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

Wintersemester 2011/12

Klausur
Algorithmen und Datenstrukturen
21.02.2012

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**	**10**	**19**	**17**	**11**	**8**	**7**	**10**	**100**
Erzielte Punkte									

1.Aufgabe: Graphen **8+4+6 Punkte**

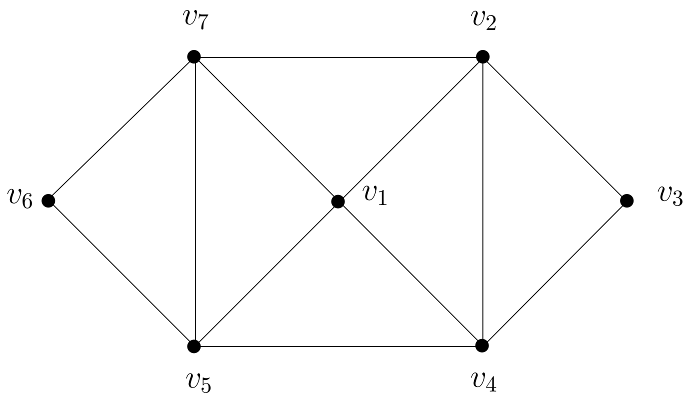

Abbildung 1: Der Graph H.

a) Wende Breitensuche auf den Graphen H aus Abbildung 1 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 (zusammenhängenden) Graphen mit 7 Knoten, der eine Eulertour und einen Hamiltonpfad, aber keinen Hamiltonkreis hat. Kennzeichne Eulertour und Hamiltonpfad.

c) Beweise, dass es in jedem zusammenhängenden ungerichteten Graphen $G = (V, E)$ einen Knoten $v \in V$ gibt, nach dessen Entfernung der Graph zusammenhängend bleibt. Hinweis: Betrachte den DFS-Baum für G.

2.Aufgabe: AVL-Bäume **10 Punkte**

Gegeben sei der AVL-Baum T aus Abbildung 2. Füge nacheinander die Elemente 3, 29 und 36 ein, so dass T ein AVL-Baum bleibt; gib den Baum nach jeder Einfüge-Operation an.
(Hinweis: Nach *jedem* Einfügen soll der geänderte Baum ein AVL-Baum sein. Zum Schluss sollen alle drei Zahlen eingefügt sein.)

3.Aufgabe: Komplexität **3+3+3+5+5 Punkte**

Seien $f, g : \mathbb{N} \mapsto \mathbb{R}$ zwei Funktionen.

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

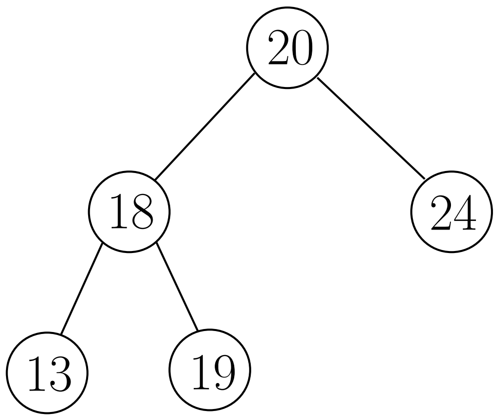

Abbildung 2: Der AVL-Baum T.

b) Zeige oder widerlege: $f \in \Omega(g) \Rightarrow g \in O(f)$

c) Zeige: $35n^4 + 18n^3 - 512 \in O(n^4)$. Gib dazu explizit geeignete Konstanten c und n_0 aus der Definition an und zeige, dass sie die Definition erfüllen.

d) Sortiere die folgenden Funktionsklassen nach Inklusion (Du musst die Antwort nicht begründen). Kennzeichne identische Klassen und beweise deren Identität.

$$O(3^n) \;\; O(\ln n) \;\; O(3) \;\; O(n^n) \;\; O(n^{\ln 3}) \;\; O(n^3) \;\; O(n!) \;\; O(3^{\ln n}) \;\; O(3n)$$

e) Maximiere die Leistungsfähigkeit eines Programms. Dafür hast Du für einen Input der Größe n eine der folgenden Möglichkeiten zur Auswahl:

(i) Anschaffung einer besseren Rechenanlage mit der sechsfachen Rechenleistung.

(ii) Verbesserung der Laufzeit des zugrundeliegenden Algorithmus von $2n^3$ auf $2n^2$.

(iii) Reduktion der Datenmenge durch einen vorgeschalteten linearen Algorithmus (Zeitbedarf $3n$) auf $\frac{n}{2}$.

Begründe Deine Wahl und zeige, für welchen Wertebereich von n Deine Wahl die Beste ist (betrachte den jeweiligen Zeitbedarf). Für welche Wertebereiche von n wären die anderen Optimierungsmöglichkeiten besser?

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

$U(n) = U(\frac{n}{3}) + 2 \cdot U(\frac{n}{8}) + U(\frac{n}{12}) + 8n$.

Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

c) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion
$V(n) = 16 \cdot V(\frac{n}{2}) + 4n^3$.
Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

d) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion
$T(n) = 3 \cdot T(\frac{n}{2}) + 8n^3 + 40 \cdot T(\frac{n}{4})$.
Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

5.Aufgabe: Hashing — 11 Punkte

Wir betrachten ein leeres Array A der Größe 8, d.h. es gibt die Speicherzellen $A[0], A[1], \ldots, A[7]$; in diesem führen wir offenes Hashing mit der folgenden Hashfunktion durch:

$$t(i, x) = (2x + 3i^2) \bmod 8$$

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:

$$3, 7, 11, 5, 12$$

(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	

6.Aufgabe: Sortieren **5+3 Punkte**

a) Eine lineare Liste $A[1, \dots, n]$ soll sortiert werden. Das Sortierverfahren Bubblesort ist wie folgt definiert:

Algorithm 1: Bubblesort(A, n)

```
begin
    for j := n − 1 DOWNTO 1 do
        for i := 1 TO j do
            if A[i] < A[i + 1] then
                vertausche (A[i], A[i + 1])
```

Zeige die Korrektheit des Verfahrens und analysiere sein Zeitverhalten.

b) Sortiere die folgenden Zahlen mit dem in der Vorlesung vorgestellten Mergesort. Kennzeichne in jedem Schritt, welche Teilfolgen gemischt werden.

$[14] \quad [3] \quad [11] \quad [9] \quad [7] \quad [2] \quad [5]$

7.Aufgabe: Algorithmenentwurf **7 Punkte**

Gegeben ist ein Array $A[1, \dots, n]$, das nur ganzzählige Einträge enthält. Entwirf einen möglichst schnellen Algorithmus, der entscheidet, ob zwei Indizes existieren, so dass $A[i] = A[j]$ gilt. Begründe die Laufzeit Deines Algorithmus.

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

a) Jeder zusammenhängende Graph mit n Knoten hat mindestens $n - 1$ Kanten. ☐ wahr ☐ falsch

b) Jeder Graph mit n Knoten hat mindestens $n - 1$ Kanten. ☐ wahr ☐ falsch

c) Gegeben sei ein sortierter Array mit Einträgen $S[i]$. LINKS bezeichne die linke Randposition, RECHTS die rechte Randposition (LINKS < RECHTS). Dann terminiert binäre Suche in $O(\text{RECHTS-LINKS})$ Schritten. ☐ wahr ☐ falsch

d) Für einen Graph mit Einheitskantenlängen liefert die Tiefensuche kürzeste Wege von einem Knoten zu allen anderen. ☐ wahr ☐ falsch

e) Die Größe einer Adjazenzliste ist $O(mn)$. ☐ wahr ☐ falsch