Technische Universität Braunschweig
Institut für Betriebssysteme und Rechnerverbund
Abteilung Algorithmik

Sommersemester 2016

Prof. Dr. Sándor P. Fekete
Dr. Christian Scheffer

Klausur
Algorithmen und Datenstrukturen
08. 08. 2016

Name: ....

Vorname: ....

Matr.-Nr.: ....

Studiengang: ....

☐ Bachelor ☐ Master ☐ Andere

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

....
Unterschrift

Hinweise:

- Bitte das Deckblatt vollständig ausfüllen.
- Die Klausur besteht aus 16 Blättern, bitte auf Vollständigkeit überprüfen.
- Erlaubte Hilfsmittel: Keine.
- Eigenes Papier ist nicht erlaubt.
- Die Rückseiten der Blätter dürfen beschrieben werden.
- 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**	**13**	**18**	**10**	**10**	**9**	**14**	**100**
Erreicht									
Note	—	—	—	—	—	—	—	—	

Aufgabe 1: Graphen **(10+3+3 Punkte)**

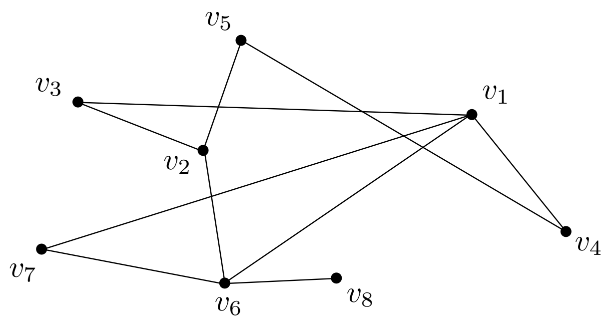


Abbildung 1: Der Graph G

a) Wende Breitensuche 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 jedesmal an, wenn sie sich ändert, und zeichne den gefundenen Baum T.

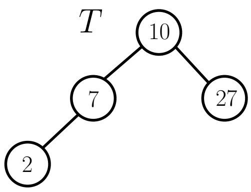
b) Zeichne einen Graphen mit 5 Knoten, der einen Hamiltonkreis, aber keinen Eulerweg hat. Kennzeichne den Hamiltonkreis.

c) Zeige oder widerlege: Ein Hamiltonpfad eines Graphen mit n Knoten hat $n-1$ Kanten.

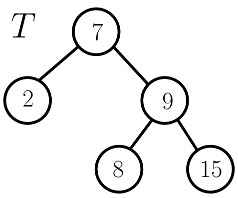
Aufgabe 2: AVL-Bäume **(2+2+2+2+2 Punkte)**

Betrachte in den Aufgabenteilen a) bis e) den Baum, der in der jeweiligen Abbildung dargestellt wird. Führe die, Insert-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 und Restructure in einen separaten Baum:

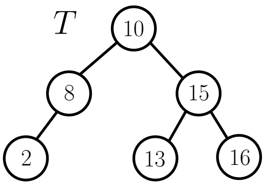
a) Insert$(T, 1)$



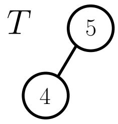
b) Insert$(T, 17)$



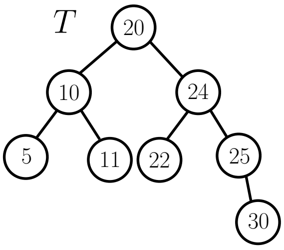
c) Insert$(T, 1)$



d) $\textsc{Insert}(T, 2)$



e) $\textsc{Insert}(T, 32)$



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

Seien $f, g, h : \mathbb{N} \to \mathbb{R}$ drei Funktionen. Zeige oder widerlege die folgenden Aussagen, indem Du die Definition von der Notationen $\mathcal{O}, \Omega, \Theta$ verwendest. Argumentiere hier mithilfe der Werte von n und mithilfe der involvierten Konstanten $(c_0, n_0, ...)$

a) $(g)^4 \in \mathrm{O}(f^2) \Rightarrow g \in \Omega(f)$

b) $g \in \Theta(f), f \in \mathcal{O}(h) \Rightarrow h \in \Theta(g)$

c) Zeige: $2n^3 + n - 3 \in \Theta(n^3)$. Gib dazu explizit geeignete Konstanten c_1, c_2 und n_0 aus der Definition an und zeige, dass sie die Definition erfüllen.

d) Gesucht ist ein Algorithmus, der n Elemente bearbeiten soll. Dazu gibt es zwei Möglichkeiten:

(i) Die Elemente werden der Reihe nach bearbeitet, das dauert $\mathrm{O}(n^4)$ pro Element.

(ii) Die Elemente werden erst sortiert. Dadurch reduziert sich die Bearbeitungszeit pro Element auf $\mathrm{O}(n^2)$.

Welche Laufzeiten ergeben sich für beide Varianten, wenn für (ii) ein möglichst schnelles Sortierverfahren verwendet wird? Begründe Deine Antwort. Welche Vorgehensweise ist asymptotisch schneller?

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

a) Wie lautet das Mastertheorem aus der Vorlesung?

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

$$T(n) = 27 \cdot T\left(\frac{n}{4}\right) + 3n + 229 \cdot T\left(\frac{n}{4}\right).$$

Bestimme die Werte aller im Mastertheorem auftretender Parameter.

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

$$U(n) = n^3 + 4 \cdot U\left(\frac{n}{2}\right) - 19 + 32 \cdot U\left(\frac{n}{4}\right).$$

Bestimme die Werte aller im Mastertheorem auftretender Parameter.

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

$$V(n) = 2 \cdot V\left(\frac{n}{4}\right) + V\left(\frac{n}{5}\right) + 7n^2.$$

Bestimme die Werte aller im Mastertheorem auftretender Parameter.

e) Was sind obere und untere Schranken für die Worst-Case-Laufzeit von QUICKSORT? Wie muss die Eingabe beschaffen sein, damit der Worst-Case erreicht wird (mit Begründung)?

Aufgabe 5: Median **(10 Punkte)**

Gegeben sei die Menge $X = \{x_1, \ldots, x_n\}$ von paarweise verschiedenen natürlichen Zahlen für $n \in \mathbb{N}$ und $k \in \mathbb{N}$ mit $1 \leq k \leq n$.

a) Wie ist das *k-te Element von X* definiert?

b) Beschreibe in eigenen Worten das Vorgehen des SELECT-Algorithmus.

c) Welche Laufzeit $T(n)$ hat der Select-Algorithmus? Gib die rekursive Darstellung von $T(n)$ an und erkläre diese.

d) Welche Auswirkung hat es auf X_2, dass X nur paarweise verschiedene Zahlen enthält?

e) Wende den Select-Algorithmus auf $X = \{4, 2, 10, 5, 9, 1, 6, 8, 11, 7, 14, 3, 15, 12, 13\}$ und $k = 7$ an. Gib hierbei nur für die oberste Rekursionsebene die berechneten Blöcke, M' und m an, indem Du das folgende Ablaufdiagramm komplettierst. Markiere die Teilmenge von X, in der das k-te Element rekursiv weiter gesucht wird.

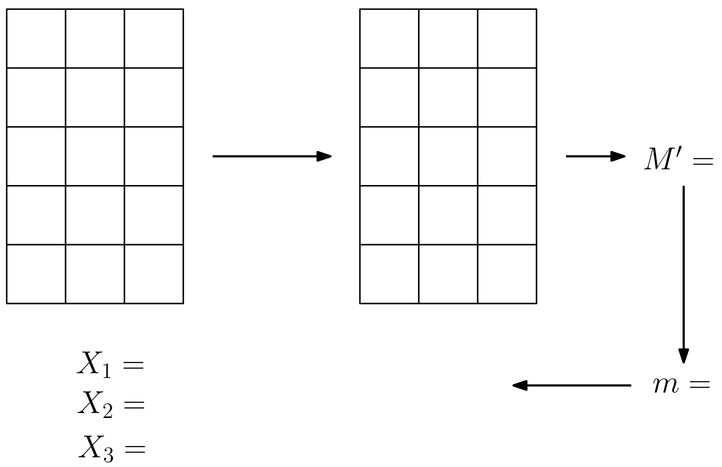


Abbildung 2: Ablaufdiagramm von Select für die oberste Rekursionsebene.

Aufgabe 6: Sortieren **(10 Punkte)**

$A[1]$	$A[2]$	$A[3]$	$A[4]$	$A[5]$	$A[6]$	$A[7]$	$A[8]$
1	9	16	45	2	17	43	82

Abbildung 3: Mergesort im Array A.

Wende die Funktion MERGE(A, 1, 4, 8) aus MERGESORT auf das gefüllte Array oben in Abbildung 3 an. ACHTUNG: Das bedeutet, nicht, dass der gesamte Algorithmus MERGESORT angewendet werden soll. Führe Schritt für Schritt das Mergen von L und R in A durch, in dem Du die unten stehende Tabelle ausführst (pro Schritt eine Zeile). Ein Schritt entspricht dem Überschreiben von $A[k]$.

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

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

Gegeben sei ein binärer Suchbaum B, der nur positive Elemente beinhaltet und n Knoten hat. Gesucht wird der kleinste Schlüssel, der durch 3 ganzzahlig teilbar ist. Falls keiner durch 7 teilbar ist, soll 0 zurückgegeben werden.

a) Gib dafür einen rekursiven Algorithmus über linken und rechten Teilbaum an, der Laufzeit $\mathrm{O}(n)$ hat.

b) Begründe, warum dein Algorithmus in $\mathrm{O}(n)$ liegt.

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

a)	SELECT ist ein rekursiver Algorithmus.	☐ wahr ☐ falsch
b)	Breitensuche wird mit dem Graphen-Scan-Algorithmus mittels eines Stapels realisiert.	☐ wahr ☐ falsch
c)	Jeder AVL-Baum ist ein binärer Suchbaum.	☐ wahr ☐ falsch
d)	Jeder Algorithmus der n Zahlen sortiert benötigt mindestens $\Omega(n \log n)$ Laufzeit.	☐ wahr ☐ falsch
e)	Ein Graph kann nur dann einen Eulerweg haben, wenn es höchstens zwei Knoten mit ungeradem Grad gibt.	☐ wahr ☐ falsch
f)	Ein Graph mit n Knoten kann maximal $2n$ Kanten haben.	☐ wahr ☐ falsch
g)	Breitensuche und Tiefensuch unterscheiden sich in der Datenstruktur, die verwendet wird.	☐ wahr ☐ falsch

Viel Erfolg ☺