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

Sommersemester 2012

Klausur
Algorithmen und Datenstrukturen
30.08.2012

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	**19**	**12**	**13**	**19**	**7**	**12**	**8**	**10**	**100**
Erzielte Punkte									

1.Aufgabe: Graphen — 9+4+6 Punkte

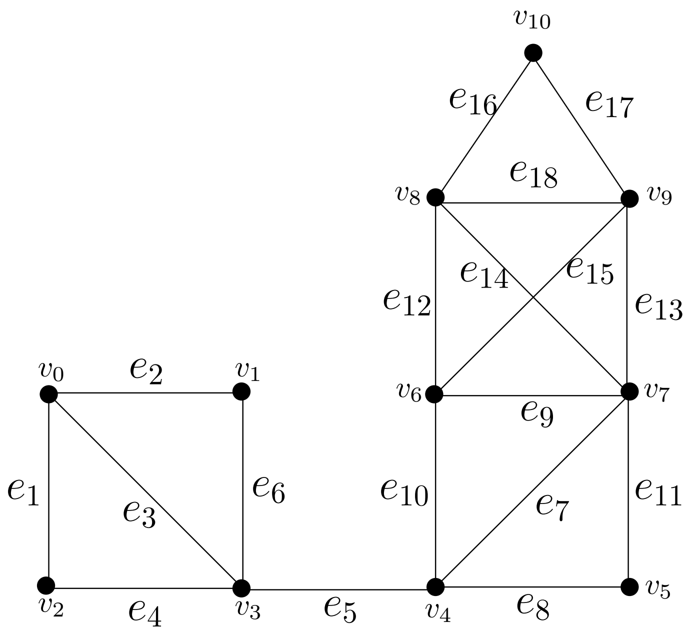

Abbildung 1: Der Graph H.

a) Wende Fleurys Algorithmus aus der Vorlesung auf den Graphen H aus Abbildung 1 an. Starte dabei mit dem Knoten v_0. Können während der Ausführung des Algorithmus in einem Schritt mehrere Kanten ausgewählt werden, wähle die mit dem kleinsten Index.

   Gib die Kanten in der Reihenfolge in der sie besucht werden an. Zeichne die gefundene Lösung.

b) Zeichnen einen Graphen mit 6 Knoten und 7 Kanten, der keinen Hamiltonpfad und keine Eulertour, aber einen Eulerweg hat. Kennzeichne den Eulerweg.

c) Ein Graph $G = (V, E)$ heißt *bipartit*, wenn sich die Knotenmenge in zwei Teilmengen zerlegen läßt, d.h. $V = A \cup B$, so dass jede Kante genau einen Knoten in A und einen Knoten in B hat, d.h.$\forall \{x, y\} \in E : x \in A, y \in B$.

   Zeige: Ein Kreis ist bipartit $\Leftrightarrow$ Die Kreislänge ist gerade.

2.Aufgabe: AVL-Bäume — 12 Punkte

Gegeben sei der AVL-Baum T aus Abbildung 2. Füge nacheinander die Elemente 16, 17 und 8 ein, so dass T ein AVL-Baum bleibt; gib den Baum nach jeder Einfüge-Operation an.

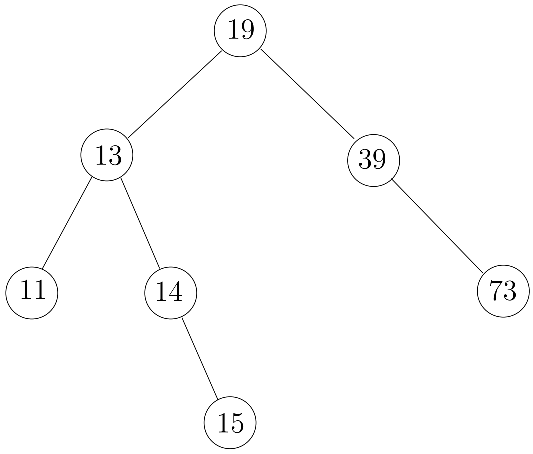

Abbildung 2: Der AVL-Baum T.

(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+4 Punkte

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

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

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

c) Zeige: $23n^3 - 12n^2 + 50n - 110 \in O(n^3)$. 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.

$$O(4n),\ O(3n\ln n),\ O(n^2),\ O(4),\ O(2^n),\ O(2((\sum_{i=1}^{n} i) - \frac{n}{2})),\ O(n^n),\ O(\ln n)$$

4.Aufgabe: Rekursionen 4+3+3+4+5 Punkte

a) Wie lautet das Mastertheorem aus der Vorlesung?

b) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion
$U(n) = 4 \cdot U(\frac{n}{3}) + 17 \cdot n^2 + 20 \cdot U(\frac{n}{6})$.
Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

c) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion $V(n) = 14 \cdot V(\frac{n}{36}) + 23n + 12 \cdot V(\frac{n}{24}) + V(\frac{n}{10})$.
Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

d) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion $T(n) = 49 \cdot T(\frac{n}{7}) + 42n$.
Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

e) Leite die Laufzeit von Mergesort mit Hilfe des Mastertheorems her. Stelle dazu zunächst eine Gleichung für die Laufzeit auf und begründe diese. Nutze anschließend das Mastertheorem, um einen geschlossenen Ausdruck für die Laufzeit abzuleiten. Gib dazu die Werte aller im Mastertheorem auftretenden Parameter an.

5.Aufgabe: Hashing **7 Punkte**

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

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

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:

$$3, 5, 7, 10, 11, 16$$

(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	

6.Aufgabe: Sortieren **11+2 Punkte**

a) Wende die Funktion MERGE(A,1,3,6) (aus Mergesort) auf folgendes Array an:

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

Gib alle Parameter und Aufrufe in der Funktion MERGE an. Trage das Ergebnis in folgendes Feld ein:

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

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

$$[8] \quad [11] \quad [3] \quad [2] \quad [5] \quad [9] \quad [4]$$

7.Aufgabe: Algorithmenentwurf **8 Punkte**

Gegeben sei ein Array A mit $n \geq 2$ positiven Zahlen. Alle Zahlen in dem Array seien unterschiedlich. Gib einen Algorithmus in Pseudocode an, der das **zweitgrößte** Element des Arrays in einer Laufzeit von $O(n)$ ausgibt. (Hinweis: Dein Algorithmus darf nicht mehr als 20 Zeilen haben.)

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

a) Suche eines Objekts in einer doppelt verketteten Liste ist in $O(1)$ möglich.
□ wahr
□ falsch

b) Quicksort hat im Worst-Case eine Laufzeit von $O(n^2)$.
□ wahr
□ falsch

c) Die Höhe eines binären Suchbaumes mit n Knoten ist immer $O(\log n)$.
□ wahr
□ falsch

d) Die Breitensuche verwendet eine Warteschlange.
□ wahr
□ falsch

e) Wenn in einem Graphen ein Weg zwischen zwei Knoten u und v existiert, dann existiert auch ein $u - v$–Pfad.
□ wahr
□ falsch