Institut für Informatik
Prof. Dr. Heiko Röglin
Tobias Brunsch

Algorithmen und Berechnungskomplexität I
Wintersemester 2013/14

Probeklausur

Bitte beachten Sie folgende Hinweise: Die Probeklausur hat keinerlei Einfluss auf die Prüfungszulassung oder die endgültige Note. Sie dient lediglich dazu, einen Eindruck von den Klausuraufgaben zu vermitteln.

Aufgabe 1

(a) Gegeben sei die Rekursionsgleichung

$$T(n) = 8 \cdot T(n/2) + 7n^3$$

für $n \geq 2$ und $T(1) = \Theta(1)$. Geben Sie eine möglichst einfache Funktion g an, für die $T(n) = \Theta(g(n))$ gilt. Begründen Sie, warum Ihre Wahl korrekt ist.

(b) Geben Sie eine möglichst einfache Funktion f mit $\sum_{i=1}^{n} \log_2(i) = \Theta(f(n))$ an.

(c) Wir betrachten ein Feld A mit Einträgen aus $\{0, \ldots, 255\}$. Prof. G. Witzt behauptet, dass er einen Algorithmus entwickelt hat, der Felder dieses Typs in Zeit $O(n \log \log n)$ sortieren kann. Ist das möglich? Begründen Sie Ihre Antwort!

(d) Welche Gestalt müssen die Ableitungsregeln einer kontextfreien Grammatik $G = (\Sigma, V, S, P)$ besitzen, damit G in Chomsky-Normalform ist?

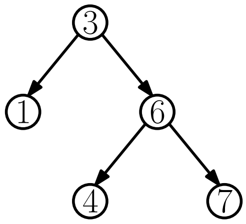
(e) Wie sieht ein lineares Programm in kanonischer Form aus?

(f) Geben Sie die Worst-Case-Laufzeit des Algorithmus von Kruskal in Θ-Notation an. Wie ändert sich diese Laufzeit, wenn man eine *Union-Find*-Datenstruktur verwendet, die für jede Folge von u Union-Befehlen und f Find-Befehlen Zeit $\Theta(u + f)$ benötigt? Begründen Sie Ihre Antwort.

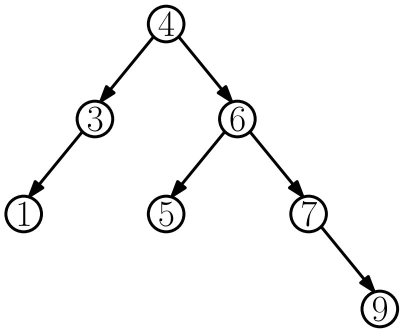
Aufgabe 2

(a) Definieren Sie, was ein binärer Suchbaum ist.

(b) Fügen Sie den Schlüssel 9 in den unten abgebildeten AVL-Baum ein.



(c) Löschen Sie den Schlüssel 1 aus dem unten abgebildeten AVL-Baum.



(d) Es bezeichne $A(h)$ die geringste Anzahl an Knoten, die ein AVL-Baum der Höhe h haben kann. Geben Sie eine Rekursionsgleichung für $A(h)$ an.

(e) Wir betrachten geschlossenes Hashing mit quadratischem Sondieren und einer Hashtabelle der Größe $m = 11$. Fügen Sie die Schlüssel

$$40, 14, 23, 6, 12, 18, 30, 24, 8, 36$$

in der angegebenen Reihenfolge gemäß der Hashfunktion $h(x) := x \bmod m$ in die Hashtabelle ein. Es genügt, das Endergebnis aller Einfügeoperationen anzugeben.

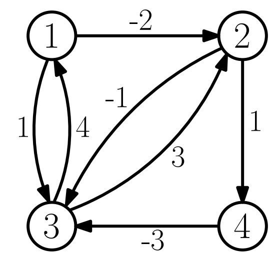
Hashtabelle *T*:

Aufgabe 3

(a) Der Floyd-Warshall-Algorithmus erzeugt auf dem unten abgebildeten Graphen eine Sequenz von Matrizen, die wir mit $D^{(0)}, D^{(1)}, \ldots, D^{(4)}$ bezeichnen. Die ersten drei dieser Matrizen sind gegeben durch

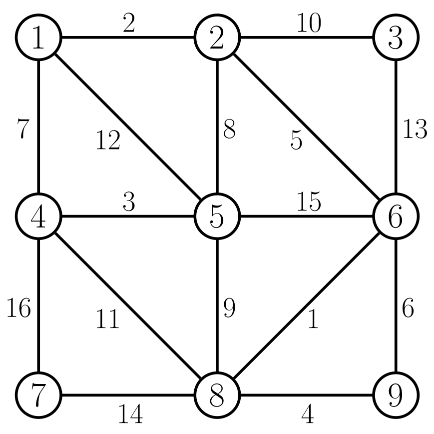
$$D^{(0)} = \begin{bmatrix} 0 & -2 & 1 & \infty \\ \infty & 0 & -1 & 1 \\ 4 & 3 & 0 & \infty \\ \infty & \infty & -3 & 0 \end{bmatrix}, \quad D^{(1)} = \begin{bmatrix} 0 & -2 & 1 & \infty \\ \infty & 0 & -1 & 1 \\ 4 & 2 & 0 & \infty \\ \infty & \infty & -3 & 0 \end{bmatrix} \quad \text{und} \quad D^{(2)} = \begin{bmatrix} 0 & -2 & -3 & -1 \\ \infty & 0 & -1 & 1 \\ 4 & 2 & 0 & 3 \\ \infty & \infty & -3 & 0 \end{bmatrix}.$$

Geben Sie die Matrix $D^{(3)}$ an.



$$D^{(3)} = \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix}$$

(b) Bestimmen Sie mithilfe des Algorithmus von Kruskal einen minimalen Spannbaum auf dem unten abgebildeten Graphen.



(c) Geben Sie die Worst-Case-Laufzeit des Algorithmus von Ford und Fulkerson aus der Vorlesung an und beweisen Sie diese.

(d) Sei $G = (V, E)$ ein zusammenhängender ungerichteter Graph mit einer Kantengewichtung $w\colon E \to \mathbb{R}_{>0}$ derart, dass keine Kantengewichte mehrfach vorkommen. Die Kanten von G seien so mit $e_1, \ldots, e_m$ durchnummeriert, dass $w(e_1) < w(e_2) < \ldots < w(e_m)$ gilt. Des Weiteren sei T ein minimaler Spannbaum von G bezüglich der Kantengewichtung w. Geben Sie für $i = 1$, für $i = 2$ und für $i = 3$ eine sowohl notwendige als auch hinreichende Bedingung an, unter der $e_i \in T$ gilt. Begründen Sie Ihre Antwort.

Hinweis: Da der Algorithmus von Kruskal nur einen möglichen minimalen Spannbaum liefert, können Sie nicht ohne Weiteres die Vorgehensweise des Algorithmus von Kruskal zur Argumentation heranziehen.

(e) Gegeben seien ein gerichteter Graph $G = (V, E)$ in Adjazenzlistendarstellung mit einer Kantengewichtung $w\colon E \to \{1, 2\}$ und ein Startknoten $s \in V$. Geben Sie einen Algorithmus an, der in Zeit $O(|V| + |E|)$ den Abstand von s zu jedem Knoten $v \in V$ berechnet. Begründen Sie, warum Ihr Algorithmus korrekt ist.

Hinweis: Überlegen Sie sich zunächst, wie Sie vorgehen würden, wenn alle Kantengewichte gleich 1 wären.

Aufgabe 4

(a) Geben Sie ein fiktives Münzsystem und einen Wechselgeldbetrag für das Wechselgeldproblem an, für die der Greedy-Algorithmus aus der Vorlesung keine optimale Lösung berechnet.

Hinweis: Das Münzsystem muss aus positiven ganzzahligen Werten bestehen und eine Münze mit dem Wert 1 enthalten.

(b) Gegeben sei ein gerichteter Graph $G = (V, E)$ mit einer Kantengewichtung $w\colon E \to \mathbb{R}$. Die Knoten des Graphen seien mit $1, \ldots, n$ durchnummeriert und für alle Kanten $(i, j) \in E$ gelte $i < j$. Geben Sie ein dynamisches Programm an, das in Zeit $O(|V| + |E|)$ die Abstände $\delta(1, v)$ für alle Knoten $v \in V$ berechnet. Begründen Sie, warum Ihr Algorithmus korrekt ist.

Hinweis: Sie können annehmen, dass für jeden Knoten v eine Liste der eingehenden Kanten $(u, v) \in E$ vorliegt.

(c) Wir betrachten das Rucksackproblem mit 4 Objekten, deren Gewichte und Nutzenwerte in der untenstehenden Tabelle aufgelistet sind, und einem Rucksack mit Kapazität 24.

i	1	2	3	4
w_i	10	7	8	6
p_i	3	2	3	2

Bestimmen Sie mithilfe des dynamischen Programmes aus der Vorlesung oder aus der Übung eine optimale Lösung für diese Instanz. Geben Sie auch die Tabelle an, die in dem dynamischen Programm berechnet wird.

Hinweis: Füllen Sie nur so viele Spalten der Tabelle aus wie nötig.

Aufgabe 5

(a) Wandeln Sie das Optimierungsproblem

$$\begin{aligned} &\min \max \{c_1 \cdot x, \ldots, c_k \cdot x\} \\ &Ax \le b \\ &x \ge 0 \end{aligned}$$

in ein äquivalentes lineares Programm um.

(b) Ein *Integer Linear Program* (ILP) ist ein lineares Programm, in dem alle Variablen nur ganzzahlige Werte annehmen dürfen. Die Koeffizienten in der Zielfunktion und in den Nebenbedingungen können weiterhin reell sein.

Formulieren Sie das Wechselgeldproblem mit einem Münzsystem $M = \{c_1, \ldots, c_k\}$ aus der Vorlesung als ILP.

(c) In einer Fabrik werden zwei Chemikalien A und B hergestellt. Dafür gibt es zwei verschiedene Prozesse 1 und 2, die beide eine Grundsubstanz X benötigen. Prozess 1 dauert 2 Stunden, benötigt 100 ml der Substanz X und liefert 2 ml von Chemikalie A sowie 1 ml von Chemikalie B. Prozess 2 dauert 3 Stunden, benötigt 200 ml der Substanz X und liefert 3 ml von Chemikalie A sowie 2 ml von Chemikalie B. Insgesamt stehen 60 Arbeitsstunden und 4 l der Substanz X zur Verfügung. Der Gewinn beim Verkauf von Chemikalie A liegt bei 16 Euro pro Milliliter und bei Chemikalie B bei 14 Euro pro Milliliter. Ziel ist es, den Gewinn zu maximieren.

Formulieren Sie dieses Problem als lineares Programm (LP). Beschreiben Sie die Bedeutung der Variablen, die Sie in Ihrem LP benutzen.

Aufgabe 6

(a) Geben Sie das Pumping Lemma für kontextfreie Sprachen an.

(b) Bringen Sie die Grammatik $G = (\{a, b, c, d\}, \{S, A, B, C, D\}, S, P)$ mit den Ableitungen

$$S \to ABC, d \qquad A \to BC, D, a \qquad B \to AC, DC, c \qquad C \to a, b, c, d \qquad D \to AB, cd$$

auf Chomsky-Normalform.

(c) Ist die Sprache $L = \{a^n b^n c^n : n \geq 1\}$ über dem Alphabet $\Sigma = \{a, b, c\}$ kontextfrei? Beweisen Sie Ihre Vermutung.