
Algorithmen auf Sequenzen

Abgabetermin: Donnerstag, den 31. Januar vor der Vorlesung

Dieses Übungsblatt enthält eine zusätzliche Notenbonusaufgabe, wobei die erzielten Punkte als Bonus-Punkte gewertet werden, d.h. dass die erzielten Punkte für den Notenbonus berücksichtigt werden, die zu erzielenden Punkte jedoch nicht.

Aufgabe (Notenbonus) 1

Betrachte das Wort $t = t_1 \cdots t_{17} = \text{ABANANAISANANANAS}$ und die zugehörige Burrows-Wheeler-Transformierte $\hat{t} = \hat{t}_1 \cdots \hat{t}_{18} = \text{\$NNBSNNNAAAAAAAAAI}$.

- Konstruiere die für die Rank-Select-Datenstruktur benötigten Arrays R , R' und R'' jeweils für B_A und B_N (für Anfragen vom Typ $\text{rank}_0(\cdot)$). Hierbei soll als Block-Länge $s' = 3$ und als Super-Block-Länge $s = 6$ verwendet werden (auch wenn dann $(s')^2 \neq s$).
- Beantworte die Rank-Anfragen $\text{rank}_0^{B_A}(17)$ und $\text{rank}_1^{B_N}(17)$ nach der in der Vorlesung vorgestellten Methode basierend auf den Arrays aus a).
- Konstruiere den Wavelet-Tree zu \hat{t} für $t\$$.
- Bestimme die Werte von $\text{Occ}(A, 14)$ und $\text{Occ}(N, 14)$ nach der in der Vorlesung vorgestellten Methode aus dem Wavelet-Tree aus c).

Hinweis: Das zu betrachtende Alphabet ist $\Sigma \cup \{\$\} = \{\$, A, B, I, N, S\}$, wobei die Ordnung auf dem Alphabet durch die Reihenfolge gegeben ist.

Aufgabe (Notenbonus) 2

Wende den Algorithmus zur 2-Approximation zur Bestimmung der minimalen Reversal-Distanz auf $(6, 7, 3, 4, 5, 8, 9, 1, 2)$ an. Gib dabei alle Zwischenschritte an und erkläre, warum eine bestimmte Reversion angewendet wird.

Aufgabe 3

Finde eine Permutation auf 4 Elementen, bei deren optimaler Sortierung durch (ungerichtete) Reversionen ein Strip aufgebrochen werden muss.

*Für die Teilnahme an der
Klausur am 13. Februar
ist auch eine Anmeldung
über TUMonline erforderlich.*