Technische Universität Braunschweig
Institut für Betriebssysteme und Rechnerverbund
Abteilung Algorithmik

Sommersemester 2017

Prof. Dr. Sándor P. Fekete
Phillip Keldenich
Arne Schmidt

Klausur
Algorithmen und Datenstrukturen
07. 08. 2017

Name: ....

Vorname: ....

Matr.-Nr.: ....

Studiengang: ....

☐ Bachelor ☐ Master ☐ Andere

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

....
Unterschrift

Hinweise:

· Bitte das Deckblatt in Druckschrift vollständig ausfüllen.
· Die Klausur besteht aus 17 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.
· Sofern nicht anders angegeben, sind alle Graphen als einfache Graphen zu verstehen.
· Die Bearbeitungszeit für die Klausur beträgt 120 Minuten.

Aufgabe	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Σ**
Punkte	**14**	**10**	**15**	**14**	**16**	**11**	**10**	**10**	**100**
Erreicht									
Note	—	—	—	—	—	—	—	—	

Aufgabe 1: Graphen **(8+1+2+2+1 Punkte)**

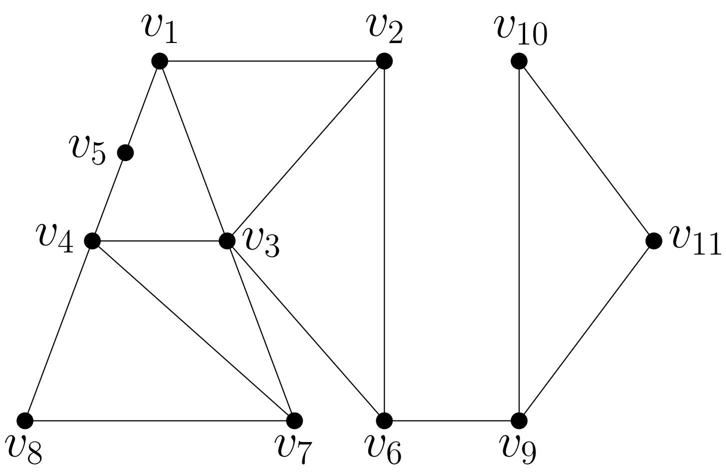


Abbildung 1: Der Graph G.

a) Wende Breitensuche auf den Graphen aus Abbildung 1 an; starte dabei bei 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 jedesmal an, wenn sie sich ändert, und zeichne den resultierenden Breitensuchbaum T.

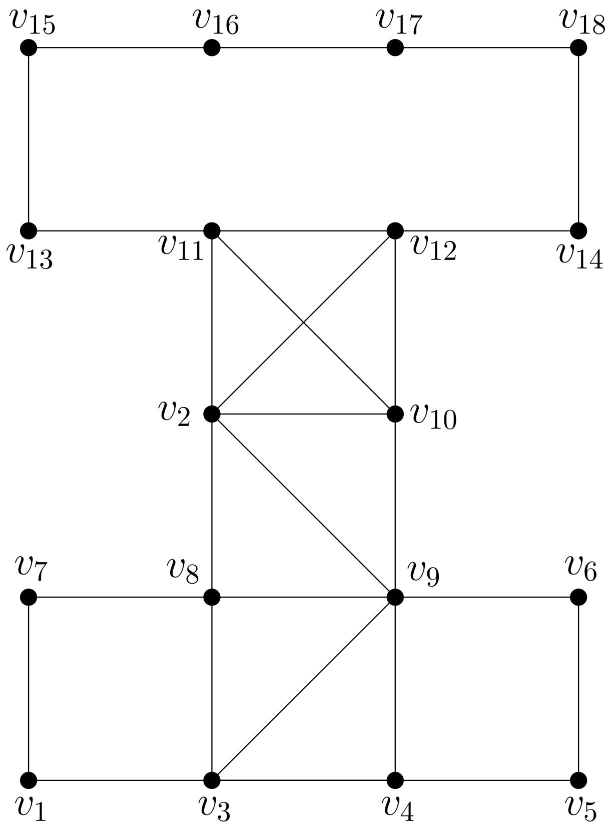


Abbildung 2: Der Graph H.

b) Betrachte den Graphen H aus Abbildung 2. Hat dieser Graph eine Eulertour? Begründe deine Antwort! Falls ja, gib für eine solche Tour die Reihenfolge (inklusive aller Wiederholungen) an, in der sie *die Knoten* von H besucht. Beginne dabei mit v_1.

c) Hat der Graph H einen Eulerweg, der bei Knoten v_1 beginnt? Begründe deine Antwort! Falls ja, gib für einen solchen Weg die Reihenfolge (inklusive aller Wiederholungen) an, in der er *die Knoten* von H besucht.

d) Hat der Graph H einen Eulerweg, der bei Knoten v_2 beginnt? Begründe deine Antwort! Falls ja, gib für einen solchen Weg die Reihenfolge (inklusive aller Wiederholungen) an, in der er *die Knoten* von H besucht.

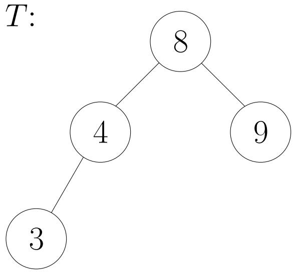
e) Zeige oder widerlege: Jeder einfache Graph, in dem jeder Knoten Grad 2 hat, besitzt eine Eulertour.

Aufgabe 2: AVL-Bäume **(2+2+2+2+1+1 Punkte)**

Betrachte in den Aufgabenteilen a) bis e) den Baum, der in der jeweiligen Abbildung dargestellt wird. Führe die Operation, die in dem jeweiligen Aufgabenteil genannt ist, und die damit verbundenen Restrukturierungsmaßnahmen (damit sind die Algorithmen aus der Vorlesung gemeint, die die AVL-Eigenschaft erhalten) auf dem entsprechenden Baum aus. Zeichne dabei das Resultat nach jeder einzelnen ausgeführten Operation Insert, Delete und Restructure in einen separaten Baum:

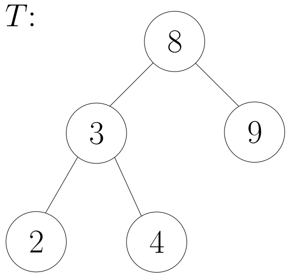
a) Insert$(T, 2)$

T:



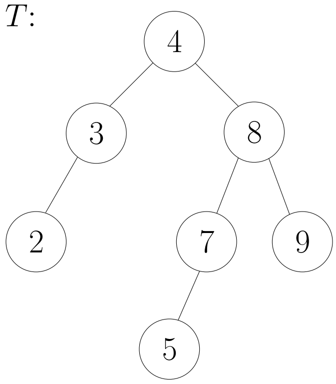
b) Insert$(T, 7)$

T:



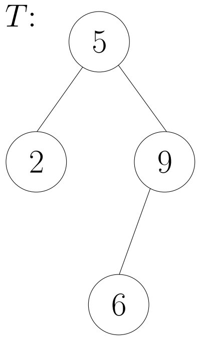
c) $\textsc{Delete}(T, 2)$

T:



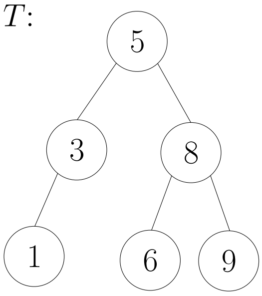
d) $\textsc{Insert}(T, 8)$

T:



e) INSERT$(T, 4)$

T:



f) Wieviele Knoten besitzt ein AVL-Baum der Höhe 5 mindestens und wieviele besitzt er maximal?

Aufgabe 3: O-Notation und Komplexität **(4+4+2+2+3 Punkte)**

Benutze in dieser Aufgabe nur die in der Vorlesung und großen Übung vorgestellten Begriffe und Definitionen. Seien $f_1, f_2, g_1, g_2 : \mathbb{N} \to \mathbb{R}$ Funktionen.

a) Gib für die folgenden Paare von Funktionsklassen an, in welcher Teilmengenbeziehung sie zueinander stehen. Schreibe dazu in die mittlere Spalte „=“, falls A genau dieselben Funktionen enthält wie B, „$\subset$“ wenn alle Funktionen aus A auch in B enthalten sind, „$\supset$“ wenn alle Funktionen aus B auch in A enthalten sind, und „$\times$“, wenn dies alles nicht zutrifft. Eine Begründung ist *nicht* erforderlich.

A	Antwort	B
$O(n)$		$O(n - \log n)$
$O(n)$		$\Theta(n)$
$\Omega(n^2)$		$\Omega(n \log n)$
$O(1)$		$O(2 + \cos(\pi n))$
$\Omega(5^n)$		$\Omega(4^n)$
$O(\log(n^2))$		$O((\log(n))^2)$
$O(1)$		$\Omega(1/n)$
$\Theta(\sqrt{n})$		$\Theta(\log n)$

b) Kreuze in der nachfolgenden Tabelle an, ob die Funktion $f(n)$ in der angebenenen Klasse liegt. Eine Begründung ist *nicht* erforderlich.

$f(n)$	$O(n)$	$\Omega(n)$	$\Theta(n \log n)$	$O(2^n)$
$n + \log n$				
$n / \log n$				
$n \cdot 2^n$				
$n^{2.1}$				

c) Sei $f : \mathbb{N} \to \mathbb{R}$ eine Funktion mit $f(n) > 0$ für alle $n \in \mathbb{N}$. Zeige oder widerlege: $f(n) \in \Theta(f(n) + 1)$.

d) Seien $f, g : \mathbb{N} \to \mathbb{R}$ zwei Funktionen mit $f(x) = f_1(x) \cdot f_2(x)$ und $g(x) = g_1(x) \cdot g_2(x)$. Zeige oder widerlege:

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

e) In dieser Aufgabe betrachten wir einen Algorithmus, der einen Graphen mit n Knoten und m gewichteten Kanten als Eingabe bekommt, welcher als Adjazenzliste kodiert ist. Der Algorithmus läuft über die Adjazenzliste und fügt dabei jede gefundene Kante mit ihrem Gewicht hinten in ein Array ein. Dieses Array wird daraufhin mit MERGESORT nach dem Kantengewicht sortiert. Schließlich werden die Elemente des sortierten Arrays der Reihe nach ausgegeben, was pro Element konstante Zeit benötigt.

Welche asymptotische Laufzeit hat der Algorithmus? Welche asymptotische Laufzeit hat der Algorithmus, wenn die Kanten statt in ein Array in einen AVL-Baum eingefügt und dann aufsteigend sortiert ausgegeben werden? *Begründe* deine Antwort!

Aufgabe 4: Rekursionen **(4+3+3+4 Punkte)**

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

$$T(n) = 64 \cdot T\left(\frac{n}{4}\right) + \sqrt{n}\log n + 7n.$$

Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

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

$$T(n) = T\left(\frac{n}{16}\right) + 15n + \sum_{i=1}^{4} T\left(\frac{n}{2^i}\right)$$

Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

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

$$T(n) = 14n \log n + 20T\left(\frac{n}{5}\right) + 4n^2 + 2T\left(\frac{n}{4}\right).$$

Bestimme die Werte aller im Mastertheorem auftretenden Parameter.

d) Auf welche der folgenden Rekursionen lässt sich das Mastertheorem anwenden?

$T(n) = T(n-1) + T(n-2)$

☐ wahr
☐ falsch

$T(n) = n^2 + \sum_{i=1}^{5} T\left(\frac{n}{i}\right)$

☐ wahr
☐ falsch

$T(n) = 42 + T\left(\frac{3n}{2}\right)$

☐ wahr
☐ falsch

$T(n) = 3n^{-1} + T\left(\frac{n}{2}\right)$

☐ wahr
☐ falsch

Aufgabe 5: Mediane **(2+2+12 Punkte)**

a) Welche Laufzeit besitzt der Select-Algorithmus, wenn man Dreiergruppen statt Fünfergruppen benutzt? Gib sowohl die Rekursionsgleichung als auch die asymptotische Laufzeit an!

b) Sei m der Median der Mediane. Wie sind die Mengen X_1, X_2 und X_3 definiert? Wieviele Elemente enthält X_2, wenn alle Elemente paarweise verschieden sind?

c) Bestimme mithilfe des Select-Algorithmus das Rang-16 Element von

$$X = \{14, 6, 1, 11, 4, 20, 3, 2, 19, 9, 5, 7, 18, 16, 8, 12, 17, 10, 13, 15\}$$

Gib dabei alle Aufrufe des Select-Algorithmus und deren Rückgabewert m an. Gib außerdem die Fünfergruppen und die Mengen M', X_1, X_2 und X_3 an.

Aufgabe 6: Sortieren **(7+3+1 Punkte)**

Wir betrachten MERGESORT zum Sortieren von n Zahlen.

$A =$	4	7	5	1	3	2	8	6
1. $A =$								
2. $A =$								
3. $A =$								
4. $A =$								
5. $A =$								
6. $A =$								
7. $A =$								

Abbildung 3: Mergesort im Array A.

a) Sortiere das Array A aus Abbildung 3 mit MERGESORT. Gib (separat und in chronologischer Reihenfolge) das Array A nach jedem Aufruf von MERGE an. Elemente, die nicht zum im jeweiligen Mergeschritt betrachteten Teilarray gehören, müssen nicht angegeben werden. Die Aufrufe von MERGE auf einem Teilarray der Länge 1 müssen nicht angegeben werden. Nutze zur Bearbeitung der Aufgabe die in Abbildung 3 gegebene Tabelle.

b) Wende auf das Array B aus Abbildung 4 PARTITION$(B, 1, 8)$ an. Gib das Ergebnis in der in Abbildung 4 gegebenen Tabelle an. Kennzeichne dabei das Pivotelement und die beiden sich ergebenden Teilarrays. *Hinweis: Es wurde stets das letzte Element des Arrays als Pivotelement genutzt.*

$B =$	1	6	3	8	5	2	7	4
Ergebnis $B =$								

Abbildung 4: Das Array B.

c) Gib die größtmögliche untere Schranke für die Laufzeit (Ω-Notation) eines vergleichsbasierten Sortierverfahrens an.

Aufgabe 7: Algorithmenentwurf **(3+5+2 Punkte)**

Gegeben seien $n = 2k$ Zahlen in einem unsortierten Array A. Können diese Zahlen in k viele Paare aufgeteilt werden, sodass für jedes Paar (a, b) die Summe $a+b$ immer dieselbe Zahl ergibt?

Sei beispielsweise $A = \{11, 3, 1, 7, 5, 9\}$. Diese Zahlen lassen sich in die Paare $(1, 11)$, $(3, 9)$, $(5, 7)$ aufteilen, da $1 + 11 = 12$, $3 + 9 = 12$ und $5 + 7 = 12$.

a) Gib eine kurze Beschreibung eines Algorithmus an, der in Zeit $O(n \log n)$ läuft und *true* zurück gibt, falls eine Aufteilung in k Paare möglich ist.

b) Gib den in a) angegebenen Algorithmus in Pseudocode an.

c) Zeige, dass Dein Algorithmus die Zeitschranke $\mathrm{O}(n \log n)$ einhält.

Aufgabe 8: Kurzfragen **(2+2+2+2+2 Punkte)**

a) Das Löschen eines Elementes aus einem AVL-Baum erfordert höchstens zwei RESTRUCTURE-Operationen. ☐ wahr ☐ falsch

b) Wenn immer das letzte Element der Eingabe als Pivotelement verwendet wird, benötigt QUICKSORT auf einem umgekehrt sortierten Array der Größe n höchstens $O(n \log n)$ Vergleichsoperationen. ☐ wahr ☐ falsch

c) Ein längster Weg zwischen zwei Knoten in einem einfachen Graphen kann durch Tiefensuche in Laufzeit $O(|V| + |E|)$ gefunden werden. ☐ wahr ☐ falsch

d) Das Suchen in einem binären Suchbaum dauert im schlimmsten Fall $O(\log n)$. ☐ wahr ☐ falsch

e) Jeder einfache Graph mit vier Knoten und vier Kanten besitzt einen Hamiltonkreis. ☐ wahr ☐ falsch

Viel Erfolg ☺