Algorithmen und Datenstrukturen

Take-Home-Exam, 1. Termin, WS 2020/21

Das Take-Home-Exam (THE) besteht aus **9 Aufgaben**. Zu jeder Aufgabe ist die erreichbare Punktzahl angegeben. Insgesamt sind maximal 100 Punkte erreichbar. Sie bestehen das THE, wenn Sie mindestens 50 Punkte erzielen. Die Bearbeitungszeit beträgt **2 Stunden**. Ihre Lösung muss bis spätestens **11:40 Uhr** als PDF gescannt und in das Moodle Elemente „AD-Klausur (1. Termin)“ hochgeladen sein. Benennen Sie die Datei nach dem Schema `<nachname>-<vorname>-<matrikelnummer>.pdf`.

Die Lösungen zu den Aufgaben müssen **handschriftlich** auf **Papier** angefertigt werden. Sie dürfen eigenes (anfangs unbeschriebenes) Papier verwenden oder dieses PDF ausdrucken und Ihre Lösungen unter die Aufgaben bzw. auf die freigelassenen Seiten schreiben (fügen Sie, falls notwendig, eigenes Papier hinzu). Schreiben Sie auf **jede Seite** Ihrer Lösung lesbar Ihren **Namen** sowie **Matrikelnummer** und nummerieren Sie die Seiten durch (falls Sie eigenes Papier benutzen, dürfen Sie diese Daten vorausfüllen).

Zugelassene Hilfsmittel sind das Buch „Introduction to Algorithms“ (von Cormen, Leiserson, Rivest und Stein), die diesjährigen Vorlesungsfolien und Übungsblätter sowie eine eigenhändig angefertigte Zusammenfassung der Vorlesung.

Machen Sie klar ersichtlich, zu welcher Aufgabe eine Lösung gehört. Werden zu einer Aufgabe zwei Lösungen angegeben, so gilt die Aufgabe als nicht gelöst. Ergebnisse, Algorithmen sowie Datenstrukturen aus den Vorlesungsfolien dürfen zitiert werden. Alle weiteren Ergebnisse dürfen **nicht** zitiert werden und müssen ggfs. neu hergeleitet werden. Das heißt es reicht nicht zu sagen „Das gilt gemäß Aufgabe 23 aus den Übungen.“ oder „Dies folgt nach Theorem 42 aus dem Cormen.“.

Viel Erfolg!

1	2	3	4	5	6	7	8	9	Σ

Klausur: ______ Bonus: ______ Endnote: ______

Name:	Matrikelnummer:

Aufgabe 1

______/ 6 P.

Betrachten Sie den Graphen in Abbildung 1. Bestimmen Sie mit Hilfe des Algorithmus von Kruskal einen minimalen Spannbaum des Graphen und skizzieren diesen. Geben Sie zusätzlich die Reihenfolge an, in der die Kanten des Spannbaums gemäß Algorithmus hinzugefügt werden.

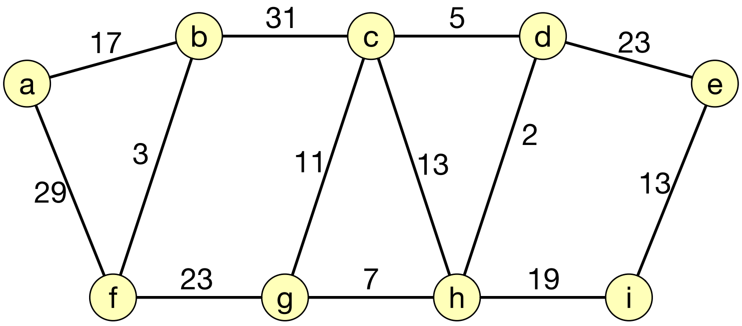


Abbildung 1

Name:	Matrikelnummer:

Name:	Matrikelnummer:

Aufgabe 2

___/ 8 P.

Gegeben sei ein AVL-Baum für die Schlüssel $\{2, 3, 5, 7, 11, 13, 17, 19\}$. Der Zustand der Datenstruktur ist in Abbildung 2 dargestellt. Skizzieren Sie die Datenstruktur nach dem Einfügen eines Elements mit Schlüssel 6 gemäß der Einfügeoperation aus der Vorlesung. Machen Sie dabei den jeweiligen Zustand vor und nach eventuell notwendigen Balancierungsschritten deutlich.

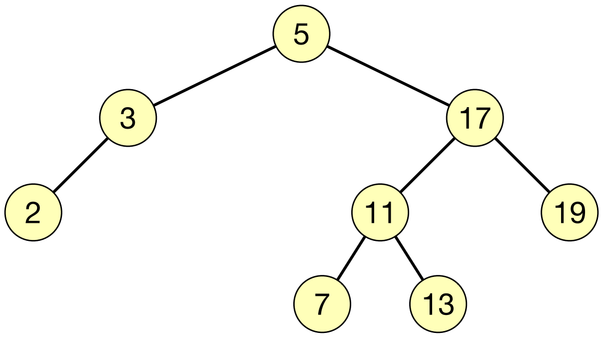


Abbildung 2

Name:	Matrikelnummer:

Name:	Matrikelnummer:

Aufgabe 3

___/ 12 P.

Jede der folgenden Teilaufgaben spezifiziert zwei Funktionen $f(n)$ und $g(n)$. Bestimmen Sie jeweils, ob die vier Relationen $f(n) = \mathrm{o}(g(n))$, $f(n) = \mathrm{O}(g(n))$, $f(n) = \Omega(g(n))$ und $f(n) = \omega(g(n))$ gelten (es sind also pro Teilaufgabe *vier* Antworten zu geben). Beweisen Sie Ihre Antwort jeweils kurz. Nutzen Sie dazu die Eigenschaften der asymptotischen Landau-Notation aus der Vorlesung (wie z. B. die Charakterisierung über Grenzwerte).

(a) $f(n) = n \cdot (\log n)^{42}$ und $g(n) = n^{1.23}$

(b) $f(n) = n \cdot |\sin n|$ und $g(n) = \sqrt{n}$

(c) $f(n) = n^2 + n^{3/2} + n \cdot \log n$ und $g(n) = \binom{n}{2}$

(d) $f(n) = 8^{\log n}$ und $g(n) = n^3 \cdot \log n$

Bemerkung. Der Ausdruck $\binom{n}{k}$ bezeichnet den Binomialkoeffizienten n über k. Die Basis des Logarithmus in Ausdrücken der Form $\log n$ ist 2.

Name:	Matrikelnummer:

Aufgabe 4

____/ 8 P.

Wir betrachten die Datenstruktur für disjunkte dynamische Mengen wie in der Vorlesung definiert. Die Grundmenge (das Universum) sei $U = \{ a, b, \ldots, z \}$, also die Menge der 26 Kleinbuchstaben a bis z.

(a) Geben Sie eine Folge von Aufrufen der Operationen MakeSet und Union an, welche die disjunkten Mengen $S_1 = \{ a \}$, $S_2 = \{ b, c, d \}$ sowie $S_3 = \{ e, f \}$ erzeugt. (2 Punkte)

(b) Skizzieren Sie die Mengenobjekte der Datenstruktur, nachdem die disjunkten Mengen gemäß Ihrer Lösung zu Punkt (a) erzeugt wurden. Eine mögliche Darstellung ist in Abbildung 3 zu sehen. (2 Punkte)

(c) Geben Sie jeweils den Rückgabewert des Aufrufes FindSet(a) und des Aufrufes FindSet(d) an, nachdem die disjunkten Mengen gemäß Ihrer Lösung zu Punkt (a) erzeugt wurden. (2 Punkte)

(d) Skizzieren Sie das Mengenobjekt, welches entsteht, wenn nach Ihrer Lösung zu Punkt (a) die Operation Union(c, f) ausgeführt wird. Nutzen Sie wieder eine Darstellung ähnlich zu Abbildung 3. (2 Punkte)

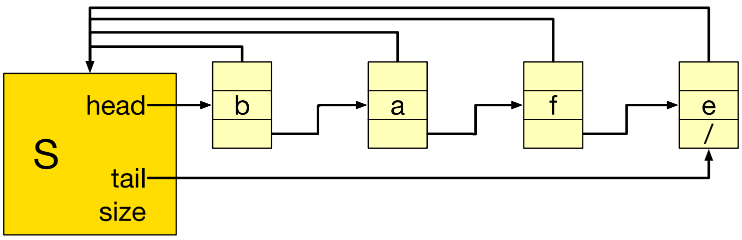


Abbildung 3: Skizze eines Mengenobjektes für die Menge $S = \{ a, b, e, f \}$.

Name:	Matrikelnummer:

Aufgabe 5

Betrachten Sie den Divide & Conquer Algorithmus, dessen Pseudocode in Algorithmus 1 gegeben ist und lösen Sie folgende Aufgaben: ___/ 12 P.

(a) Geben Sie eine Rekursionsgleichung für die Laufzeit $T(n)$ des Algorithmus bei Eingabe $(A, 1, n)$ für ein Array A der Länge $n \in \mathbb{N}$ an. (6 Punkte)

(b) Analysieren Sie die Laufzeit von Algorithmus 1 möglichst genau in der O-Notation mit Hilfe der *Substitutionsmethode*. Sie können dabei annehmen, dass n eine Zweierpotenz ist. (6 Punkte)

Bemerkung. Ein Induktionsbeweis der Laufzeit ist *nicht* notwendig, aber aus Ihrer Rechnung muss klar hervorgehen, wie Sie zu Ihrer Lösung kommen.

Algorithmus 1: ALGO(A, l, r)

```
n ← r − l + 1
if n ≤ 2
    if n = 2: return |A[r] − A[l]|
    else:     return 0
else
    p ← ⌊(l + r)/2⌋
    a ← Algo(A, l, p)
    b ← Algo(A, p + 1, r)
    if a > b: c ← a
    else:     c ← b
    for i ← l to p
        for j ← p + 1 to r
            x ← |A[i] − A[j]|
            if x > c: c ← x
    return c
```

Name:	Matrikelnummer:

Aufgabe 6

___/ 15 P.

Gesucht ist eine Datenstruktur, welche die drei folgenden Operationen unterstützt:

- EINFÜGEN(x): Fügt die Zahl x in die Datenstruktur ein. Sie können zur Vereinfachung davon ausgehen, dass kein Element jemals doppelt in die Datenstruktur eingefügt wird.
- LÖSCHEN(x): Entfernt x, falls sich x in der Datenstruktur befindet. Andernfalls bleibt die Datenstruktur unverändert.
- RANG(x): Gibt die Anzahl der Elemente der Datenstruktur zurück, welche kleiner oder gleich x sind. Befinden sich z. B. die Zahlen $\{2, 3, 5, 7, 11, 13\}$ in der Datenstruktur, so soll RANG$(5) = 3$ und RANG$(17) = 6$ gelten.

Die Datenstruktur soll alle drei Operationen in Laufzeit $O(\log n)$ unterstützen. Dabei bezeichnet n die Anzahl der Elemente, die sich aktuell in der Datenstruktur befinden.

Beschreiben Sie in wenigen kurzen Sätzen, wie Ihre Datenstruktur aufgebaut ist und wie die angegebenen Operationen realisiert werden. Hierbei ist *kein* Pseudocode gefordert. Es soll jedoch klar werden, dass die Datenstruktur korrekt arbeitet und die geforderte Laufzeit eingehalten wird.

Name:	Matrikelnummer:

Name:	Matrikelnummer:

Aufgabe 7

/ 15 P.

Ein Unternehmen erhält die Genehmigung, eine Landstraße auf einer Straßenseite mit Werbetafeln zu versehen. Entlang der $n \in \mathbb{N}$ Kilometer langen Landstraße gibt es im Abstand von je genau einem Kilometer $n+1$ geeignete Standorte. Für den Standort bei Kilometer $i \in \{0, 1, \ldots, n\}$ sind Werbekunden bereit, einen Betrag $w_i > 0$ an das Unternehmen zu zahlen. Allerdings besagt die Straßenverkehrsordnung, dass zwischen je zwei Werbetafeln mindestens zwei Kilometer Abstand bestehen müssen. Errichtet das Unternehmen also bei Kilometer i eine Werbetafel, so darf bei Kilometer $i-1$ und $i+1$ keine Werbetafel errichtet werden.

Helfen Sie dem Unternehmen anhand der folgenden Schritte eine Auswahl an Standorten zu bestimmen, die den Gesamtgewinn aus der Vermietung der Standorte maximiert.

(a) Für $k \in \{0, 1, \ldots, n\}$ bezeichne $\mathrm{OPT}(k)$ den maximal erzielbaren Gesamtgewinn, wenn das Unternehmen *nur* die Standorte an den Kilometern 0, 1, ..., k für die Errichtung von Werbetafeln in Betracht zieht. Finden Sie eine rekursive Formulierung für $\mathrm{OPT}(k)$ und begründen Sie kurz deren Korrektheit. (8 Punkte)

(b) Bezeichne $W[0 \ldots n]$ ein Array der Länge $n+1$, so dass $W[i] = w_i$ dem Eintrag des Standortes bei Kilometer i entspricht. Beschreiben Sie in Pseudocode einen Algorithmus, der mittels dynamischer Programmierung bei Eingabe (W, n) in Zeit $\mathrm{O}(n)$ den maximalen Gesamtgewinn, den das Unternehmen erzielen kann, ermittelt. Die Liste der Standorte, die dieses Maximum erreicht, muss *nicht* bestimmt werden. (7 Punkte)

Name:	Matrikelnummer:

Name:	Matrikelnummer:

Aufgabe 8

_____/ 12 P.

Das *Tankstopp-Problem* modelliert die folgende Situation: Auf der Autofahrt von Ihrem Startpunkt t_0 bis zu Ihrem Ziel t_n wollen Sie so wenige Tankstopps wie möglich einlegen. Unterwegs kommen Sie der Reihe nach an den Tankstellen $t_1, t_2, \dots, t_{n-1}$ vorbei. Sie starten mit vollem Tank und eine Tankfüllung Ihres Wagens reicht für D Kilometer. Der Abstand zwischen den Tankstellen t_{i-1} und t_i beträgt d_i Kilometer, wobei $0 < d_i \leq D$ gilt. Betrachten Sie die folgende Greedy-Strategie:

> *„Fahre stets so weit wie möglich ohne Tankstopp und tanke erst an der am weitesten entfernten noch erreichbaren Tankstelle voll.“*

Beweisen Sie, dass diese Greedy-Strategie eine optimale Lösung liefert, die Anzahl der Tankstopps also minimal ist.

Bemerkung. Die Lösung einer Strategie A, die k Tankstopps benötigt, kann zum Beispiel als Folge von Indizes $a(1) < a(2) < \cdots < a(k)$ mit $a(i) \in \{1, 2, \dots, n\}$ beschrieben werden. Der i-te Tankstopp findet dabei an der Tankstelle $t_{a(i)}$ statt.

Name:	Matrikelnummer:

Name:	Matrikelnummer:

Aufgabe 9

_____/ 12 P.

Im Folgenden definieren wir zwei Entscheidungsprobleme:

- UNABHÄNGIGEMENGE: Eine *unabhängige Menge* eines ungerichteten Graphen $G = (V, E)$ ist eine Teilmenge von Knoten $U \subseteq V$, so dass keine zwei Knoten aus U adjazent sind. Es gilt also $\{u, v\} \notin E$ für alle $u, v \in U$ mit $u \neq v$. Im Entscheidungsproblem UNABHÄNGIGEMENGE ist ein kodiertes Paar $\langle G, k\rangle$ gegeben, wobei $G = (V, E)$ ein ungerichteter Graph ist und $k \in \mathbb{N}$. Es muss entschieden werden, ob G eine unabhängige Menge $U \subseteq V$ der Größe $|U| \geq k$ enthält.

  Abbildung 4a zeigt ein Beispiel für eine Instanz des Problems.

- KEINESCHNITTE: Gegeben ist ein kodiertes Paar $\langle S, l\rangle$, wobei $S = \{I_1, I_2, \dots\}$ eine endliche Menge von Intervallen $I_i \subseteq \mathbb{R}$ ist und $l \in \mathbb{N}$. Es muss entschieden werden, ob es eine Teilmenge $T \subseteq S$ von S der Größe $|T| \geq l$ gibt, so dass alle Intervalle in T paarweise disjunkt sind.

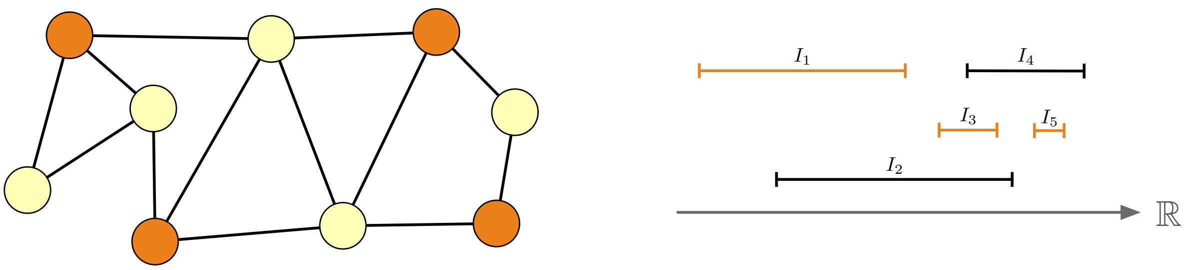
  Abbildung 4b zeigt ein Beispiel für eine Instanz des Problems.

Bemerkung. Sie dürfen zur Lösung der folgenden Aufgaben benutzen, dass UNABHÄNGIGEMENGE NP-schwer ist.

(a) Ist UNABHÄNGIGEMENGE NP-vollständig? Beweisen Sie Ihre Antwort. (4 Punkte)

(b) Beweisen Sie KEINESCHNITTE $\leq_p$ UNABHÄNGIGEMENGE. (6 Punkte)

(c) Was folgt daraus für die Komplexität von KEINESCHNITTE? (2 Punkte)



(a) Ein ungerichteter Graph $G = (V, E)$. Die orange gefärbten Knoten bilden eine unabhängige Menge U der Größe 4.

(b) Eine Menge S bestehend aus fünf reellen Intervallen. Die drei orange gefärbten Intervalle sind paarweise disjunkt.

Abbildung 4

Name:	Matrikelnummer:

Name:	Matrikelnummer:

Name:	Matrikelnummer: