

Prof. Dr. Sándor P. Fekete
Stephan Friedrichs

Klausur
Algorithmen und Datenstrukturen
04. 09. 2014

Name:

Vorname:

Matr.-Nr.:

Studiengang:

Mit der Veröffentlichung meines Klausurergebnisses unter meiner Matrikelnummer bin ich einverstanden.

.....
Unterschrift

Bachelor Master Andere

Hinweise:

- Bitte das Deckblatt vollständig ausfüllen.
- Die Klausur besteht aus 14 Blättern, bitte auf Vollständigkeit überprüfen.
- Erlaubte Hilfsmittel: Keine.
- Eigenes Papier ist nicht erlaubt.
- Die Rückseiten der Blätter dürfen beschrieben werden.
- Die Klausur ist mit 50 % der Punkte bestanden.
- Antworten die *nicht* gewertet werden sollen bitte deutlich durchstreichen. Kein Tippex verwenden.
- Mit *Bleistift* oder in *Rot* geschriebene Klausurteile können nicht gewertet werden.
- Werden mehrere Antworten gegeben, werten wir die mit der geringsten Punktzahl.
- Sämtliche Algorithmen, Datenstrukturen, Sätze und Begriffe beziehen sich, sofern nicht explizit anders angegeben, auf die in der Vorlesung vorgestellte Variante.
- Die Bearbeitungszeit für die Klausur beträgt 120 Minuten.

Aufgabe	1	2	3	4	5	6	7	8	Σ
Punkte	16	10	16	14	12	13	9	10	100
Erreicht									
Note	—	—	—	—	—	—	—	—	

Aufgabe 1: Graphen

(10+3+3 Punkte)

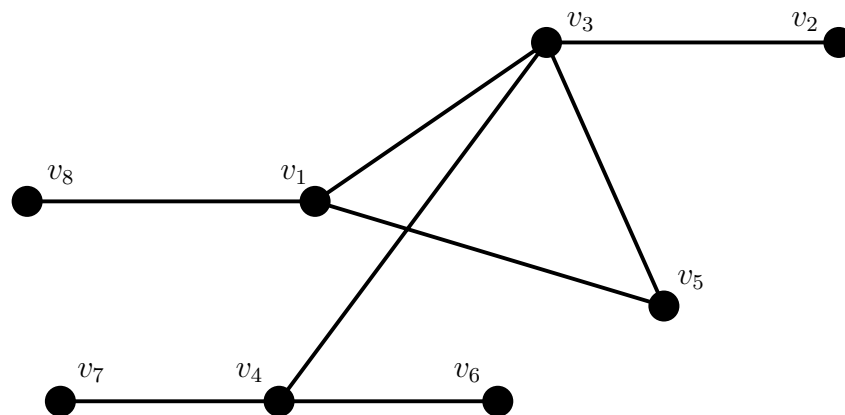


Abbildung 1: Der Graph G .

- a) Wende Tiefensuche auf den Graphen G 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 R des Algorithmus GRAPH-SCAN jedes Mal an, wenn sie sich ändert, und zeichne den gefundenen Baum T .

b) Zeichne einen Graphen mit mindestens 4 Knoten, der einen Hamiltonkreis, aber keinen Eulerweg besitzt.

c) Zeige oder widerlege: Jeder vollständige Graph mit $n \geq 3$ Knoten, wobei n ungerade ist, besitzt eine Eulertour.

Aufgabe 2: AVL-Bäume

(2+2+2+2+2 Punkte)

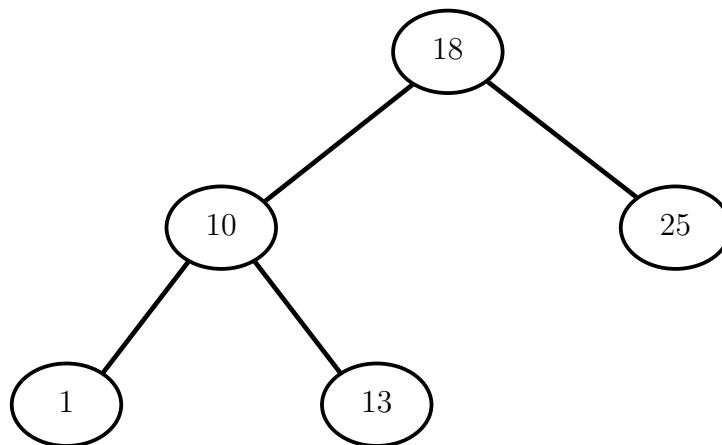


Abbildung 2: Der AVL-Baum T .

Gegeben ist der AVL-Baum T aus Abbildung 2. Führe nacheinander folgende Operationen aus (damit sind die Algorithmen aus der Vorlesung gemeint, die die AVL-Eigenschaft erhalten); zeichne dabei alle Schritte jeder Operation in einen separaten Baum:

- $\text{INSERT}(T, 12)$
- $\text{INSERT}(T, 5)$
- $\text{INSERT}(T, 3)$
- $\text{INSERT}(T, 2)$
- $\text{INSERT}(T, 14)$

Aufgabe 3: Komplexität**(4+4+4+4 Punkte)**Seien $f_1, f_2, f_3, g_1, g_2 : \mathbb{N} \rightarrow \mathbb{R}$ Funktionen.

a) Zeige oder widerlege:

$$f_1 \in \Theta(f_2) \text{ und } f_2 \in \Theta(f_3) \Leftrightarrow f_1 \in \Theta(f_3)$$

b) Seien $f, g : \mathbb{N} \rightarrow \mathbb{R}$ zwei Funktionen mit $f(x) \mapsto f_1(x) \cdot f_2(x)$ sowie $g(x) \mapsto g_1(x) \cdot g_2(x)$. Zeige oder widerlege:

$$f_1 \in O(g_1) \text{ und } f_2 \in O(g_2) \Rightarrow f \in O(g)$$

- c) Seien $f, g : \mathbb{N} \rightarrow \mathbb{R}$ zwei Funktionen mit $f(x) \mapsto f_1(x) \cdot f_2(x)$ sowie $g(x) \mapsto g_1(x) \cdot g_2(x)$. Zeige oder widerlege:

$$f \in O(g) \quad \Rightarrow \quad f_1 \in O(g_1) \text{ und } f_2 \in O(g_2)$$

- d) Gesucht ist ein Algorithmus, der die Elemente a_1, \dots, a_n bearbeiten soll. Dazu gibt es zwei Möglichkeiten:
- (i) Die Elemente werden der Reihe nach bearbeitet. Das dauert i Zeiteinheiten für das Element a_i .
 - (ii) Die Elemente werden erst mit QUICKSORT sortiert. Dadurch verändert sich die Bearbeitungszeit für jedes einzelne Element auf 34 Zeiteinheiten.

Welche asymptotische Laufzeit hat Variante (i)? Welche asymptotische Laufzeit ergibt sich für Variante (ii) im Worst- und welche im Average-Case? Welche Vorgehensweise ist im Worst-Case asymptotisch schneller?

Aufgabe 4: Rekursionen

(4+4+3+3 Punkte)

a) Wie lautet das Mastertheorem aus der Vorlesung?

b) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion

$$T(n) = 4 \cdot T\left(\frac{n}{2}\right) + 7n.$$

Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

- c) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion

$$T(n) = n + T\left(\frac{2n}{3}\right) + 14 - 3n + 5 \cdot T\left(\frac{n}{3}\right) + 18n^2.$$

Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

- d) Bestimme mit Hilfe des Mastertheorems das asymptotische Wachstum der Rekursion

$$T(n) = 7n^3 + T\left(\frac{5n}{6}\right) + T\left(\frac{n}{6}\right) + n^{2,5} - 14.$$

Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

Aufgabe 5: Hashing**(12 Punkte)**

Wir betrachten ein anfangs leeres Array A der Größe 7, es gibt also die Speicherzellen $A[0], A[1], \dots, A[6]$. In diesem führen wir offenes Hashing mit der folgenden Hashfunktion durch:

$$t(i, x) = (2x - i^2 - i) \pmod{7}$$

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:

4, 28, 0, 6, -3

Dabei sollen die Schlüssel in der gegebenen Reihenfolge eingefügt werden und der Rechenweg soll klar erkennbar sein. Trage die Elemente in das Array in Abbildung 3 ein.

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]

Abbildung 3: Die Hashtabelle A.

Aufgabe 6: Sortieren**(8+1+2+2 Punkte)**

0077					
2014					
0338					
1990					
4534					
1113					
2343					
7625					
1974					
1168					
3447					
1954					

Abbildung 4: Radixsort im Array A.

Wir betrachten RADIXSORT zum Sortieren von n Zahlen mit jeweils maximal $d \in \mathbb{N}$ dezimalen Ziffern. Sei \mathcal{S} der in jeder Iteration von RADIXSORT verwendete Sortieralgorithmus.

- Sortiere das Array A aus Abbildung 4 mit RADIXSORT. Gib A nach jeder Iteration an, nutze dazu Abbildung 4.
- Welche Eigenschaft muss \mathcal{S} aufweisen?

c) Wie ist die asymptotische Laufzeit von RADIXSORT für ganze Zahlen mit maximal d Ziffern, wenn die Laufzeit von \mathcal{S} in $\Theta(n)$ liegt?

d) Wie ist die asymptotische Laufzeit von RADIXSORT für ganze Zahlen mit $d \in \Theta(n)$ Ziffern, wenn die Laufzeit von \mathcal{S} in $\Theta(n)$ liegt?

Aufgabe 7: Algorithmenentwurf**(6+3 Punkte)**

Gegeben sei ein binärer Baum T , in dessen Knoten Zahlen abgespeichert sind.

- a) Gib einen rekursiven Algorithmus an, der in $O(n)$ alle Elemente von T addiert. Ist der Baum leer, soll 0 zurückgegeben werden. $\text{empty}(T)$ prüft, ob der Baum leer ist, $\text{left}(T)$ bzw. $\text{right}(T)$ geben den linken bzw. rechten Teilbaum von T zurück und $\text{key}(T)$ die in der Wurzel von T abgespeicherte Zahl. Die Addition zweier Zahlen benötigt konstante Zeit.

- b) Begründe, warum der Algorithmus aus a) in $O(n)$ liegt.

Aufgabe 8: Kurzfragen

(2+2+2+2+2 Punkte)

- a) Wenn zwei Algorithmen in $\Theta(f)$ liegen, haben sie für jede Eingabe dieselbe absolute Laufzeit in Sekunden. wahr
 falsch
- b) Einfügen in eine Hashtabelle mit offenem Hashverfahren hat die Worst-Case Laufzeit $O(1)$. wahr
 falsch
- c) MERGESORT ist ein stabiles Sortierverfahren. wahr
 falsch
- d) Doppelt verkettete Listen können in $O(1)$ konkateniert werden. wahr
 falsch
- e) Jeder Binärbaum mit maximal 3 Elementen ist höhenbalanciert. wahr
 falsch

Viel Erfolg 😊