Prüfung aus

# Diskrete und geometrische Algorithmen (Hetzl)

Zuname: ______________________________ Matrikelnummer: ____________________

Vorname: ______________________________

---

Titelseite ab hier bitte freilassen!

TU Wien, 29.1.2019 Arbeitszeit: 100 Minuten

1)

2)

3)

4)

1) (7 P.) Lösen Sie die Rekursionsgleichung $a_n = 5a_{n-1} - 6a_{n-2}$ (für $n \geq 2$) mit den Anfangswerten $a_0 = 1$ und $a_1 = 5$.

2) (7 P.) Wir definieren die Menge von Zeichen

$$A = \{a, b, c\} \cup \{ \ (_i \mid i \in \mathbb{N}\} \cup \{ \ )_i \mid i \in \mathbb{N}\}$$

wobei wir sowohl eine indizierte öffnende Klammer $(_i$ als auch eine indizierte schließende Klammer $)_i$ als jeweils ein einzelnes Zeichen betrachten. Sei $A^*$ die Menge aller endlich langen Zeichenketten die nur aus Zeichen aus $A$ bestehen. Die Menge der *indiziert wohlgeklammerten Ausdrücke* ist die kleinste Menge $W \subseteq A^*$ für die gilt:

1. Die leere Zeichenkette $\varepsilon \in W$.
2. Die Zeichen $a, b, c \in W$.
3. Falls $w_1, w_2 \in W$, dann ist auch $w_1 w_2 \in W$.
4. Falls $w \in W$ und $i \in \mathbb{N}$, dann ist auch $(_i w)_i \in W$.

So ist zum Beispiel $(_3c)_3ab(_3ab(_1c)_1)_3 \in W$ aber $bc(_1b(_2a)_1cc)_2 \notin W$.
Geben Sie einen Algorithmus in Pseudocode an, der eine Zeichenkette $v \in A^*$ als Eingabe erhält und entscheidet, ob $v \in W$ ist. Die Eingabe wird in Form eines Datenfelds übergeben. Die Laufzeit des Algorithmus muss $O(|v|)$ sein wobei $|v|$ die Anzahl der Zeichen in $v$ ist.

3) (7 P.) Sei $G = (V, E)$ ein nicht-leerer endlicher ungerichteter Graph, dann definieren wir $m(G) = \max\{|V_i| \mid 1 \leq i \leq k\}$ wobei $(V_1, E_1), \ldots, (V_k, E_k)$ die Zusammenhangskomponenten von $G$ sind. Geben Sie einen Algorithmus in Pseudocode an, der einen nicht-leeren endlichen ungerichteten Graph $G = (V, E)$ als Eingabe erhält und $m(G)$ als Ausgabe liefert. Der Eingabegraph wird in Form einer Adjazenzliste übergeben wobei $V = \{1, \ldots, n\}$. Die Laufzeit des Algorithmus muss $O(|V| + |E|)$ sein.

4) (3 P.) Geben Sie ein lineares Programm in Standardform an, das drei optimale Lösungen besitzt die nicht auf einer Gerade liegen.

Prüfung aus

# Diskrete und geometrische Algorithmen (Hetzl)

Zuname: ______________________________ Matrikelnummer: ____________________

Vorname: ______________________________

---

Titelseite ab hier bitte freilassen!

TU Wien, 1.3.2019 Arbeitszeit: 100 Minuten

1)

2)

3)

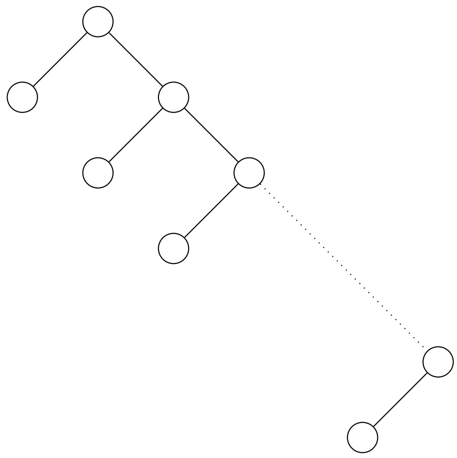
4)

1) (4 P.) Geben Sie asymptotische Lösungen der folgenden Rekursionsgleichungen an:

a) $T(n) = 9T(\frac{n}{3}) + n(n+1)$

b) $T(n) = 3T(\frac{n}{4}) + n \log n$

2) (4 P.) Geben Sie, für jedes gerade $n \in \mathbb{N}$ paarweise unterschiedliche Schlüssel $x_1, \ldots, x_n$ an, so dass ein Suchbaum in Kammform (sh. Abbildung) entsteht wenn, beginnend mit dem leeren Suchbaum, die Schlüssel $x_1, \ldots, x_n$ in dieser Reihenfolge eingefügt werden.



3) (7 P.) Sei

$$A = (a_{i,j})_{\substack{1 \le i \le m \\ 1 \le j \le n}}$$

eine Matrix. Als *Untermatrix von A* bezeichnen wir eine Matrix der Form

$$(a_{i,j})_{\substack{p \le i \le q \\ r \le j \le s}}$$

wobei $p, r \geq 1$, $q \leq m$ und $s \leq n$.

Geben Sie einen Algorithmus in Pseudocode an der eine Matrix $A \in \{0,1\}^{m \times n}$ als Eingabe erhält und der als Ausgabe das größte $k \in \mathbb{N}$ zurückliefert so dass $A$ eine $k \times k$ Untermatrix enthält die nur aus Einsern besteht. Die Laufzeit des Algorithmus muss $O(m \cdot n)$ sein.

*Hinweis: Dynamische Programmierung, betrachten Sie $m_{q,s}$, die maximale Größe einer quadratischen Einser-Matrix die an der Stelle $(q, s)$ endet.*

4) (9 P.) Eine *Strecke* ist ein ungeordnetes Paar $\{p, q\}$ wobei $p, q \in \mathbb{R}^2$ und $p \neq q$. Geben Sie einen Algorithmus in Pseudocode an, der als Eingabe eine endliche Menge von Strecken erhält, bei denen es sich um die Kanten eines konvexen Polygons handelt und der als Ausgabe eine im Uhrzeigersinn sortierte Liste der Knoten des Polygons zurückgibt.

Nehmen Sie der Einfachheit halber an, dass die Eingabe keine zwei Strecken $\{p, q\}, \{q, r\}$ enthält so dass $p$, $q$ und $r$ auf einer Gerade liegen.

Die Laufzeit des Algorithmus muss bei Eingabe von $n$ Strecken $O(n \log n)$ sein.

Prüfung aus

# Diskrete und geometrische Algorithmen (Hetzl)

Zuname: ______________________________ Matrikelnummer: ____________________

Vorname: ______________________________

---

Titelseite ab hier bitte freilassen!

TU Wien, 26.4.2019 Arbeitszeit: 100 Minuten

1)

2)

3)

4)

1) (5 P.) Geben Sie für unendlich viele $n \in \mathbb{N}$ ein Datenfeld $A_n$ der Länge $n$ an, auf dem Einfügesortieren Laufzeit $\Theta(n\sqrt{n})$ hat und bewiesen Sie dass dem so ist.

2) (7 P.)

a) Geben Sie einen Algorithmus in Pseudocode an, der als Eingabe einen Suchbaum $B$ mit $n$ Elementen erhält und als Ausgabe ein Datenfeld liefert, dass die Elemente von $B$ in aufsteigender Reihenfolge enthält. Die Laufzeit des Algorithmus muss $O(n)$ sein.

b) Geben Sie einen Algorithmus in Pseudocode an, der als Eingabe zwei Suchbäume $B_1$ und $B_2$ mit $n_1$ bzw. $n_2$ Elementen enthält und festellt ob deren Schlüsselmengen disjunkt sind. Die Laufzeit des Algorithmus muss $O(n_1 + n_2)$ sein.

3) (7 P.) Ein *gerichteter Zyklus* in einem gerichteten Graphen $G = (V, E)$ ist ein Pfad der Form $v_1, v_2, \ldots, v_{k-1}, v_k = v_1$ mit $k \geq 2$ so dass für alle $i, j \in \{1, \ldots, k-1\}$ mit $i \neq j$ auch $v_i \neq v_j$ ist. Geben Sie einen Algorithmus in Pseudocode an, der als Eingabe einen endlichen gerichteten Graphen $G = (V, E)$ sowie ein $s \in V$ erhält und feststellt ob $s$ in $G$ auf einem gerichteten Zyklus liegt. Der Eingabgraph wird als Adjazenzliste übergeben. Die Laufzeit des Algorithmus muss $O(|V| + |E|)$ sein.

4) (5 P.) Wenden Sie den Simplex-Algorithmus auf das folgende lineare Programm in Standardform an. Falls das Programm eine optimale Lösung hat geben Sie diese an.

$$z = 2x_1 + x_2 \max!$$
$$x_1 + 2x_2 \leq 8$$
$$3x_1 + x_2 \leq 12$$

Prüfung aus

# Diskrete und geometrische Algorithmen (Hetzl)

Zuname: ______________________________ Matrikelnummer: ____________________

Vorname: ______________________________

---

Titelseite ab hier bitte freilassen!

TU Wien, 2.7.2019 Arbeitszeit: 100 Minuten

1)

2)

3)

4)

1) (6 P.) Lösen Sie die Rekursionsgleichung $a_n = 4a_{n-1} - 3a_{n-2}$ (für $n \geq 2$) mit den Anfangswerten $a_0 = 1$ und $a_1 = 2$.

2) (6 P.) Geben Sie einen Algorithmus in Pseudocode an, der als Eingabe ein aufsteigend sortiertes Datenfeld $A$ paarweise verschiedener ganzer Zahlen erhält und feststellt ob ein $i \in \{1, \ldots, n\}$ existiert mit $A[i] = i$. Die Laufzeit des Algorithmus muss $O(\log n)$ sein wobei $n$ die Anzahl der Elemente in $A$ ist.

3) (6 P.) Seien $B_1$ und $B_2$ Suchbäume die die selbe Menge von Einträgen enthalten. Gibt es eine Folge von Links- und Rechtsrotationen mit denen $B_1$ in $B_2$ transformiert werden kann? Beweisen Sie die Korrektheit Ihrer Antwort.

4) (6 P.) Geben Sie einen Algorithmus in Pseudocode an, der als Eingabe ein Datenfeld $A$ paarweise verschiedener Elemente sowie ein $k \in \{1, \ldots, A.\mathit{Länge}\}$ erhält und als Ausgabe eine zufällige $k$-elementige Teilmenge von $A$ als Datenfeld liefert. Die Laufzeit des Algorithmus muss $O(k)$ sein.

Sie dürfen dazu die Existenz einer Prozedur $\textsc{Zufall}(i, j)$ voraussetzen, die für $i, j \in \mathbb{N}$ mit $i \leq j$ in Zeit $O(1)$ eine im Intervall $[i, j] \subseteq \mathbb{N}$ uniform verteilte Zufallszahl liefert.

Prüfung aus

# Diskrete und geometrische Algorithmen (Hetzl)

Zuname: ______________________________ Matrikelnummer: ____________________

Vorname: ______________________________

---

Titelseite ab hier bitte freilassen!

TU Wien, 4.10.2019 Arbeitszeit: 100 Minuten

1)

2)

3)

4)

1) (6 P.) Lösen Sie die Rekursionsgleichung $a_n = 6a_{n-1} - 5a_{n-2}$ (für $n \geq 2$) mit den Anfangswerten $a_0 = 0$ und $a_1 = 1$.

2) (6 P.) Geben Sie einen Algorithmus in Pseudocode an, der als Eingabe ein aufsteigend sortiertes Datenfeld $A$ paarweise verschiedener ganzer Zahlen erhält und feststellt ob ein $i \in \{1, \ldots, n\}$ existiert mit $A[i] = i$. Die Laufzeit des Algorithmus muss $O(\log n)$ sein wobei $n$ die Anzahl der Elemente in $A$ ist.

3) (7 P.) Sei

$$A = (a_{i,j})_{\substack{1 \le i \le m \\ 1 \le j \le n}}$$

eine Matrix. Als *Untermatrix von $A$* bezeichnen wir eine Matrix der Form

$$(a_{i,j})_{\substack{p \le i \le q \\ r \le j \le s}}$$

wobei $p, r \ge 1$, $q \le m$ und $s \le n$.

Geben Sie einen Algorithmus in Pseudocode an der eine Matrix $A \in \{0,1\}^{m \times n}$ als Eingabe erhält und der als Ausgabe das größte $k \in \mathbb{N}$ zurückliefert so dass $A$ eine $k \times k$ Untermatrix enthält die nur aus Einsern besteht. Die Laufzeit des Algorithmus muss $O(m \cdot n)$ sein.

*Hinweis: Dynamische Programmierung, betrachten Sie $m_{q,s}$, die maximale Größe einer quadratischen Einser-Matrix die an der Stelle $(q,s)$ endet.*

4) (5 P.) Wenden Sie den Simplex-Algorithmus auf das folgende lineare Programm in Standardform an. Falls das Programm eine optimale Lösung hat geben Sie diese an.

$$z = x_1 + 2x_2 \text{ max!}$$
$$2x_1 + x_2 \leq 8$$
$$x_1 + 3x_2 \leq 12$$