

Übung zu Algorithmen auf Sequenzen Blatt 1

Ausgabe: Donnerstag, 09. April **Besprechung:** Dienstag, 14. April

Aufgabe 1.1

Wie viele Teilstrings und wie viele Teilsequenzen hat eine Sequenz der Länge n stets? Wie viele unterschiedliche Teilstrings / Teilsequenzen eine Sequenz der Länge n hat, hängt allerdings von der Sequenz selbst ab. Geben Sie hierfür untere und obere Schranken an, sowie die Extrembeispiele.

Aufgabe 1.2

Es gibt 2^4 binäre Strings der Länge $n = 4$. Geben Sie einen möglichst kurzen String an, in dem alle 16 verschiedenen Strings als Substrings vorkommen. Wie ist die Minimallänge eines solchen Strings? Geben Sie zur Berechnung der Länge eine Formel an.

Aufgabe 1.3

Überlegen Sie sich, wie man das \tilde{c} -Array aus der in der Vorlesung genannten *rank*-Datenstruktur korrekt befüllen kann.

Aufgabe 1.4

Berechnen Sie für die *rank*-Datenstruktur mit zwei Ebenen für die kumulativen Summen den optimalen Wert für den Samplingfaktor k für die Szenarien $n = \{10^3, 10^6, 10^9, 10^{10}\}$ Einträge und Wortlänge $W = \{32, 64, 128, 256\}$ Bits. Wie viel Prozent an Speicherplatz spart man dadurch im Gegensatz zur Variante mit einer Ebene ein.

Aufgabe 1.5

Schreiben Sie die folgenden Befehle als Einzeilerbefehle nur mit Hilfe von arithmetischen, logischen oder Bitoperationen. Die Befehle sollen **ohne** Branch-Anweisungen auskommen. Sie dürfen also explizit keine `if`, `then`, `else` Kontrollstrukturen und keinen `?`-Operator benutzen. Gehen Sie davon aus, dass Zahlen als 32 Bit signed Integer gespeichert werden.

- `if(x == true) y = 1; else y = -1;`
- `if(x == 'a') y = ~0; else y = 0;`
- `x = (x < 0) ? -x : x;`