Technische Universität Braunschweig
Institut für Betriebssysteme und Rechnerverbund
Abteilung Algorithmik

Prof. Dr. Sándor Fekete
Ramin Kosfeld
Chek-Manh Loi

Wintersemester 2025/2026

Klausur
Algorithmen und Datenstrukturen
18.02.2026

Nachname: .....

Vorname: .....

Matr.-Nr.: .....

Studiengang: .....

☐ Bachelor ☐ Master ☐ Andere: .....

Hinweise:

- Bitte das Deckblatt in Druckschrift vollständig ausfüllen.
- Die Klausur besteht aus 13 Blättern, bitte auf Vollständigkeit überprüfen.
- Die Bearbeitungszeit für die Klausur beträgt 120 Minuten.

- 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.

Aufgabe	1	2	3	4	5	6	7	8	Σ
Max	14	17	13	13	7	11	13	10	98
Erreicht									

Aufgabe 1: Graphenscan **(8+3+3 Punkte)**

a) Betrachte den Graphen G aus Abbildung 1. Wende Breiten- und Tiefensuche mit Startknoten v_1 auf G an und zeichne die resultierenden Bäume in Abbildung 2. Falls zu einem Zeitpunkt mehrere Knoten für den nächsten Schritt in Frage kommen, wähle denjenigen mit dem kleinsten Index.

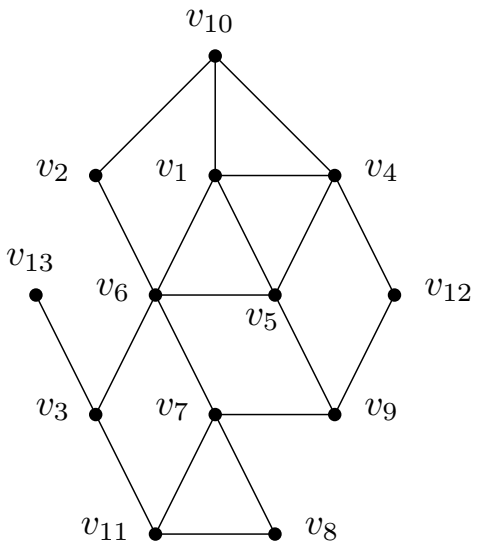
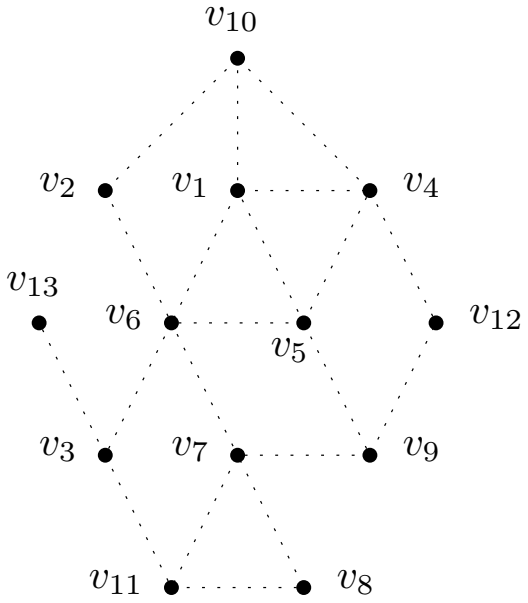
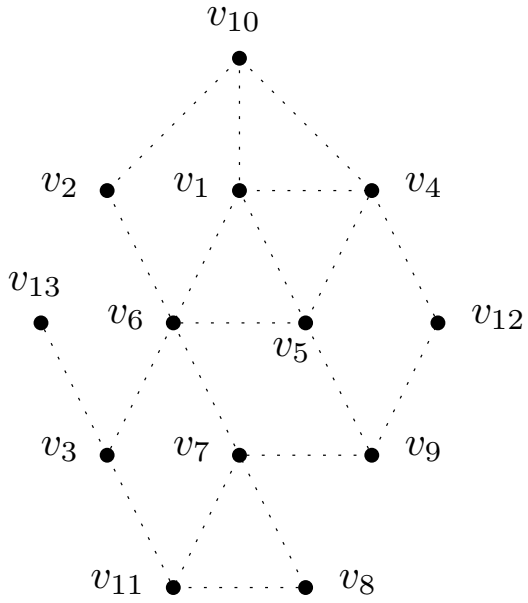


Abbildung 1: Der Graph G.



(a) Breitensuchbaum von G.



(b) Tiefensuchbaum von G.

Abbildung 2

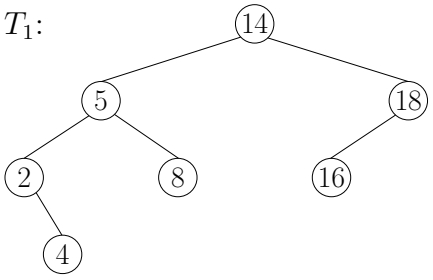
b) In der Vorlesung haben wir gesehen, wie man Adjazenzmatrizen für einfache, ungerichtete Graphen erstellt. Beschreibe, wie man mit einer Adjazenzmatrix auch ungerichtete Graphen darstellen kann, die *nicht* einfach sind.

c) Sei I ein Graph mit mindestens 3 Knoten. Zeige: Existiert in I ein Hamiltonkreis H, dann zerfällt I in höchstens k Zusammenhangskomponenten, wenn k Knoten entfernt werden.

Aufgabe 2: Dynamische Datenstrukturen **(2+3+4+4+4 Punkte)**

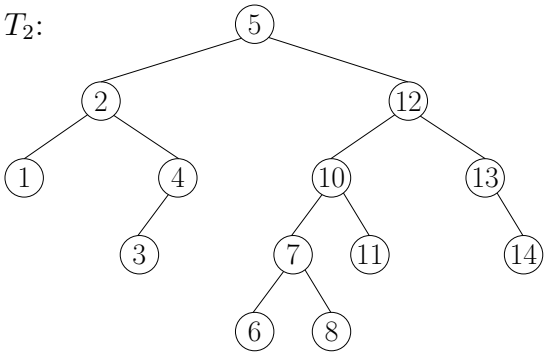
a) Führe INSERT(T_1, 3) auf dem AVL-Baum aus. Gib den Baum nach der Einfügeoperation sowie nach jeder RESTRUCTURE-Operation an.
(Hinweis: Die Einfügeoperation darf in dem angegebenen Baum durchgeführt werden.)

T_1:



b) Führe DELETE(T_2, 1) auf dem AVL-Baum aus. Gib den Baum nach der Löschoperation sowie nach jeder RESTRUCTURE-Operation an. (Hinweis: Die Löschoperation darf in dem angegebenen Baum durchgeführt werden.)

T_2:



c) Sei x ein Knoten in einem binären Suchbaum T. Beschreibe kurz, wie man in $\mathcal{O}(h)$ Zeit den Knoten x aus T löscht. Dabei ist h die Höhe des Baumes. Unterscheide zwischen den drei Fällen dass x ein Blatt ist, x einen Kindknoten oder zwei Kindknoten besitzt. (Hinweis: Hier ist kein Pseudocode nötig.)

d) Zeichne einen Max-Heap in Baumstruktur, der genau die folgenden Zahlen enthält:

$$1, 33, 7, 4, 2, 0, 6, 9, 67$$

e) Gib den Pseudocode für den Algorithmus ListSearch(L, k) an, der eine verkettete Liste als Eingabe bekommt und das erste Listenelement mit Wert k ausgibt. Ist k nicht in der Liste enthalten, wird `NIL` zurückgegeben.

Aufgabe 3: Wachstum von Funktionen **(5+4+4 Punkte)**

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

$f(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n \log n)$	$\mathcal{O}(n^2)$	$\Omega(n)$	$\Omega(n \log n)$	$\Omega(n^2)$
$n^2 - 20n$						
$\log(n) + 3n$						
$\log(n!)$						
$\frac{1}{n^2}$						
$\frac{n^2}{\log(n)}$						

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

$$f(n) := 2n^3 - 3n + 5\log(n) \in \Omega(n^3)$$

c) Seien $f, g, h : \mathbb{N} \to \mathbb{R}$ Funktionen.
Zeige oder widerlege: $f(n) \in \Theta(g(n)), f(n) \in \mathcal{O}(h(n)) \Rightarrow h(n) \in \Omega(g(n))$

Aufgabe 4: Rekursionen **(13 Punkte)**

Bestimme das asymptotische Wachstum der folgenden Funktionen mithilfe des in der Vorlesung vorgestellten Mastertheorems, oder begründe in einem Satz, warum man das Theorem nicht anwenden kann. Gib beim Anwenden alle im Mastertheorem auftretenden Parameter an.

a)

$$G(n) = 4n^2 + G\left(\frac{1}{2}n\right) + 3n \log n + G\left(\frac{2n}{3}\right) + 2n^2 \log n$$

b)

$$T(n) = 7 \cdot T\left(\frac{n}{3}\right) + 8 \cdot T\left(\frac{n}{6}\right) + 4n$$

c)

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

d)

$$Q(n) = 26n^3 + 26 \cdot Q\left(\frac{n}{3}\right) + 26\sqrt{n} - 26$$

e)

$$L(n) = 2 \cdot L\left(\frac{n}{16}\right) + L\left(\frac{n}{4}\right) + 14\sqrt{n} + 2e^{\pi}$$

Aufgabe 5: Mediane **(2+1+4 Punkte)**

a) Betreachte die untenstehenden, vorsortieren Fünfergruppen.

(i) Markiere den Median der Mediane m. Markiere außerdem in jeder Gruppe, welche Elemente größer bzw. kleiner als m sind.

6	5	3	10	11	1	2	13	14
20	7	8	22	23	9	4	17	19
21	15	12	24	29	28	16	32	30
27	34	33	25	41	39	18	35	31
44	38	40	26	42	43	45	37	36

(ii) Welche Elemente müssen wir weiterhin betrachten, wenn wir aus den 45 Elementen das Element mit Rang 42 bestimmen wollen? (Hinweis: Die Zahlen 1 bis 45 tauchen jeweils einmal auf.)

b) Mit dem Median der Mediane von Fünfergruppen kann das Rang-k Element in $\mathcal{O}(n)$ Zeit gefunden werden. Argumentiere anhand der Rekursionsgleichung, ob sich die Laufzeit ändert, wenn stattdessen Gruppen der Größe 7 verwendet werden.

Aufgabe 6: Sortieren **(3+5+3 Punkte)**

a) Wende MERGE$(B, 1, 4, 8)$ auf das Array B aus Abbildung 3 an. Trage das Ergebnis in die in Abbildung 3 vorgesehene Zeile ein.

index	1	2	3	4	5	6	7	8
B	2	4	6	9	2	3	5	8
Ergebnis B								

Abbildung 3: Das Array B

b) Wir betrachten RADIXSORT zum Sortieren der in Abbildung 4 gegebenen Zahlen. Sortiere das Array A aus Abbildung 4 mit RADIXSORT. Gib A nach jeder Iteration an; nutze dazu Abbildung 4:

737			
912			
816			
961			
862			
314			

Abbildung 4: Radixsort auf Array A (linke Spalte).

c) Sei A ein unsortiertes Array mit n Elementen von ganzen Zahlen. Zeige oder widerlege: Das Entscheiden, ob A eine Zahl mehrfach enthält, hat immer eine Laufzeit in $\Omega(n^2)$.

Aufgabe 7: Algorithmenverständnis **(1+2+2+3+5 Punkte)**

Betrachte folgenden Algorithmus.

(Hinweis: In diesem Pseudocode hat das erste Element im Array den Index 1.)

Eingabe: Ein Array A mit n natürlichen Zahlen.
Ausgabe: TRUE oder FALSE.

```
function CheckArray(A)
    x ← 0
    for i = 1, ..., n − 1 do
        x ← x + A[i]
        y ← 0
        for j = i + 1, ..., n do
            y ← y + A[j]
        if x = y then
            return True
    return False
```

Algorithmus 1

a) Welche Ausgabe liefert der Algorithmus 1 für das Array $A = [2, 0, 3, 4, 1]$?

b) Beschreibe, welche Eigenschaft des Arrays A der Algorithmus 1 überprüft.

c) Welche asymptotische worst-case Laufzeit in Abhängigkeit von der Anzahl der Elemente n hat der Algorithmus 1? Gib eine kurze Begründung an.

d) Gib an, wie viele Zuweisungen (in Abhängigkeit von n) in Algorithmus 1 maximal und minimal auf die Variablen x und y ausgeführt werden. (Hinweis: Eine asymptotische Einordnung ist ausreichend, wenn das einfacher ist.)

e) Gibt einen Algorithmus in Pseudocode an, der die gleiche Eigenschaft wie Algorithmus 1 in asymptotisch besserer Zeit überprüft. Gib außerdem die asymptotische Laufzeit deines Algorithmus an.

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

Kreuze jeweils alle korrekten Aussagen an. Es gibt nur Punkte für vollständig korrekt angekreuzte Teilaufgaben.

(Hinweis: In jeder Teilaufgabe ist immer mindestens eine Aussage korrekt.)

a) In welcher Datenstruktur kann das größte Element in $\mathcal{O}(1)$ Zeit gefunden werden?

- AVL-Baum ☐
- Max-Heap ☐
- Stapel ☐

b) Es gibt AVL-Bäume der Höhe h, die...

- ...3^h Knoten besitzen. ☐
- ...$\Omega(h)$ Restructureoperationen nach einer Einfügeoperation benötigen. ☐
- ...$\Omega(h)$ Restructureoperationen nach einer Löschoperation benötigen. ☐

c) In welchen Fällen besitzt Mergesort eine Laufzeit von $\mathcal{O}(n \log n)$?

- Im Best Case. ☐
- Im Average Case. ☐
- Im Worst Case. ☐

d) Jeder Pfad in einem Graphen G...

- ...besucht keinen Knoten doppelt. ☐
- ...besucht jede Kante höchstens einmal. ☐
- ...ist ein Hamiltonpfad. ☐

e) Welche Laufzeitschranken sind korrekt?

- Vergleichsbasiertes Sortieren von n Zahlen: $\Omega(n^2)$ ☐
- Finden einer Eulertour in einem Graphen: $\mathcal{O}(n + m)$ ☐
- Test auf Zusammenhang eines Graphen: $\mathcal{O}(n + m)$ ☐

Viel Erfolg ☺