

Algorithmen und Datenstrukturen (PI.ADS.AD.VO)	schriftliche Einzelprüfung	28.06.2011	1
--	----------------------------	------------	---

Aufgabe 1 [20]

- [10] Geben Sie ein Programm in C++ artigem Pseudocode an, dessen Aufwand von der Ordnung $\Omega(\log n)$ und $O(n^2)$, aber nicht von der Ordnung $\Theta(\log n)$ oder $\Theta(n^2)$ ist.
- [10] Geben Sie in C++ artigem Pseudocode eine rekursive Funktion an, deren Laufzeitordnung $\Theta(n^3 \log n)$ ist. Zeigen Sie mit Hilfe des Mastertheorems, dass Ihrer Funktion wirklich die geforderte Laufzeitordnung hat.

Aufgabe 2 [20]

Addieren Sie zu Ihrer Matrikelnummer die Zahl 345726198. Die Ziffern der Summe seien in der Reihenfolge von links nach rechts in einem Array gespeichert. Sortieren Sie die Ziffern aufsteigend mit

- [8] Counting Sort.
- [8] Heapsort.
- [4] Selection Sort.

Geben Sie jeweils alle benötigten Zwischenschritte so genau an, dass der Ablauf des Algorithmus klar ersichtlich wird.

Aufgabe 3 [15]

- [4] In einen zu Beginn leeren binären Suchbaum werden die Buchstaben des Wortes HALBZEIT (in dieser Reihenfolge von links nach rechts) nacheinander eingefügt. Skizzieren Sie den Zustand des resultierenden binären Suchbaums.
- [11] Geben Sie in C++ artigem Pseudocode eine Definition für die Datenstruktur eines binären Suchbaums an. Schreiben Sie weiters ein Funktion, die den binären Suchbaum postorder traversiert und dabei jeweils den Inhalt der Knoten auf den Bildschirm ausgibt.

Aufgabe 4 [25]

- [10] Fügen Sie in eine zu Beginn leere Hashtabelle der Größe 7 die Zahlen 9, 11, 4, 1, 8 und 6 nacheinander in dieser Reihenfolge ein. Verwenden Sie als Hashfunktion die Funktion $h(k) = k \bmod 7$. Zur Behandlung von Kollisionen ist double hashing mit der Kollisionsfunktion (zweite Hashfunktion) $g(k) = (k \bmod 10) * 3$ (letzte Stelle von k mal 3) zu verwenden. Skizzieren Sie die resultierende Hashtabelle.
- [5] In der resultierenden Hashtabelle soll der Wert 35 gesucht werden. Geben Sie die Positionen der Tabelle, die überprüft werden müssen, in der Reihenfolge an, in der sie durchsucht werden.
- [5] Was würde passieren, wenn jetzt noch versucht würde, den Wert 17 einzufügen?
- [5] Was ist die grundlegende Idee aller Hashverfahren? Wie unterscheiden sich statische von dynamischen Hashverfahren? Welche dynamischen Hashverfahren kennen Sie (Aufzählung der Namen reicht aus)?

Aufgabe 5 [20]

Gegeben ist die folgende Adjazenzmatrix, die die Kosten der Verbindungen zwischen den Knoten eines gerichteten Graphen beschreibt:

$$\begin{pmatrix} 0 & 0 & 0 & 4 & z1 \\ 0 & 0 & 0 & 0 & z2 \\ 3 & 5 & 0 & z3 & 0 \\ 0 & 0 & 0 & 0 & z4 \\ 0 & z5 & 0 & 6 & 0 \end{pmatrix}$$

- [2] Ersetzen Sie in der Adjazenzmatrix die Gewichte z1 bis z7 durch Werte, die Sie aus Ihrer Matrikelnummer wie folgt ermitteln: z_i ergibt sich aus der i-ten Stelle der Matrikelnummer (von rechts beginnend nummeriert) plus 1. Für die Matrikelnummer 1234567 wäre z2 beispielsweise 7 (=6+1). Skizzieren Sie den Graphen, der durch diese Adjazenzmatrix beschrieben wird.
- [10] Bestimmen Sie mit dem Algorithmus von Dijkstra die kürzesten Wege vom Knoten 1 zu allen anderen Knoten des Graphen (Dabei entspricht Knoten 1 dem Knoten der ersten Zeile/Spalte in der Adjazenzmatrix).
- [8] Bestimmen Sie mit dem Algorithmus von Kruskal einen minimal spannenden Baum des Schattens des Graphen. (Sie erhalten den Schatten des Graphen, indem Sie die Richtungen der Kanten vernachlässigen. Werden dann zwei Knoten durch zwei Kanten verbunden, so werden diese Kanten zu einer zusammengefasst. Anders ausgedrückt: Zwei Knoten x und y im Schatten sind genau dann durch eine ungerichtete Kante verbunden, wenn im ursprünglich gerichteten Graphen zumindest eine der Kanten von x nach y oder von y nach x existiert. Als Gewicht der ungerichteten Kante wählen sie jeweils das Minimum aller durch sie repräsentierten gerichteten Kanten.) Notieren Sie alle Zwischenschritte so genau, dass klar ist, wann welche Kante zum spannenden Baum hinzugefügt wird.