Technische Universität Braunschweig
IBR - Abteilung Algorithmik
Prof. Dr. Sándor Fekete
Christiane Schmidt

Sommersemester 2011

Klausur
Algorithmen und Datenstrukturen
01.09.2011

Name: ....

Vorname: ....

Matr.-Nr.: ....

Studiengang: ....

Bachelor ☐ Master ☐ Diplom ☐ Andere ☐

Mit der Veröffentlichung des Klausurergebnisses nur mit der Matrikelnummer über die Mailingliste und auf der Homepage bin ich einverstanden.

....
Unterschrift

Hinweise:

· Bitte das Deckblatt ausfüllen. Die Heftung der Blätter darf nicht entfernt werden. Eigenes Papier ist nicht erlaubt. Die Rückseiten dieser Blätter dürfen beschrieben werden.

· Die Klausur besteht aus 15 Blättern.

· Hilfsmittel: keine.

· Die Klausur ist mit 50 von 100 Punkten bestanden.

· Alle Graphen in dieser Klausur sind einfache Graphen, d. h. sie haben keine Multikanten und keine parallelen Kanten; das gilt auch für die von Dir zu konstruierenden Graphen.

· Mit *Bleistift* oder *in rot* geschriebene Klausurteile können nicht gewertet werden.

· Die Bearbeitungszeit für die Klausur ist 120 Minuten.

· **Bearbeitete Aufgaben bitte unten ankreuzen.**

Punktzahlen für die Korrektur freilassen!

Aufgabe	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	Σ
Bearbeitet ($\times$)									
Punkte	**18**	**11**	**12**	**17**	**11**	**11**	**10**	**10**	**100**
Erzielte Punkte									

1.Aufgabe: **Graphen** — 8+4+6 Punkte

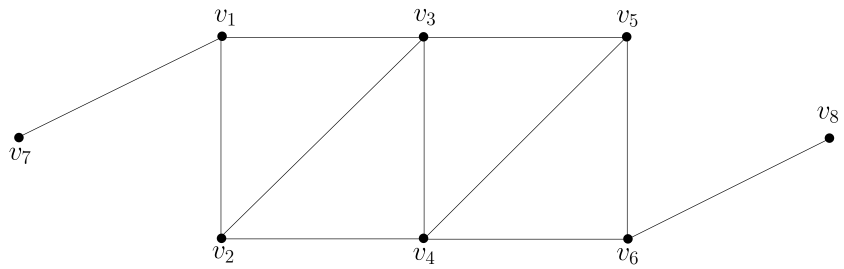

Abbildung 1: Der Graph H.

a) Wende Tiefensuche auf den Graphen H aus Abbildung **??** an; starte dabei mit dem Knoten v_1. Falls zu einem Zeitpunkt mehrere Knoten für den nächsten Schritt in Frage kommen, wähle denjenigen mit dem kleinsten Index. Gib die Menge Q jedesmal an, wenn sie sich ändert und zeichne den gefundenen Baum T.

b) Zeichnen einen Graphen mit 6 Kanten, der einen Hamiltonkreis und einen Eulerweg, aber keine Eulertour hat. Kennzeichne den Hamiltonkreis und den Eulerweg.

c) Zeige: Jeder Graph $G = (V, E)$ enthält einen Pfad der Länge $\delta(G) = \min\{d(v)|v \in V\}$ (wobei $d(v)$ der Grad des Knotens v ist).

2.Aufgabe: **Binäre Suchbäume (kein AVL-Baum!)** — 7+4 Punkte

a) Füge nacheinander die folgenden Elemente in einen zu Beginn leeren binären Suchbaum ein. Gib den Baum nach jeder Einfügeoperation an:

$$13, 8, 20, 24, 16, 22, 26$$

b) Lösche die 20 aus dem konstruierten Baum. Beschreibe kurz, wie du dabei vorgehst und gib den Baum nach dem Löschen an.

3.Aufgabe: Komplexität **3+3+3+3 Punkte**

Seien $f, g, h : \mathbb{N} \mapsto \mathbb{R}$ zwei Funktionen. Sei $f \in \Omega(g), h \in \Omega(f)$.

a) Zeige oder widerlege: $f \in O(g)$

b) Zeige oder widerlege: $g \in O(h)$

c) Zeige oder widerlege: $f \in \Theta(h)$

d) Zeige: $3n^6 - n^4 + 12n^2 - 200 \in O(n^6)$. Gib dazu explizit geeignete Konstanten c und n_0 aus der Definition an und zeige, dass sie die Definition erfüllen.

4.Aufgabe: Rekursionen **4+4+5+4 Punkte**

a) Wie lautet das Mastertheorem aus der Vorlesung?

b) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion
$U(n) = U(\frac{n}{2}) + 8 \cdot U(\frac{n}{3}) + 3n^2$.
Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

c) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion
$V(n) = 64 \cdot V(\frac{n}{4}) + 17n^2$.
Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

d) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion
$T(n) = 10 \cdot T(\frac{n}{5}) + 2 \cdot T(\frac{n}{2}) + 10 \cdot T(\frac{n}{10}) + 25n^2$.
Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

5.Aufgabe: Hashing **11 Punkte**

Wir betrachten ein leeres Array A der Größe 10, d.h. es gibt die Speicherzellen $A[0], A[1], \ldots, A[9]$; in diesem führen wir offenes Hashing mit der folgenden Hashfunktion durch:

$$t(i,x) = (2x + i^2 + 2i) \bmod 10$$

Dabei ist x ein einzusetzender Schlüssel und i die Nummer des Versuches, x in eine unbesetzte Speicherzelle des Arrays zu schreiben (beginnend bei $i = 0$).
Berechne zu jedem der folgenden Schlüssel die Position, die er in A bekommt:

$$6, 11, 16, 27, 7$$

(Hinweis: Die Schlüssel sollen in der gegebenen Reihenfolge eingefügt werden und der Rechenweg sollte klar erkennbar sein.)

Trage die Elemente in folgendes Array ein:

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

6.Aufgabe: Sortieren **7+4 Punkte**

a) Wende die Funktion MERGE(A,1,4,7) (aus Mergesort) auf folgendes Array an:

$$A[1] = 8 \quad A[2] = 3 \quad A[3] = 7 \quad A[4] = 4 \quad A[5] = 2 \quad A[6] = 11 \quad A[7] = 5$$

Trage das Ergebnis in folgendes Feld ein:

A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]

b) Wende die Funktion PARTITION(A,1,5) (aus Quicksort) auf folgendes Array an:

$$A[1] = 4 \qquad A[2] = 17 \qquad A[3] = 8 \qquad A[4] = 3 \qquad A[5] = 12$$

Das Referenzelement soll dabei wie in der Vorlesung gewählt werden (also A[5]). Gib das Array nach **jeder** Tausch-Operation an.

7.Aufgabe: Datenstrukturen — 4+6 Punkte

a) Gib den Pseudocode für die Operatioinen PUSH(S, x) und POP(S) für einen Stack S, sowie die Operationen ENQUEUE(Q, x) und DEQUEUE(Q) für eine Warteschlange Q an.

b) Zeige wie ein Stack S unter der Verwendung zweier Warteschlangen Q_1 und Q_2 implementiert werden kann. Gib dafür die neuen Operationen PUSH'(S, x) und POP'(S) an.

8.Aufgabe: Kurzfragen — 2+2+2+2+2 Punkte

a)	Das Löschen eines Elements hat in einer Warteschlange im Worst-Case eine höhere Laufzeit als in einem AVL-Baum.	☐ wahr ☐ falsch
b)	Fleurys Algorithmus liefert kürzeste Wege in Graphen.	☐ wahr ☐ falsch
c)	Die Breitensuche verwendet eine Warteschlange.	☐ wahr ☐ falsch
d)	Für einen vollständigen Graphen mit n Knoten benötigt die Inzidenzmatrix mehr Speicherplatz als die Adjazenzmatrix	☐ wahr ☐ falsch
e)	Lineares und Quadratisches Sondieren verwenden offene Adressierung.	☐ wahr ☐ falsch

Viel Erfolg!!!