Welt verstehen und mitgestalten zu können. Eine wichtige Facette im Kompetenzaufbau ist die Förderung eines übergreifenden Verständnisses von Informatiksystemen, sodass sich Schüler*innen auch unbekannte Systeme erschließen können [Ge16]. Ein Informatiksystem ist dabei eine Zusammenstellung von Hardware-, und Software- und Netzwerkkomponenten. Die Herausforderung beim Erschließen von Informatiksystemen besteht darin, dass ein Großteil des Systems nicht direkt sichtbar ist und mit diesem auch nur indirekt agiert wird. Schüler*innen interagieren in der Regel sogar nur mit der Oberfläche eines Informatiksystems, z.B. bei der Benutzung einer Chat-App. Wichtige Komponenten wie der Transport einer Nachricht durch ein Netzwerk, die Verarbeitung dieser durch mehrere Server oder ihre Speicherung bleiben für Schüler*innen im Verborgenen. Eine Aufgabe von Informatikunterricht besteht darin, den Schüler*innen die nötigen Kompetenzen zu vermitteln, um das Verborgene zu verstehen. Stechert [St09] beschreibt zu diesem Zweck drei Kompetenzbereiche: (i) Aspekte des nach außen sichtbaren Verhaltens, (ii) Aspekte der inneren Struktur, die auf Strukturmodellen und vernetzten fundamentalen Ideen der Informatik basieren, (iii) ausgewählte Implementierungsaspekte zur Entwicklung einer konkreten Realisierung.

Im Workshop werden die Teilnehmer*innen zunächst ein Vorgehen zur Analyse von Informatiksystemen kennen lernen. Am Beispiel des Projekts Smartlights werden sie selbst aktiv und nehmen die Rolle der Schüler*innen ein. Die Auswahl der Werkzeuge richtet sich dabei nach den Bedürfnissen der Teilnehmer*innen. Anschließend wird gemeinsam über die Grenzen, Schwierigkeiten und Potentiale des Ansatzes diskutiert. Zum Abschluss werden dokumentierte reduzierte und rekonstruierte Informatiksysteme von Studierenden präsentiert, um den Teilnehmer*innen weitere Anregungen für den schulischen Einsatz zu geben.

Keywords: Informatiksystem; IOT; Didaktische Reduktion und Rekonstruktion

Literatur

- [Ge16] Gesellschaft für Informatik e. V. (Hrsg.): Bildungsstandards Informatik für die Sekundarstufe II. LOG IN/183/184, Beilage zum Heft, 2016.
- [St09] Stechert, P.: Fachdidaktische Diskussion von Informatiksystemen und der Kompetenzentwicklung im Informatikunterricht. Univ.verlag Potsdam, Potsdam, 2009.

¹ Universität Duisburg-Essen, Didaktik der Informatik, mike.barkmin@uni-due.de

² Universität Duisburg-Essen, Didaktik der Informatik, torsten.brinda@uni-due.de

Agile Schule – Weiterentwicklung des Projektunterrichts in der Informatik und darüber hinaus (Workshop)

Peter Brichzin¹, Petra Kastl² und Ralf Romeike³

Abstract: In Start-ups geht heute fast nichts mehr ohne Agile Softwareentwicklung. Auch in der Schule wird Projektunterricht dank agiler Methoden gewinnbringender und kann SchülerInnen zur Selbstorganisation anleiten. Der Workshop Agile Schule beschreibt wie [BRK19] aus der schulpraktischen Perspektive Hintergründe und Philosophie agiler Methoden, illustriert diese an konkreten Praxisbeispielen und stellt Umsetzungshinweise in Form eines praxiserprobten „Methodenkoffers“ zur Verfügung. Erfahrungen zeigen, dass SchülerInnen in agilen Projekten erworbene Selbstorganisationskompetenzen auch in anderen Fächern erfolgreich einsetzen.

Anliegen des Workshops ist es, für die TeilnehmerInnen das Potenzial agiler Methoden in schulischen Informatikprojekten konkret aufzuzeigen. So werden zu Beginn das iterative Vorgehen und typische agile Werte über ein Spiel erfahrbar gemacht. Danach werden zu ausgewählten grundlegenden agilen Praktiken jeweils der professionelle Hintergrund den Varianten schulischer Umsetzung, möglichen Stolpersteinen sowie Tipps und Tricks gegenübergestellt. Exemplarisch seien hier User-Stories genannt, die Anforderungen an eine Software aus Sicht des Kunden beschreiben. In Form von didaktischen Varianten können User-Stories auch eingesetzt werden, den Lernprozess zu strukturieren, indem diese Varianten über eine entsprechende Priorisierung in den Projektablauf eingeflochten werden und einen entsprechenden Zeitrahmen innerhalb der Iterationsplanung erhalten. Als Varianten verwenden wir einerseits Modeling-Stories, die einen konkreten Modellierungsauftrag z. B. die Erstellung eines Klassendiagramms enthalten. Modeling-Stories können von den SchülerInnen eines Teams bei Bedarf in ihre Planung aufgenommen werden. Ebenso können sie von der Lehrkraft vorgegeben werden, um eine Reflexion über Strukturen anzustoßen. Eine weitere didaktische Variante sind Student-Stories, die letztendlich Lernaufgaben darstellen. Der Mehrwert für den Unterricht der vorgestellten Methoden wird anhand praktischer Beispiele illustriert, sodass die TeilnehmerInnen für zukünftige Schulprojekte selbst agile Methoden auswählen und didaktisch anpassen können.

Keywords: Agile Methoden, Softwareentwicklung, Pair Programming, Projektunterricht.

Literaturverzeichnis

[BRK19] Brichzin, P.; Kastl, P.; Romeike, R: Agile Schule, hep-Verlag, Bern 2019.

¹ Gymnasium Ottobrunn, Karl-Stieler-Str. 1, 85521 Ottobrunn, peter@brichzin.de

² Gesamtschule Schenkklengsfeld, Dreienbergstraße 28-36, 36277 Schenkklengsfeld, kastl@gesamtschule-schenkklengsfeld.de

³ Freie Universität Berlin, Didaktik der Informatik, Königin-Luise-Str. 24-26, 14195 Berlin, ralf.romeike@fu-berlin.de

InstaHub Datenbanken und Datenschutz mit einem extra für den Unterricht entwickelten sozialen Netzwerk unterrichten.

Julian Dorn¹

Das Internet ist ein wesentlicher Bestandteil von Jugendlichen. Nach der JIM 2018 verbringen diese täglich durchschnittlich 3,5 Stunden online. Obwohl Datenbanken der Datenspeicher fast jeder Webseite sind, ist kaum ein/e SchülerIn motiviert sich mit diesen zu beschäftigen. Nach endlosen Skandalen bei Facebook veröffentlichen Jugendliche noch immer unbekümmert intimste Informationen in sozialen Netzwerken. InstaHub ändert das! Anstelle nur von Gefahren und Funktionsweisen zu reden, lassen sie sich hiermit tatsächlich erleben. Nach acht bundesweiten Workshops und über 300 angemeldeten LehrerInnen ist InstaHub.org jetzt auch auf der INFOS.

InstaHub.org ist ein voll funktionsfähiges soziales Netzwerk, welches sehr ähnlich der bekannten Plattform Instagram ist. Größter Unterschied ist, dass SchülerInnen hier nicht einfach nur Benutzer, sondern Administratoren sind. So bietet schon die Webseite deutlich mehr Funktionen: Benutzer und Fotos können gelöscht oder verändert werden und Details wie das Geburtsdatum eingesehen werden. Ergänzend dazu ermöglicht eine Extraseite den direkten Zugriff auf die Datenbank. Als Administrator hat der oder die SchülerIn mithilfe einer intuitiven SQL-Eingabe und einer graphischen Abfragesprache die volle Gewalt über sein eigenes soziales Netzwerk.

InstaHub kann auf Wunsch um ein eigenes Benutzertracking und fiktive personalisierte Werbung erweitert werden.

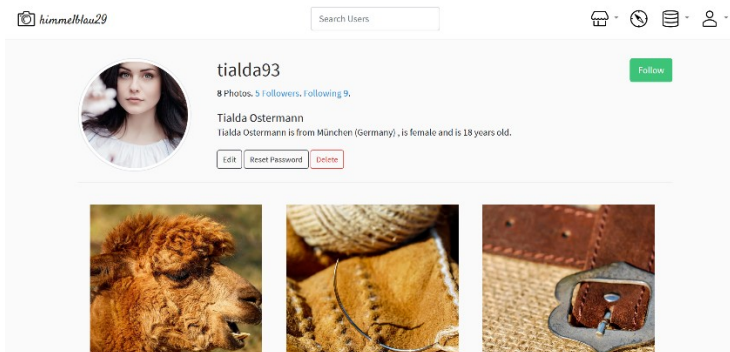


Abb. 1: Beschreibung der Abbildung

¹ Wissen, Leipzig, dorn@wi-wissen.de

JavaScript als ideale textbasierte Programmiersprache im Unterricht anhand von ada7 differenziert und motivierend unterrichten.

Julian Dorn¹

JavaScript ist eine sehr gut für den Informatikunterricht geeignete Programmiersprache. Mit ihr lassen sich Programme entwickeln, welche zu Beginn motivierende Szenarien abbilden und später in Form von Webseiten, Apps oder Spielen direkt der Lebenswirklichkeit der SchülerInnen entsprechen. Dadurch dass der Unterricht vollständig im Browser durchgeführt werden kann, entfallen Wartungsaufwände in der Schule und alle SchülerInnen können auch zu Hause üben.

Zum Erlernen von JavaScript wird die vom Autor entwickelte und kostenfrei zur Verfügung stehende Plattform ada7.de vorgestellt:

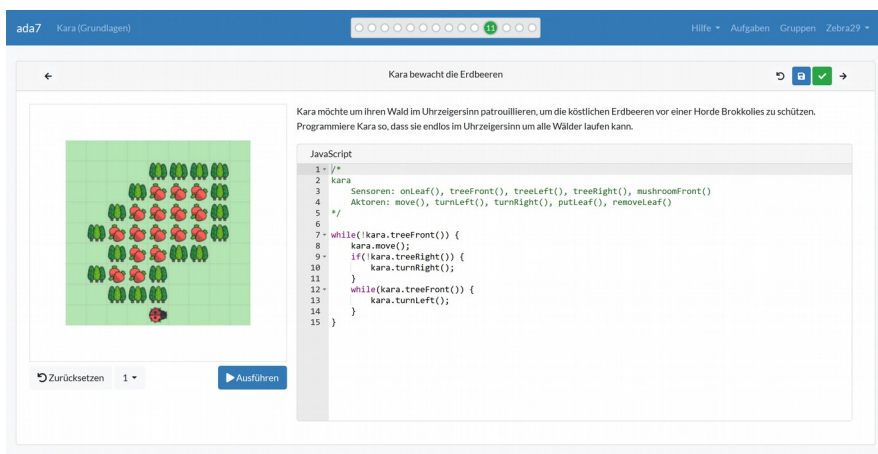


Abb. 1: Beschreibung der Abbildung

Es wird Blended Learning als Methode zum Lehren vorgestellt, womit eine individuelle Förderung auch bei heterogenen SchülerInnengruppen möglich ist. Dadurch kann ein reibungsloser Unterricht gestaltet werden, in welchem alle SchülerInnen einen maximalen Lernerfolg erreichen können, ohne dass für Einzelne Pausen entstehen. So wird vermieden, dass SchülerInnen sich langweilen und so der Lernerfolg und die Disziplin gestört wird. Die Lehrkraft behält dank eines Dashboards dennoch immer die Übersicht über den Lernfortschritt Einzelner.

¹ Wissen, Leipzig, dorn@wi-wissen.de

„Wie funktioniert das?“ - Informatische Konzepte in der Vor- und Grundschule spielerisch begreifen und anwenden

Anja Gärtig-Daugs¹, Alexander Werner², Ute Schmid³

Damit Kinder zu souveränen Akteuren in einer zunehmend von Digitalisierung geprägten Welt werden, bedarf es zielgerichteter Bildungsangebote [SG18]. Im Workshop werden die von der Forschungsgruppe Elementarinformatik entwickelten Lehr-Lern-Konzepte der Experimentierkiste Informatik vorgestellt, die anschließend in Kleingruppen erprobt werden können. Die Materialien der Experimentierkiste sind handlungsorientiert gestaltet und regen zum forschenden und entdeckenden Lernen an. Sie ermöglichen Kindern, sich grundlegende Funktionsprinzipien digitaler Medien zu erschließen und diese kreativ zu nutzen [SWK18].

Ausgehend vom kindlichen Interesse und der Erfahrungswelt der Kinder wurden bislang Konzepte zu folgenden Oberthemen entwickelt:

- *Digitale Repräsentation* (Wie stellt ein Computer Bilder dar?) mit den Lerneinheiten „Pixel“, „Analog und digital“, „TuxPaint – Bilder am Computer malen, speichern, drucken“
- *Algorithmen* (Was ist ein Algorithmus?) mit den Einheiten „Such-/Sortieralgorithmen“, „Algorithmen im Alltag“, „Programmieren mit ScratchJr/Scratch“

Literaturverzeichnis

- [SG18] Schmid, U.; Gärtig-Daugs, A.: Notwendigkeit der Integration elementarinformatischer Lerneinheiten in den Vor- und Grundschulunterricht. In (Bastian, J. Hrsg.): MedienPädagogik. Zeitschrift für Theorie und Praxis der Medienbildung. Themenheft Nr. 31: Digitale Bildung - Medienbezogene Bildungskonzepte für die nächste Gesellschaft, S. 78-106, 2018.
- [SWG18] Schmid, U.; Weitz, K.; Gärtig-Daugs, A.: (2018). Informatik in der Grundschule. Eine informatisch-pädagogische Perspektive auf informatikdidaktische Konzepte. Informatik-Spektrum 41/3, S. 200-207, 2018.

¹ Otto-Friedrich-Universität Bamberg, Forschungsgruppe Elementarinformatik/Kompetenzzentrum für digitales Lehren und Lernen, Luitpoldstraße 19, 96052 Bamberg, anja.gaertig-daug@uni-bamberg.de

² Otto-Friedrich-Universität Bamberg, Forschungsgruppe Elementarinformatik/Lehrstuhl für empirische Bildungsforschung, Markusstraße 8a, 96045 Bamberg, alexander.werner@uni-bamberg.de

³ Otto-Friedrich-Universität Bamberg, Forschungsgruppe Elementarinformatik/Professur für Angewandte Informatik, insb. Kognitive Systeme, 96045 Bamberg, ute.schmid@uni-bamberg.de

Workshop: Internet of Things zum Nachbauen

Laura Hembrock,¹ Patrick Tschorn,² Michael Brinkmeier³

Abstract: Das Lebensumfeld von Schülerinnen und Schülern wird heutzutage immer mehr vernetzt. Wo früher noch Einkaufslisten per Hand geschrieben werden mussten, kann diese Aufgabe nun Assistenten wie Alexa, Cortana und Siri überlassen werden, oder aber der Kühlschrank sendet einfach selbst ein Bild seines Inhaltes in den Supermarkt.

Der Workshop „IoT zum Nachbauen“⁴ bietet eine gute Gelegenheit ein Projekt durchzuführen, bei dem die Schülerinnen und Schüler ein Beispiel für die Vernetzung von Informatiksystemen selber erfahren können.

Im Laufe des Workshops soll von den Teilnehmenden Stadiontechnik für einen Tischkicker implementiert werden. Dazu gehören z.B. Toranzeigen für die beiden gegnerischen Teams und eine Anzeige für die verbleibende Spielzeit. Gesteuert wird die Stadiontechnik von einem Schiedsrichter mithilfe eines Tablets. Von diesem wird ein Calliope Mini mit HTTP-Requests angesteuert, der wiederum ein entsprechendes Signal per Funk an die Mikrocontroller mit den einzelnen Funktionen der Stadiontechnik sendet.

Der Workshop beginnt mit einer kurzen Einführung zu abbozza! und dem Calliope Mini. Während der gesamten Zeit wird die Methode des Pair Programming verwendet.

Nach der Einführung treffen sich alle Teilnehmenden, um gemeinsam die gewünschten Funktionen zu besprechen. Hier ist eine Variation von Gruppe zu Gruppe möglich, da die Teilnehmenden sehr frei entscheiden können, was implementiert werden soll. Stehen die umzusetzenden Funktionen fest, wird mit den Teilnehmenden über die Notwendigkeit eines Kommunikationsprotokolls diskutiert. Im Anschluss an die Diskussion wird mit den Teilnehmenden ein Kommunikationsprotokoll entwickelt und die einzelnen Funktionen der Stadiontechnik auf die Programmier-Teams aufgeteilt. Ein Team bekommt dabei die Aufgabe den Master-Calliope, welcher die HTTP-Requests empfängt, zu programmieren und ein weiteres soll die App für das Tablet des Schiedsrichters mit dem MIT App-Inventor erstellen.

Während die Programmier-Teams nach der Besprechung in die Implementierung ihrer Funktionen starten, bekommt das App-Team zuerst eine Einweisung in die Arbeit mit dem App-Inventor. Sind alle Teams mit ihren Aufgaben fertig, wird die Stadiontechnik mit einem Spiel am Tischkicker getestet.

Keywords: Internet of Things; Calliope Mini; graphische Programmierung; Mittelstufe; K-12 education

¹ Universität Osnabrück, Institut für Informatik, Wachsbleiche 27, 49090 Osnabrück lhembrock@uos.de

Die Autorin wird im Rahmen des PhyCoS-Projektes von der Stiftung Stahlwerk Georgsmarienhütte, der Stiftung der Sparkassen im Landkreis Osnabrück und der Stiftung der Sparkasse Osnabrück gefördert.

² Universität Osnabrück, Institut für Informatik, Wachsbleiche 27, 49090 Osnabrück patschorn@uos.de

³ Universität Osnabrück, Institut für Informatik, Wachsbleiche 27, 49090 Osnabrück mbrinkmeier@uos.de

⁴ Die Materialien sind unter <https://tinyurl.com/workshopus> zu finden.

Quadrologik – Modularisierung von Schaltnetzen und Schaltwerken

Tobias Jördens¹, Jens Gallenbacher²

Abstract: Aufbauend auf dem von uns auf der INFOS2017 vorgestellten System Quadrologik wurde ein Programmiergerät entwickelt, mit dem Schaltnetze und Schaltwerke modularisiert werden können.

Der Grundgedanke für die hier vorgestellte Erweiterung von Quadrologik ist, dass durch Legen erstellte (und als solche getestet) Schaltungen von einem Programmiergerät erfasst werden, um diese dann in ein Logikmodul zu übertragen. Dieses Logikmodul simuliert die gelegte Schaltung und kann von da an in weiteren Schaltungen genutzt werden. Auf diese Weise können auch mehrere identische Module generiert werden, um zum Beispiel komplexere Schaltungen zu realisieren.

Im Workshop wird dieses Verfahren vorgestellt und anhand von Beispielen die Möglichkeiten und didaktischen Ideen zur Modularisierung dargestellt. Da die Modularisierungstiefe von Schaltwerken im momentanen Modell noch endlich ist, wollen wir im Workshop über die didaktischen Möglichkeiten und Notwendigkeiten diskutieren.

Keywords: Modularisierung, Modellbildung, Technische Informatik, Digitaltechnik, Logikgatter, Logikstecksystem, Logische Schaltungen, Rechnertechnik

¹ Schuldorf Bergstraße und Technische Universität Darmstadt, Didaktik der Informatik, Hochschulstraße 10, 64289 Darmstadt, tj@di.tu-darmstadt.de

² Technische Universität Darmstadt, Didaktik der Informatik, Hochschulstraße 10, 64289 Darmstadt, jg@di.tu-darmstadt.de

Mehr als Programmieren lernen - einen Mikrocontroller-Roboter zum Leben erwecken

Zweitägiges Workshopkonzept zur Mädchenförderung mit dem B-O-B-3

Andrea Langer¹, Nadine Bergner² und Ulrik Schroeder¹

Abstract: Da die Informatik auch 2019 weit von der gewünschten Frauenquote entfernt ist, bedarf es innovativer Ideen und neuartiger Zugänge, um Mädchen an die Informatik und insbesondere an die Programmierung heranzuführen. Im Rahmen des seit 2008 laufenden Mädchenförderprojektes go4IT!³ der RWTH Aachen wurde ein bereits etablierter Robotik-Workshop mit Lego Mindstorms durch ein Workshop-Programm mit dem Roboter B-O-B-3 ergänzt. Aufbauend auf den Erfahrungen aus über 250 go4IT! [Le15] wurde ein Workshop zum Informatikeinstieg konzipiert, der Themen wie Aufbau eines Informatiksystems, Sensorik und Binärsystem ebenso behandelt wie den Einstieg in die Algorithmik und Programmierung. Durchgeführt wird der neue B-O-B-3-Workshop (wie auch der bisherige Mindstorms-Workshop) mit 12-14 Mädchen der Klassenstufe 6 bis 8 an zwei aufeinanderfolgenden Tagen direkt in den Schulen, um die Einstiegshürde für die Mädchen möglichst gering zu halten. Geleitet werden die Workshops von zwei (hilfs-)wissenschaftlichen Mitarbeiterinnen aus dem MINT-Bereich, die gleichzeitig als Role Models dienen. Das neue Mädchenförderkonzept hat dabei drei besondere Stärken: (1) Die Montage des Roboters durch die Mädchen geschieht mittels Löten der einzelnen Komponenten auf die vorgefertigte Platine, was im Gegensatz zum eher spielerisch wirkenden Aufbau eines Lego Mindstorms ein hohes Maß an Professionalität bewirkt. (2) Durch den kompetenzorientierten Wechsel der Programmiersprachen von der grafischen Sprache Ardublockly zur textuellen Programmierung mittels Arduino gelingt die Kombination aus weichem Einstieg für Novizinnen und dem professionellen Endergebnis. (3) Die Identifikation mit dem Informatiksystem des B-O-B-3 und die Nachhaltigkeit der einmaligen Maßnahme werden durch den Übergang der Roboter in den Besitz der Mädchen verstärkt. Das neue Workshop-Konzept wurde bisher mit 8 Gruppen an verschiedenen Schulen in NRW erprobt. Das Feedback der Mädchen am Ende der Workshops war durchweg gut und nach ihrer eigenen Einschätzung eine Fortführung der Programmierung sehr wahrscheinlich. Insbesondere positiv erwähnt wurde dabei die Löterfahrung und die Durchführung als genderspezifische Maßnahme. Trotz und vielleicht auch gerade wegen einiger Schwierigkeiten und Hürden beim Umstieg auf die textuelle Programmierung, sehen die Mädchen die Bewältigung derselben meist als persönliches Erfolgserlebnis und als durchaus positive Erfahrung.

Literaturverzeichnis

[Le15] Leonhardt, T.: Etablierung eines begabungsfördernden Lernumfeldes für Mädchen im Bereich der Informatik. RWTH Aachen, Aachen, 2015.

¹ RWTH Aachen, Lehr- und Forschungsgebiet Informatik 9, Ahornstr. 55, 52074 Aachen, a.langer@informatik.rwth-aachen.de bzw. schroeder@informatik.rwth-aachen.de

² TU Dresden, Professur für Didaktik der Informatik, Nöthnitzer Str. 46, 01187 Dresden, nadine.bergner@tu-dresden.de

³ <http://www.schuelerlabor.informatik.rwth-aachen.de/go4it>

Über Lottofeen zu (mehr) Computernerds

Undoing Gender im Informatikunterricht durch eine selbstreflektierte Konfrontation mit geschlechtsspezifischen Stereotypisierungen

Kris Markussen, Kirsten Alich, Jana Gerdes, Mareike Daeglau, Nils Pancratz, Ira Diethelm¹

Extended Abstract: Lehrende in der Informatik werden trotz aller Aufgeschlossenheit ggü. Gender- und Diversity-Themen von der in diesem Fachbereich präsenten männlich konnotierten Fachkultur, unreflektierten Machtverhältnissen, fachspezifischen Verhaltensmustern und vorherrschenden geschlechtsspezifischen Normen- und Wertesystemen beeinflusst. Diese inhärenten gesellschaftlichen Strukturen erschweren eine gleichberechtigte Teilhabe aller Geschlechter am Informatikunterricht. Eine weitere Rolle spielen die in der jeweiligen Fachkultur herrschenden (männlichen) Selbstverständnisse und die im Sinne des „Doing Gender“ hergestellten Geschlechterordnungen. Bath fordert in diesem Kontext eine Auseinandersetzung nicht nur mit Frauen in MINT, sondern v. a. mit Gender Studies in MINT im Sinne der „Gender Science“. Sie schlägt u. a. eine „Reflektion des eigenen Anteils an binären Geschlechterkonstruktionen“ [Ba15, 111] in MINT vor. An dieser Stelle setzen wir mit unserem Workshop an. Mit dem an das Identitätenlotto² angelehnten Spiel soll unter Berücksichtigung der Vorerfahrungen der Teilnehmenden zunächst ein Perspektivwechsel vollzogen werden. Anhand expl. Fallbeispiele werden wir Perspektiven anderer Identitäten betrachten und uns mit eigenen Vorurteilen, Zuschreibungen und Handlungsmustern konfrontieren und unser Unterrichtsverhalten reflektieren. Dazu werden im Workshop folgende Fragen in Bezug auf die betrachteten Identitäten gestellt: Welches genderspezifische Selbstverständnis herrscht in der Fachkultur der Informatik? Welche Expert*innen repräsentieren auf welcher Grundlage die Fachkultur und welche Ein- und Ausschlüsse sind damit verbunden? Wie werden Hierarchien, Gleichberechtigung und Teilhabe gestaltet? Welche Mitgestaltungsmöglichkeiten haben Lernende? Aus dieser Reflexion sollen anschließend konstruktive Lösungsansätze entwickelt werden, die in einem gendersensitiven Informatikunterricht angewandt werden können.

Keywords: Gendersensitiver Informatikunterricht; Diversität; reflexives unterrichtliches Handeln

Anmerkung: Dieser Workshop wurde im Rahmen des Projekts *smile* (www.smile-smart-it.de), das vom BMBF unter den Förderkennzeichen 01FP1611 bis 01FP1615 gefördert wird, konzipiert.

Literatur:

[Ba15] Bath, C.: Sensibilisierung von Lehrenden, aber wofür? Von „Frauen in MINT“ zu „Gender Studies in MINT“. In (Augustin-Dittmann, S.; Gotzmann, H., Hrsg.): MINT gewinnt Schülerinnen. Erfolgsfaktoren von Schülerinnen-Projekten in MINT. VS, Wiesbaden, S. 111–126, 2015.

¹ Universität Oldenburg, Didaktik der Informatik, Uhlhornsweg 84, 26129 Oldenburg, vorname.nachname@uol.de

² Beim *Identitätenlotto* (© spielecht Spielverlag & Juliette Wedl, www.identitaetenlotto.de) handelt es sich um ein Lehr-/Lernspiel, das die geschlechtliche Vielfalt und damit verbundene Diskriminierung und Mobbing aufzeigt.

Pair Programming im Klassenraum

Eine Digital Game-based Methode für die Sekundarstufe I

Maximilian Marowsky¹, Paul Ohm², Julian Dierker³

Abstract: PearUp ist eine digitale Lernplattform, die im Rahmen eines Studierendenprojekts der Universität Osnabrück entwickelt wird. Die Lernplattform unterstützt Informatiklehrer dabei und befähigt fachfremde Lehrkräfte dazu, qualitativ hochwertigen Informatikunterricht anbieten zu können, um Schülern der Sekundarstufe I spielerisch die Grundlagen der Informatik zu vermitteln. Der Ansatz der Lernplattform PearUp ist der des Digital Game-Based Learning: Im Rahmen einer Spielhandlung gründen Schüler in Zweiergruppen ein virtuelles IT-Start-Up, das sich auf das Lösen gegenwartsnaher Informatikaufgaben spezialisiert hat. Die Aufgaben werden in Form von Aufträgen verschiedener Firmen im Bereich der Jobbörse angeboten. Anhand dieser realitätsnahen Aufträge lernen die Schüler die Grundlagen des Programmierens kennen und erleben zugleich, wie aufregend und vielseitig die Arbeitswelt der Informatik sein kann. Erledigte Aufträge generieren virtuelle Einnahmen wie Geld, Erfahrungspunkte und Auszeichnungen. Mit steigender Kompetenz und Stufe erhöht sich die Anzahl der verfügbaren Aufträge sowie die Möglichkeiten das eigene Startup zu personalisieren, indem z.B. Arbeitsplatz und -ort frei gewählt werden können. Das erwirtschaftete Geld hingegen kann in verschiedene Upgrades investiert werden, um weitere Spielmechaniken freizuschalten, wie Werbung zu schalten, mit Auftraggebern verhandeln oder kollaborative Großaufträge mit anderen Kleingruppen bearbeiten zu können. PearUp setzt dabei auf die Arbeitsmethode des Pair Programming, bei der zwei Schüler zusammen am selben Endgerät arbeiten, um eine höhere Qualität zu erreichen. Der Name der Lernplattform spiegelt also den Kerngedanken des Projekts wider, dass gemeinsames Programmieren sowohl Freude bereitet als auch effektiv und lehrreich ist. Die Lehrkraft behält dabei stets den Überblick über all ihre Klassen und deren Gruppen, kann Aufgabenpakete auswählen oder flexibel zusammenstellen, direkt oder indirekt Hilfestellung leisten und sowohl die Leistungen jeder Gruppe als auch die individuellen Leistungen der einzelnen Schüler nachverfolgen und den Lernprozess so gezielt lenken. Unterstützt wird sie dabei von modernen Algorithmen der Learning Analytics. Das erste Aufgabenpaket „Einstieg in die Algorithmik“ kann bereits ausführlich getestet werden. Weitere Aufgabenpakete zu den Themen „Datensicherheit“, „Objektorientiertes Programmieren“ und „Maschinelles Lernen“ befinden sich derzeit im Rahmen von Abschlussarbeiten an der Universität Osnabrück in Entwicklung. Zusätzlich soll die Funktionalität um ein Aufgabenerstellungs-Tool erweitert werden, das Lehrkräften eine Entwicklungsumgebung bietet, mit der sie, ihren Wünschen entsprechend, Aufgaben erstellen können.

¹ Universität Witten/Herdecke, Psychologie, Martinistr. 46, 49078 Osnabrück, Maximilian.Marowsky@uni-wh.de

² Universität Osnabrück, Lehramt, Martinistr. 46, 49078 Osnabrück, pohm@uos.de

³ Universität Osnabrück, Lehramt, Ernst-Sievers-Straße 76a, 49078 Osnabrück, jdierker@uos.de

Workshop: Maschinelles Lernen Unplugged

Elisaweta Ossovski,¹ Michael Brinkmeier²

Abstract: Künstliche Intelligenz, bei der Maschinelles Lernen als Grundtechnik eine wichtige Rolle spielt, wird immer häufiger in den Medien thematisiert und bewertet, ohne auf die konkrete Funktionsweise einzugehen. Dadurch kann gerade bei Schülerinnen und Schülern ein falsches Bild davon entstehen. Der bereits mit einigen Gruppen erprobte Workshop³ „Maschinelles Lernen Unplugged“ stellt dabei einen ersten Versuch dar, wie Schülerinnen und Schülern das Teilgebiet Maschinelles Lernen mit einem Unplugged-Konzept näher gebracht werden kann.

Im Anschluss an einen kurzen theoretischen Impuls, der eine Motivation für die handlungsorientierte Phase gibt, sollen die Teilnehmenden mithilfe einer Pinnwand, Pinnnadeln sowie einer Holzleiste eine lernende Maschine entwickeln, die zwei Arten von Schrauben klassifizieren kann. Dabei fällt zur Datengenerierung eine zu klassifizierende Schraube zufällig auf eine bewegliche Platte und wird von oben fotografiert. Auf diesem Bild wird eine Bounding Box bestimmt und die Maße dieses Rechtecks stellen mit dem jeweils größeren Wert als erste Vektorkomponente und dem jeweils kleineren Wert als zweite Vektorkomponente die Daten dar.

Beim Lernprozess werden 20 zufällige Datenkarten aus dem Datensatz einzeln auf der Pinnwand markiert und die Holzleiste entsprechend in Richtung des eingefügten Punktes bewegt, wobei die Holzleiste, die zu Beginn an einem Punkt an der x-Achse fixiert wurde, um eine immer geringere Distanz gedreht wird. Dies entspricht der graphischen Repräsentation eines Perzeptrons, wobei das Lernverfahren zugunsten der Handhabbarkeit vereinfacht wurde. Danach präsentieren die Teilnehmenden jeweils den anderen Gruppen kurz ihr Ergebnis. Bei der Vorstellung sollen diese auch ihre Gedanken während des Durchführens erläutern und die Fragen beantworten, welche Überlegungen zur Wahl der initialen Gerade geführt haben und ob sie eine Fortsetzung des Verfahrens für sinnvoll halten. Die Intention dabei ist die Erkenntnis der Vorteile eines Rechners, um Präzision gewährleisten und auch kleine Veränderungen sinnvoll durchführen zu können. Diese Eigenschaften kann die handlungsorientierte Darstellung nicht erfüllen, da eine Bewegung der Holzleiste um wenige Millimeter keine bedeutenden Ergebnisse hervorrufen würde. Die unterschiedlichen Ergebnisse der Gruppen können gegebenenfalls durch weitere auf Bildern dokumentierte Ergebnisse ergänzt und zum Anlass genommen werden, um die Auswirkungen verschiedener Parameter wie die Wahl der initialen Gerade sowie die Auswahl und Anzahl der Datenkarten zu diskutieren. Den Abschluss des Workshops bildet eine Diskussion über weitere Fälle wie andere Größenverhältnisse der Schraubenarten, nicht linear separable Daten sowie die Möglichkeiten mehr als nur zwei Arten zu klassifizieren.

Keywords: Maschinelles Lernen; Linearer Klassifizierer; Unplugged; Oberstufe; K-12 education

¹ Universität Osnabrück, Institut für Informatik, Wachsbleiche 27, 49090 Osnabrück, ossovski@uni-osnabrueck.de

Die Autorin wird im Rahmen eines Promotionsstipendiums vom Ernst Ludwig Ehrlich Studienwerk gefördert.

² Universität Osnabrück, Institut für Informatik, Wachsbleiche 27, 49090 Osnabrück, mbrinkmeier@uni-osnabrueck.de

³ Die Materialien sind unter <https://tinyurl.com/workshopus> zu finden.

Wie gelingt eine obligatorische Weiterbildung für tausende von Lehrpersonen der Volksschule? Ein Erfahrungsbericht mit erlebbaren Praxisteilen aus dem Grundlagenkurs Medien und Informatik.

Tobias M. Schifferle¹, Eike Rösch¹, Peter Suter¹

Keywords: Lehrplan, Informatikdidaktik, Grundschule, Volksschule, Ausbildung, Weiterbildung.

2017 wurden mit dem «Lehrplan 21» Informatik und Medienbildung als obligatorische Inhalte in der Volksschule der deutschsprachigen Kantone der Schweiz verankert. Seitdem werden die Themen ab dem Kindergarten integriert und ab der 5. Klasse als eigenständiges Fach «Medien und Informatik» mit einer Wochenstunde unterrichtet. Damit Lehrerinnen und Lehrer das Fach im Kanton Zürich unterrichten dürfen, müssen sie in ihrer unterrichtsfreien Zeit eine 90-stündige Weiterbildung, den Grundlagenkurs Medien und Informatik» (GMI), an der Pädagogischen Hochschule Zürich besuchen. Bis 2021 werden gut 3'000 Lehrpersonen, getrennt nach Niveau Primar- oder Sekundarschule, den GMI absolvieren.

Nach Abschluss des ersten Kursjahres wurde evaluiert, wie bereit sich die Lehrpersonen nach dem GMI fühlen, das neue Fach zu unterrichten. Mit n=273 Antwortenden (70% weiblich) betrug die Rücklaufquote 50%. Davon sind 91% der Lehrpersonen (75% davon stark und sehr stark) der Meinung, der GMI habe ihr Interesse, das Fach zu unterrichten, geweckt. 91% trauen sich nach dem Besuch des GMI zu, das Fach Medien und Informatik angemessen zu unterrichten. Knapp die Hälfte der Befragten wünscht sich zurzeit keine zusätzliche Weiterbildung. 39% der Lehrpersonen möchten eine Weiterbildung in 1-2 Jahren, während je 13% entweder gar keine oder möglichst schnell weitere Weiterbildungen wünschen. Wenn Weiterbildungen gewünscht werden, dann sollen sie für eine Mehrheit (88%) kürzer als 8 Stunden sein.

Die Resultate der Befragung zeigen, dass das Ziel des GMI, die Lehrpersonen in dieser ersten Phase der Einführung des Faches unterrichtsfähig zu machen, erreicht worden ist.

Im Workshop wird berichtet, wie es gelungen ist, diese sehr grosse, heterogene und beim Kursstart meist informatik-unerfahrene Gruppe von Volksschullehrpersonen für den Informatikunterricht zu motivieren und ihnen das nötige Start-Know-How und Vertrauen in die Informatik und -didaktik zu vermitteln. Die Workshopeteilnehmenden erleben Activities aus der Weiterbildung und erhalten zusammen mit Hintergrundinformationen zu Aufbau und Didaktik ein Bild dieser handlungsorientierten Weiterbildung, welches anschliessend mit den Teilnehmenden diskutiert wird.

¹ Pädagogische Hochschule Zürich, Zentrum Medienbildung und Informatik, Lagerstrasse 2, Zürich, 8090, {tobias.schifferle, eike.roesch, peter.suter}@phzh.ch

Maschinelles Lernen im Unterricht mit Jupyter Notebook

Michael Schlichtig, Simone Opel, Carsten Schulte¹, Rolf Biehler, Daniel Frischemeier, Susanne Podworny, Thomas Wassong²

Keywords: Data Science; Machine Learning; KI; Künstliche Neuronale Netze; Entscheidungsbäume; Decision Trees; Big Data; Workshop; Maschinelles Lernen; Artificial Neural Networks; Python; Jupyter Notebook

Data Science und Big Data durchdringt in ihren diversen Facetten unser tägliches Leben – kaum ein Tag, an dem nicht verschiedene Meldungen über technische Innovationen, Einsatzmöglichkeiten von Künstlicher Intelligenz (KI) und Maschinelles Lernen (ML) und ihre ethischen sowie gesellschaftlichen Implikationen in den unterschiedlichen Medien diskutiert werden. Aus diesem Grund erscheint es uns immens wichtig, diese Fragestellungen und Technologien auch in den Unterricht der Sekundarstufe II zu integrieren.

Um diesem Anspruch gerecht zu werden, entwickelten wir im Rahmen eines Forschungsprojekts ein Curriculum, welches wir als konkretes Unterrichtskonzept innerhalb eines Projektkurses erprobt, evaluiert weiterentwickelt wird. Bei der Implementierung entschieden wir uns, zur aktiven Umsetzung von Konzepten von ML als Plattform Jupyter Notebook mit Python zu verwenden, da diese Umgebung durch die Verbindung von Code und Hypertext zur Dokumentation und Erklärung Medienbrüche im Lernprozess verringern kann. Zudem ist Python zur Implementierung der Methoden von ML sehr gut geeignet.

Im Themenfeld des ML als Teilgebiet der KI legen wir den Fokus auf zwei unterschiedliche Lernverfahren um verschieden Aspekte von ML, u.A. wie Nachvollziehbarkeit unter gesellschaftlichen Gesichtspunkten zu vermitteln. Diese sind *Künstliche Neuronale Netze* (bei denen die Berechnung und Bedeutung der Kantengewichte zwischen den Neuronen für den Menschen insbesondere bei komplexeren Netzen kaum nachvollziehbar erschienen) und *Entscheidungsbäume* (strukturierte und gerichtete Bäume zur Darstellung von Entscheidungsregeln, welche auch für Schülerinnen und Schüler meist gut nachvollziehbares und verständliches KI-Modell darstellen). In diesem Workshop stellen wir konkrete Umsetzungsbeispiele inklusive der Programmierung für beide Verfahren mit Jupyter Notebook und Python als Teil einer Unterrichtssequenz vor und diskutieren diese.

¹ Universität Paderborn, Didaktik der Informatik, Fürstenallee 11, 33102 Paderborn, michael.schlichtig@uni-paderborn.de, simone.opel@uni-paderborn.de, carsten.schulte@uni-paderborn.de

² Universität Paderborn, Didaktik der Mathematik, Warburger Str. 100, 33098 Paderborn, biehler@math.uni-paderborn.de, dafr@math.uni-paderborn.de, podworny@math.uni-paderborn.de, wassong@math.uni-paderborn.de

Künstliche Intelligenz im Informatikunterricht der Sek II – Unterrichtsmodul “Lernende Algorithmen”

Pascal Schmidt¹, Stefan Strobel²

Abstract: Der Workshop zeigt einen unterrichtspraktischen Weg auf, wie in das Thema “Lernende Algorithmen” am Beispiel von Entscheidungsbäumen im Informatikunterricht der Sekundarstufe II eingeführt werden kann.

Keywords: Entscheidungsbäume, allgemeinbildender Informatikunterricht, Sekundarstufe II.

Inhaltsüberblick

Im unplugged-Teil werden der Ablauf der Klassifikation nachvollzogen und von Hand (unterschiedliche) Entscheidungsbäume konstruiert [Er08]. Die Mehrdeutigkeit dieser Konstruktion leitet schrittweise zu einer rekursiven Baumerzeugung über, bei der ein gieriger Algorithmus zur Attributauswahl und der gewichtete Gini-Index als Attributauswahlmaß genutzt werden [BV08], [BK98].

Im plugged-Teil besteht die Möglichkeit, anhand bereitgestellter Codevorlagen die Methode zur Baumerzeugung zu implementieren. Zudem kann der Einfluss von Datenduplikaten bzw. widersprüchlichen Daten untersucht und der Algorithmus weiterentwickelt werden. Anhand von Beispielen kann zudem Einsicht in die Grenzen der gierigen Attributauswahl gewonnen werden [LS19].

Literatur

- [BV08] Bankhofer, U.; Vogel, J.: Datenanalyse und Statistik. Gabler-Verlag, Wiesbaden, 2008.
- [BK98] Borgelt, C.; Kruse, R.: Attributauswahlmaße für die Induktion von Entscheidungsbäumen: Ein Überblick. In: Data Mining: Theoretische Aspekte und Anwendungen. Physica-Verlag, Heidelberg, S. 77-98, 1998.
- [Er08] Ertel, W.: Grundkurs Künstliche Intelligenz: Eine praxisorientierte Einführung. Vieweg-Verlag, Wiesbaden, 2008.
- [LS19] Lambert, A.; Schmidt, P.: Algorithmen im Mathematik- und Informatikunterricht. In: Der Mathematikunterricht, Heft 65(4). Friedrich-Verlag, Seelze, 2019.

¹ Universität des Saarlandes, Campus, Geb. E 2 4, 66123 Saarbrücken, pascal.schmidt@uni-saarland.de

² Gymnasium am Rotenbühl, Neugrabenweg 66, 66123 Saarbrücken, stefan-strobel@posteo.eu

Autorenverzeichnis

A

Alich, Kirsten, 381
Arndt, Mona, 369

B

Barkmin, Mike, 373
Baum, Kevin, 49
Berges, Marc, 161
Bergner, Nadine, 380
Best, Alexander, 59, 237
Biehler, Rolf, 285, 385
Borowski, Christian, 237
Brämer, Martin, 367
Brichzin, Peter, 374
Brinda, Torsten, 25, 373
Brinkmeier, Michael, 378, 383
Brüggen, Niels, 25

C

Capovilla, Dino, 35
Chytas, Christos, 295

D

Daeglau, Mareike, 295, 381
Dierker, Julian, 382
Diethelm, Ira, 23, 25, 69, 295, 347,
381
Dorn, Julian, 375, 376

E

Ehmann, Matthias, 161
Engbring, Dieter, 171

F

Fandrich, Anatolij, 295
Freudenberg, Rita, 237

Fricke, Martin, 237
Frischmeier, Daniel, 285, 385

G

Gall, Rainer, 161
Gallenbacher, Jens, 221, 379
Gärtig-Daug, Anja, 377
Geldreich, Katharina, 181
Gerdes, Jana, 381
Glücks, Sebastian, 69
Göbel, Lisa, 191
Gräbl, Christoph, 361
Greubel, André, 161
Gumpert, Andrea, 201
Günzel-Weinkamm, Nicole, 161

H

Hacke, Alexander, 79
Haller, Verena, 161
Haselmeier, Kathrin, 89, 99, 237
Hausner, Niko, 109
Hellmig, Lutz, 191
Hembrock, Laura, 378
Hennecke, Martin, 161
Hermida, Martin, 231
Herper, Henry, 237
Heuer, Ute, 161
Hielscher, Michael, 211
Hildebrandt, Claudia, 363
Hinz, Volkmar, 237
Hochthurn, Tamara, 221
Honegger, Beat Döbeli, 231
Hubwieser, Peter, 181
Humbert, Ludger, 99, 119, 237

J

Jördens, Tobias, 379

K

Kaczmarek, Oliver, 21

Kapp, Florian, 247

Kastl, Petra, 374

Killich, Klaus, 99

Kirsch, Nadine, 49

Klinge, Niklas, 351

Knaus, Thomas, 25

Koch, Andreas, 257

Kommer, Sven, 25

Kopf, Christine, 25

Köster, Hilde, 367

Kronawitter, Julia, 161

L

Langer, Andrea, 380

Leschke, Rainer, 25

Lindner, Annabel, 161, 325

Losch, Daniel, 119

M

Magenheim, Johannes, 139

Markussen, Kris, 381

Marowsky, Maximilian, 382

Matzner, Matthias, 363

Meier, Urs, 265

Michaeli, Tilman, 129

Micheuz, Peter, 275, 365

Missomelius, Petra, 25

Mittag, Alexander, 257

Modrow, Eckart, 335

Müller, Dorothee, 99, 237

Müller, Kathrin, 139

O

Ohm, Paul, 382

Opel, Simone, 285, 385

Ossowski, Elisaweta, 383

P

Pancratz, Nils, 295, 347, 381

Petrenko, Ilona, 149

Podworny, Susanne, 285, 385

Pöhner, Nicolai, 161

Przybylla, Mareen, 79

R

Reese, Kerstin, 49

Romeike, Ralf, 129, 325, 374

Rösch, Eike, 384

S

Schifferle, Tobias M., 384

Schlichtig, Michael, 285, 385

Schmalfeldt, Thomas, 305

Schmid, Regina, 231

Schmid, Ute, 377

Schmidt, Ann-Katrin, 315

Schmidt, Pascal, 49, 386

Schroeder, Ulrik, 380

Schulte, Carsten, 139, 247, 285, 315,
385

Schwill, Andreas, 79, 237

Seegerer, Stefan, 325

Straube, Philipp, 367

Strecker, Kerstin, 335

Strobel, Stefan, 386

Suter, Peter, 384

T

Talbot, Mike, 181

Thomas, Marco, 149, 237, 351

Tilemann, Friederike, 25

Tschorn, Patrick, 378

W

Wachter, Lukas, 49
Wagenknecht, Christian, 211
Waldvogel, Bettina, 355
Wassong, Thomas, 285, 385
Wegner, Juliane, 369
Weich, Andreas, 25

Wendlandt, Katharina, 109
Wendlandt, Matthias, 109
Werner, Alexander, 377
Wolf, Verena, 49

Z

Zaugg, Pascal, 201