

Text Indexing and Information Retrieval

Übungsblatt 1

Besprechung: 15.10.2018

Aufgabe 1 (Theorie)

Erstellen Sie einen Trie für die Wörter {aardvark, affenpinscher, ant, baboon, bat, bonobo, cat, goat, goose, gorilla, grasshopper, zebra}. Zeigen Sie zumindest für die Wurzel die verschiedenen Varianten, wie die ausgehenden Kanten repräsentiert werden können.

Aufgabe 2 (Praxis)

Implementieren Sie einen Trie, sodass damit Strings lexikographisch sortiert werden können. Verwenden Sie dabei eine einfache Repräsentation der Trie-Knoten aus der Vorlesung.

Testen Sie die Laufzeit mit dem auf der Webseite zur Verfügung gestellten Strings und vergleichen Sie sie mit `gnu sort`.

Aufgabe 3 (Praxis)

Wie sieht für die Strings aus Aufgabe 2 der gewichtsbalancierte Binärbaum an der Wurzel aus? (Repräsentation (5) aus dem Skript.)

— Bitte wenden —

Aufgabe 4 (Theorie)

- a) Analysieren Sie die Laufzeit für das Sortieren von k Strings der Gesamtlänge N mit Hilfe von Tries. Die Alphabetgröße sei wieder σ . Nehmen Sie an, dass die ausgehenden Kanten an internen Knoten mit balancierten binären Suchbäumen (z.B. Rot-Schwarz-Bäume) implementiert sind. Die Alphabetgröße ist σ .
- b) Recherchieren Sie nach *theoretisch schnelleren* Sortierverfahren für Strings und beschreiben Sie ein solches Verfahren, auch an einem Beispiel (z.B. anhand der Strings aus Aufgabe 1). *Hinweis: das bestmögliche Verfahren hat die Laufzeit $O(N + \sigma)$.*