

1. Aufgabenblatt vom Donnerstag, den 24. Oktober 2002 zur Vorlesung

Algorithmen und Datenstrukturen (für Bioinformatik) (Frank Hoffmann)

Abgabe am Donnerstag, den 31. Oktober 2002 bis 16⁰⁰

1. **O-Notation** (3 Punkte).

Ordnen Sie die folgenden Funktionen derart, dass $g_i = \Omega(g_{i+1})$ gilt; begründen Sie jeweils kurz, warum dies so ist ($\log n = \log_2 n$):

$$(4/3)^n, (\sqrt{3})^{\log n^3}, \log^* n, n^2, n^3, \log^2 n, 2^{2^n}, n2^n, \ln n, 2^n, 4^{\log n}, (n+1)!, \sqrt{\log n}, n!, n.$$

2. **Amortisierte Analyse** (3 Punkte).

Auf einer Datenstruktur wird eine Folge von n Operationen ausgeführt. Die Kosten der i -ten Operation sind i , falls i eine exakte Zweierpotenz ist und 1 sonst. Bestimmen Sie die amortisierten Kosten pro Operation.

3. **AVL-Suchbäume** (4 Punkte)

Begründen und beschreiben Sie (in Pseudocode) eine Methode *isEqual(k)*, die für einen AVL-Baum zählt, wie oft ein gegebener Schlüssel k im Baum gespeichert ist. Ihr Verfahren sollte in Zeit $O(\log n + s)$ laufen, wobei s die Gesamtanzahl der Vorkommen des Schlüssels k im Baum der Größe n ist. Man nennt eine solche Laufzeit output-sensitiv, da sie direkt von der Anzahl s abhängt. Diskutieren Sie auch kurz den Fall beliebiger binärer Suchbäume.

4. **Zirkuläre Strings** (3 Punkte)

Beschreiben und analysieren Sie einen Linearzeitalgorithmus, der für zwei Strings α, β gleicher Länge überprüft, ob einer die zyklische Verschiebung des anderen ist.

5. **Anzahl der Matches** (3 Punkte)

Geben Sie ein generisches Beispiel eines Textes der Länge m und einer Menge von Mustern mit Gesamtlänge n , so dass die Anzahl der Matches Größenordnungsmäßig über $n + m$ liegt.

6. **Implementierung Z-Boxen, Abgabe: 7.11.** (7 Punkte).¹

Implementieren Sie den in der Vorlesung vorgestellten Algorithmus zur Bestimmung der Z-Boxen und der r_k und l_k -Werte eines Strings. Testen Sie Ihre Implementierung an Hand einiger Beispiele. Bis zu 5 Zusatzpunkte sind möglich für eine gelungene Visualisierung der Ausgabe.

**Praktische
Aufgabe!**

¹Implementierungen im Rahmen der praktischen Übungsaufgaben müssen in der Programmiersprache Java erstellt werden. Die Abgabe dieser Aufgaben erfolgt per email an den jeweiligen Tutor. Dabei bitte nur lauffähigen und compilierbaren Quellcode schicken. Besteht die Implementierung aus mehreren Dateien, sind diese — unter Beibehaltung der Verzeichnisstruktur — in ein Archiv (tar,tgz,zip) zu verpacken.