Technische Universität Braunschweig
Institut für Betriebssysteme und Rechnerverbund
Abteilung Algorithmik

Prof. Dr. Sándor P. Fekete
Arne Schmidt

Sommersemester 2020

Klausur
Algorithmen und Datenstrukturen
12.08.2020

Name: ....

Vorname: ....

Matr.-Nr.: ....

Studiengang: ....

Klausurraum: ....

Platznummer: ....

Klausurcode:
Dieser wird benötigt, um das Ergebnis der Klausur abzurufen.

☐ Bachelor ☐ Master ☐ Andere

Hinweise:

- Bitte das Deckblatt in Druckschrift vollständig ausfüllen.
- Die Klausur besteht aus 13 Blättern, bitte auf Vollständigkeit überprüfen. Die Heftung darf nicht entfernt werden
- 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	**15**	**10**	**15**	**15**	**15**	**12**	**8**	**10**	**100**
Erreicht									
Note	—	—	—	—	—	—	—	—	

Aufgabe 1: Graphen **(5+4+3+3 Punkte)**

a) Wende den Algorithmus von Fleury zum Finden einer Eulertour auf den Graphen G aus Abbildung 1 an. Starte beim Knoten v_1 und gib die gefundene Eulertour als Knotenliste an. Kommen zu einem Zeitpunkt mehrere Knoten infrage, wähle denjenigen mit dem kleinsten Index.

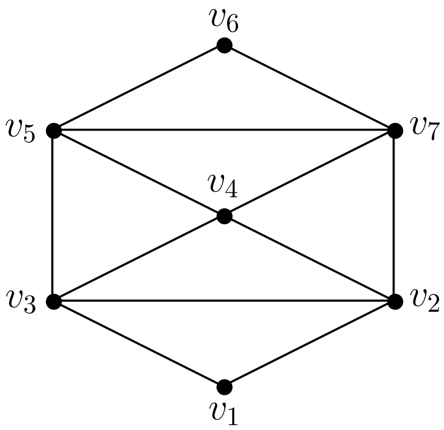


Abbildung 1: Der Graph G

b) Besitzt der Graph H aus Abbildung 2 einen Hamiltonkreis? Begründe deine Antwort.

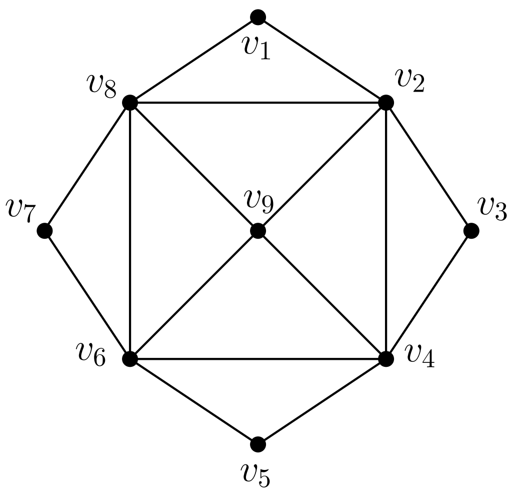


Abbildung 2: Der Graph H

c) Betrachte den Graphen W_5 aus Abbildung 3.
Zeige oder widerlege: Es können Kanten zwischen bestehenden Knoten so eingefügt werden, dass der resultierende (einfache) Graph eine Eulertour besitzt.

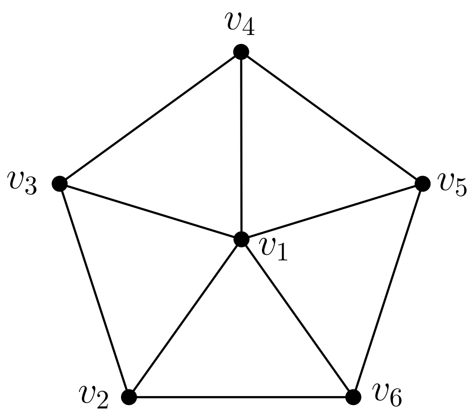


Abbildung 3: Der Graph W_5

d) Betrachte den Graphen W_5 aus Abbildung 4. Lösche drei Kanten aus dem Graphen, sodass der resultierende (zusammenhängende) Graph eine Eulertour enthält. Warum reicht es nicht, weniger Kanten zu löschen?

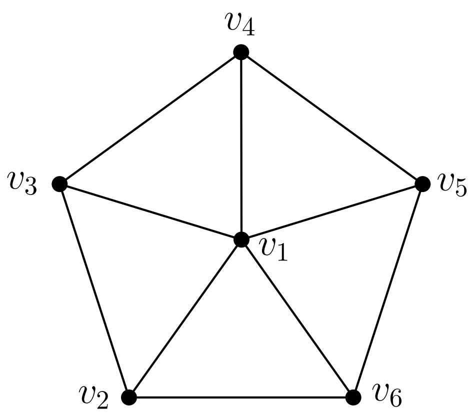
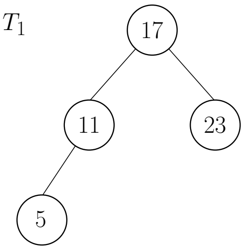


Abbildung 4: Der Graph W_5

Aufgabe 2: Dynamische Datenstrukturen **(2+2+2+4 Punkte)**

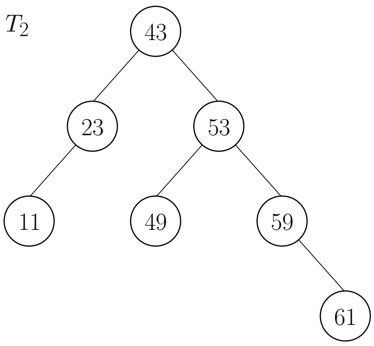
a) Führe Insert(T_1, 6) auf dem AVL-Baum aus. Gib den Baum nach der Einfügeoperation sowie nach jeder Restructure-Operation an.
(Hinweis: Die Einfügeoperation darf in dem angegebenen Baum durchgeführt werden.)

T_1



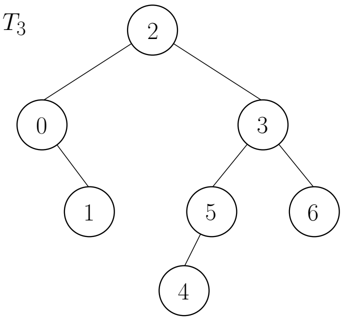
b) Führe Delete(T_2, 43) auf dem AVL-Baum aus. Gib den Baum nach der Löschoperation sowie nach jeder Restructure-Operation an.

T_2



c) Führe DELETE(T_3, 0) auf dem AVL-Baum aus. Gib den Baum nach der Löschoperation sowie nach jeder Restructure-Operation an.

T_3



d) Betrachte das Array $M = [1, 2, 3, 4, 5, 6, 7]$. Wende auf M die Funktion BUILD-MAXHEAP an und gib M nach jedem Tausch zweier Zahlen entweder als Array oder in der Baumstruktur an.

Aufgabe 3: Wachstum von Funktionen **(5+5+5 Punkte)**

a) Kreuze an, in welchen Klassen die angegeben Funktionen liegen.

Klasse / Funktion	$\Omega(n)$	$\Theta(n)$	$\Omega(n \log n)$	$O(n^2)$
2^n				
$7n - 15$				
$17n\sqrt{n}$				
$3(\log n)^2 - \log n$				
$n + \log(n!)$				

b) Seien $f, g : \mathbb{N} \to \mathbb{R}$ Funktionen.
Zeige oder widerlege: $(f(n))^2 \in \Omega(g(n)) \Rightarrow f(n) \in \Omega(g(n))$

c) Seien $f, g, h : \mathbb{N} \to \mathbb{R}$ Funktionen.
Zeige oder widerlege: $f(n) \in O(h(n))$ und $g(n) \in O(h(n)) \Rightarrow f(n) \cdot g(n) \in O((h(n))^2)$

Aufgabe 4: Rekursionen **(4+2+3+3+3 Punkte)**

a) Wie lautet das Mastertheorem aus der Vorlesung?

b) Auf welche der folgenden Rekursionsgleichungen kann das Mastertheorem angewendet werden? Eine Begründung ist nicht erforderlich.

(i) $T(n) = T(n-1) + T(n-2) + 1$ ☐ wahr ☐ falsch

(ii) $T(n) = 3 \cdot T(\frac{n}{2}) + n^{-1}$ ☐ wahr ☐ falsch

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

$$T(n) = 16 \cdot T\left(\frac{n}{16}\right) + 2\sqrt{n}.$$

Bestimme zusätzlich die Werte aller auftretenden Parameter.

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

$$T(n) = 7n + 2 \cdot T\left(\frac{3n}{5}\right) + 6 \cdot T\left(\frac{n}{5}\right) - 21\log n - 5 + n^2.$$

Bestimme zusätzlich die Werte aller auftretenden Parameter.

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

$$T(n) = T\left(\frac{7n}{9}\right) + 2 \cdot T\left(\frac{4n}{9}\right) + 7n^2.$$

Bestimme zusätzlich die Werte aller auftretenden Parameter.

Aufgabe 5: Mediane **(2+6+5+2 Punkte)**

a) Sei X eine Menge von n paarweise verschiedenen natürlichen Zahlen. Wie lautet die Definition eines Rang-k Elements $m \in X$?

b) Beschreibe kurz, wie ein Rang-k Element in einer Menge X von paarweise verschiedenen Zahlen in $O(n)$ Zeit gefunden werden kann.
(Hinweis: Eine Laufzeitanalyse und ein Korrektheitsbeweis sind nicht erforderlich.)

c) Betrachte folgendes Problem.

Eingabe: X eine Menge von n paarweise verschiedenen Elementen und $\ell \in \{1, \ldots, n\}$.

Ausgabe: Die ℓ kleinsten Elemente in X.

(i) Zeige, dass das Problem in $O(n)$ Zeit gelöst werden kann.

(ii) Welche Laufzeit (in Θ-Notation) ergibt sich, wenn man die ℓ kleinsten Elemente in X zusätzlich in sortierter Reihenfolge ausgeben muss?
(Hinweis: Eine Angabe ohne Begründung ist ausreichend.)

Aufgabe 6: Sortieren **(7+5 Punkte)**

a) Betrachte BUBBLESORT aus Algorithmus 1.

```
function BUBBLESORT(A)
    for i = n downto 2 do
        for j = 1 to i − 1 do
            if A[j] > A[j + 1] then
                Vertausche A[j] und A[j + 1]
    return A
```

Algorithmus 1: BUBBLESORT auf einem Array A

Wende BUBBLESORT auf das folgende Array an. Gib das Array jedes Mal an, wenn zwei Elemente vertauscht werden.

$A =$	5	3	1	2	6	4
$A =$						
$A =$						
$A =$						
$A =$						
$A =$						
$A =$						
$A =$						

b) Zeige: Jeder Sortieralgorithmus, der nur benachbarte Felder vertauschen darf, benötigt mindestens $\Omega(n^2)$ Tauschoperationen.

Aufgabe 7: Algorithmenentwurf **(4+4 Punkte)**

Betrachte einen binären Suchbaum B mit Wurzel $\mathrm{root}(B)$ und Schlüsseln $S(v)$ für alle $v \in B$.

a) Entwirf einen **rekursiven** Algorithmus, der die Elemente in B sortiert ausgibt. Wir nennen diese Funktion INORDER. Es darf angenommen werden, dass wir mit INORDER$(\mathrm{root}(B))$ starten.
(Hinweis: Ein Wert X kann mit **output** X ausgegeben werden.)

b) Zeige, dass INORDER $O(n)$ Zeit benötigt.

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

Kreuze die korrekten Aussagen an. Es gibt nur Punkte für vollständig korrekt angekreuzte Teilaufgaben.
(Hinweis: In jeder Teilaufgabe ist immer **mindestens** eine Aussage korrekt.)

a) Welche Bedingungen sind notwendig, damit ein Graph (ohne isolierte Knoten) eine Eulertour besitzt?

- Der Graph ist zusammenhängend. ☐
- Der Graph besitzt keine ungeraden Knoten. ☐
- Der Graph ist einfach. ☐

b) Jeder Hamiltonpfad ist…

- …ein Kantenfolge. ☐
- …ein Weg. ☐
- …ein Pfad. ☐

c) Aus einer unsortierten Menge von n vergleichbaren Elementen kann man in $O(n)$ Zeit…

- …eine Liste erstellen. ☐
- …einen Max-Heap erstellen. ☐
- …einen binären Suchbaum erstellen. ☐

d) Wofür ist Tiefensuche geeignet?

- In einem Graphen kürzeste Wege finden ☐
- Aus einem Labyrinth herausfinden ☐
- Einen Graphen auf Zusammenhang testen ☐

e) Welche der folgenden Sortieralgorithmen besitzen eine Laufzeit von $O(n \log n)$?

- Mergesort ☐
- Quicksort ☐
- Insertionsort ☐

Viel Erfolg ☺