Technische Universität Braunschweig
Institut für Betriebssysteme und Rechnerverbund
Abteilung Algorithmik

Sommersemester 2022

Prof. Dr. Sándor P. Fekete
Matthias Konitzny
Dr. Arne Schmidt

Klausur
Algorithmen und Datenstrukturen
17.08.2022

Name: ....

Vorname: ....

Matr.-Nr.: ....

Studiengang: ....

☐ Bachelor ☐ Master ☐ Andere

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

Hinweise:

- Bitte das Deckblatt in Druckschrift vollständig ausfüllen.
- Die Klausur besteht aus 12 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	**20**	**14**	**15**	**9**	**12**	**13**	**7**	**10**	**100**
Erreicht									

Aufgabe 1: Graphen **(8+2+5+5 Punkte)**

a) Betrachte den Graphen G aus Abbildung 1. Wende den Algorithmus von Fleury auf G an, um eine Eulertour zu bestimmen. Starte dabei bei Knoten v_1 und gib die Eulertour als Knotenliste an. Kommen in einem Schritt des Algorithmus mehrere Knoten in Frage, wähle denjenigen mit dem kleinsten Index.

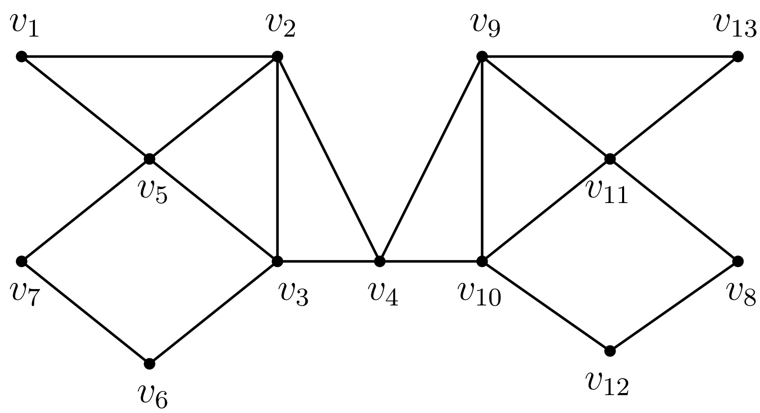


Abbildung 1: Der Graph G

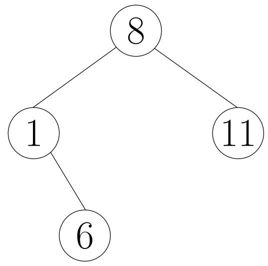
b) Welche zwei Bedingungen müssen erfüllt sein, damit ein Graph **immer** eine Eulertour besitzt?

c) Zeige: Jeder zusammenhängende Graph kann so mit Kanten (keine Knoten) erweitert werden, sodass der erhaltene, nicht notwendigerweise einfache Graph eine Eulertour enthält.

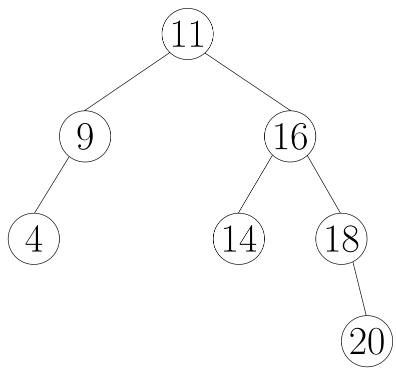
d) Gilt c) weiterhin, wenn der erhaltene Graph einfach sein muss? Begründe deine Antwort.

Aufgabe 2: Dynamische Datenstrukturen **(2+2+2+3+5 Punkte)**

a) Führe $\textsc{Insert}(T_1, 4)$ 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.)



b) Führe $\textsc{Delete}(T_2, 11)$ auf dem AVL-Baum aus. Gib den Baum nach der Löschoperation sowie nach jeder Restructure-Operation an.



c) Wie lange benötigt das Einfügen eines Elementes in eine einfach verkettete Liste? Begründe deine Antwort.

d) Wie lange benötigt das Löschen eines identifizierten Elementes aus einer einfach verketteten Liste? Begründe deine Antwort.

e) Nenne zwei weitere in der Vorlesung vorgestellte Listentypen. Beschreibe jeweils wie die Elemente dieser Listen im Speicher verwaltet werden.

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

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

$f(n)$	$O(n)$	$\Omega(n)$	$O(n \log n)$	$\Omega(n \log n)$	$O(n^2)$	$\Omega(n^2)$
$3n + 6$						
$19n^2 - 30n$						
2^n						
$\log(n!)$						
$\frac{12n^2-5}{3n \log n+5}$						

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

c) Bestimme geeignete Konstanten, um zu zeigen, dass folgende Aussage korrekt ist.

$$f(n) := 4n^2 - 8n + 7 \in \Omega(n^2)$$

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

Bestimme in den Teilaufgabe jeweils mit Hilfe des Mastertheorems das asymptotische Wachstum der angegebenen rekursiven Funktion. Gib jeweils die Werte aller auftretenden Parameter an.

a)

$$T(n) = T\left(\frac{2n}{3}\right) + 5 \cdot T\left(\frac{n}{3}\right) + 2\sqrt{n} + 7n^2$$

b)

$$U(n) = 27 \cdot U\left(\frac{n}{3}\right) - 21 \log n - 5 + 4n^2$$

c)

$$T(n) = 5 \cdot T\left(\frac{n}{3}\right) + 12 \cdot T\left(\frac{n}{6}\right) + 7n^2$$

Aufgabe 5: Mediane **(2+1+6+3 Punkte)**

a) Sei X eine Menge von paarweise verschiedenen natürliche Zahlen. Wie lautet die Definition eines Rang-k Elements $m \in X$? Wie lautet weiterhin die Definition des Median der Menge X?

b) Welche Zahlen sind ein Median der Menge $X = \{5, 3, 2, 100, 12, 7\}$?

c) Beschreibe die Schritte die benötigt werden, um das Rang-k Element in einer Menge X von paarweise verschiedenen Zahlen in $O(n)$ Zeit zu finden.
(Hinweis: Ein Korrektheitsbeweis ist nicht erforderlich. Ein Laufzeitbeweis ist bei d) erforderlich.)

d) Wie lautet die Rekursionsgleichung zur Bestimmung der Laufzeit um ein Rang k Element zu finden? Ordne die einzelnen Teile der Rekursionsgleichung den Schritten aus c) zu und gib außerdem die asymptotische Laufzeit an.

Aufgabe 6: Sortieren **(6+4+3 Punkte)**

a) Sortiere das Array aus Abbildung 2 mit dem QuickSort-Algorithmus aus der Vorlesung. Gib das Ergebnis von jedem Partition-Aufruf an, sofern sich das Array ändert. Markiere außerdem das genutzte Pivot-Element.
(Hinweis: Der Platz unter der Tabelle kann für Nebenrechnungen verwendet werden.)

$A =$	9	3	8	2	4	6	5	1	7
1. $A =$									
2. $A =$									
3. $A =$									
4. $A =$									
5. $A =$									

Abbildung 2: QuickSort auf Array A.

b) Welche Eigenschaft erfüllt ein stabiler Sortieralgorithmus? Ist QuickSort in der aus der Vorlesung bekannten Version stabil? Begründe deine Antwort.

c) Welche Laufzeit besitzt QuickSort für n Zahlen im Worst-Case? Begründe kurz unter welchen Voraussetzungen dieser schlechteste Fall eintritt.

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

a) Gegeben sei ein binärer Suchbaum B mit Wurzelknoten r. Entwirf einen **rekursiven** Algorithmus in Pseudocode (maximal 10 Zeilen), welcher die Anzahl Knoten ausgehend von r in $O(n)$ Zeit bestimmt.

Wir nennen die Funktion COUNT. Ein typischer Aufruf der Funktion bekommt den Wurzelknoten mittels COUNT(r) übergeben.

b) Begründe kurz, warum die Laufzeit $O(n)$ eingehalten wird.

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) In einem einfachen Graphen ...

- ... ist die Anzahl der Knoten mit ungeradem Grad immer gerade. ☐
- ... gibt es keine isolierten Knoten. ☐
- ... gibt es weder doppelte Kanten noch Schleifen. ☐

b) Die Adjazenzmatrix für einen Graphen mit n Knoten und m Kanten ...

- ... benötigt immer $O(m)$ viel Speicherplatz. ☐
- ... benötigt für $m \in O(n)$ mehr Speicherplatz als eine Adjazenzliste. ☐
- ... enthält auf der Diagonalen ausschließlich Nullen. ☐

c) Die Breitensuche ...

- ... eignet sich um aus einem Labyrinth herauszufinden. ☐
- ... nutzt eine Warteschlange als Datenstruktur. ☐
- ... erzeugt immer einen eindeutigen Suchbaum. ☐

d) Mergesort...

- ... basiert auf dem Prinzip „Teile und Herrsche“. ☐
- ... ist ein vergleichsbasiertes Sortierverfahren. ☐
- ... hat im Best Case eine Laufzeit von $O(n)$. ☐

e) Wird eine Subroutine, welche die Laufzeit $O(n)$ besitzt, $O(\log n)$ mal wiederholt, so ist die asymptotische Gesamtlaufzeit...

- ... unbekannt, da zunächst die absolute Laufzeit bekannt sein muss. ☐
- ... $O(n)$. ☐
- ... $O(n \log n)$. ☐

Viel Erfolg ☺