Eidgenössische Technische Hochschule Zürich
Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Swiss Federal Institute of Technology Zurich

Institut für Theoretische Informatik
Prof. Dr. Angelika Steger

8. Februar 2018

Algorithmen und Komplexität (D-MATH)
Klausur Winter 2018

Kandidat/in:

Name: ....

Vorname: ....

Stud.-Nr.: ....

Ich bezeuge mit meiner Unterschrift, dass ich die Prüfung unter regulären Bedingungen ablegen konnte und dass ich die unten stehenden allgemeinen Bemerkungen gelesen und verstanden habe.

Unterschrift: ....

Allgemeine Bemerkungen und Hinweise:

- Diese Prüfung besteht neben diesem doppelseitigen Deckblatt aus 3 beidseitig bedruckten Aufgabenblättern mit insgesamt 6 Aufgaben.
- Als einziges Hilfsmittel sind 10 beidseitig handschriftlich beschriebene A4-Blätter erlaubt.
- Falls Sie während der Prüfung in irgendeiner Weise gestört oder beeinträchtigt werden, melden Sie dies sofort der Aufsichtsperson. Spätere Klagen werden nicht akzeptiert.
- **Schreiben Sie nicht mit Bleistift! Abgaben, die mit Bleistift geschrieben sind, werden nicht bewertet. Das Verwenden von roter und grüner Farbe sowie von Tippex ist ebenfalls nicht erlaubt.**
- Alle Mobiltelefone und sonstigen elektronischen Geräte müssen vollständig ausgeschaltet sein und im Gepäck verstaut werden.
- **Alle Antworten müssen für den Korrektor verständlich begründet werden. Schreiben Sie die wesentlichen Lösungsgedanken in klaren Sätzen oder Stichworten hin. Unverständliche oder nicht begründete Antworten werden nicht bewertet.**
- Pro Aufgabe ist höchstens eine gültige Version eines Lösungsversuchs zulässig. Streichen Sie ungültige Lösungsversuche klar durch.
- Sie dürfen alle Aufgaben in beliebiger Reihenfolge lösen. Konzentrieren Sie sich jeweils auf eine Aufgabe, aber teilen Sie sich Ihre Zeit ein.
- Abschreiben und sonstige Versuche des Betrugs führen zum sofortigem Ausschluss von der Prüfung und können rechtliche Folgen haben.
- Die Prüfung dauert zwei Stunden. Falls Sie vorzeitig fertig werden sollten, melden Sie sich durch Handaufhalten bei einer der Aufsichtspersonen und verlassen Sie still den Raum. **In den letzten 20 Minuten der Prüfung kann der Raum nicht mehr verlassen werden.**
- **Vergessen Sie nicht, dieses Deckblatt zu unterschreiben, und beschriften Sie jedes abgegebene Blatt mit Ihrem Namen. Die Aufgabenblätter sind mit abzugeben.**

Viel Erfolg!

	Erreichte Punktzahl (maximal)	Visum
1	(10)	
2	(8)	
3	(12)	
4	(10)	
5	(10)	
6	(10)	
Σ	(60)	

Aufgabe 1 (total 10 Punkte)

Welche der folgenden Aussagen treffen zu, und welche nicht? Für jede richtige Antwort erhalten Sie 1 Punkt (keine Minuspunkte). Die Antworten müssen Sie *nicht* begründen.

(a) Ist ein Graph $G = (V, E)$ in Form von Adjazenzlisten abgespeichert, so lässt sich für gegebene $u, v \in V$ in Zeit $O(1)$ überprüfen ob $\{u, v\} \in E$.

wahr ☐ falsch ☐

(b) Aus $f(n) = O(2^n)$ und $g(n) = O(n^3)$ folgt $g(n) = O(f(n))$.

wahr ☐ falsch ☐

(c) $(\log(n))^{\log n} = \Omega(n^5)$.

wahr ☐ falsch ☐

(d) Sei $G = (V, E)$ ein Graph, so dass jeder Knoten Grad höchstens 10 hat. Dann kann in Zeit $O(|V|)$ getestet werden, ob G zusammenhängend ist.

wahr ☐ falsch ☐

(e) Es gibt einen $(2, 3)$-Baum T der Höhe 2 und einen $(2, 3)$-Baum T' der Höhe 3, sodass T mehr Schlüssel enthält als T'.

wahr ☐ falsch ☐

(f) Mithilfe der Union-Find Datastruktur lässt sich die Laufzeit des Algorithmus von Prim auf $\mathcal{O}(n \log n + m)$ verbessern.

wahr ☐ falsch ☐

(g) Gegeben sei ein Fibonacci-Heap F mit n Schlüsseln. Nun wird die Extract-Min Operation ausgeführt. Danach befinden sich $O(\log n)$ Schlüssel in der Wurzelliste.

wahr ☐ falsch ☐

(h) Falls $\mathcal{P} \neq \mathcal{NP}$, dann gibt es keinen polynomiellen Algorithmus für 3-Sat.

wahr ☐ falsch ☐

(i) Die Formel $(x_1 \vee x_2) \wedge \overline{x_2} \wedge x_3 \wedge (\overline{x_1} \vee \overline{x_3})$ ist erfüllbar.

wahr ☐ falsch ☐

(j) Falls eine Sprache L berechenbar ist, so gibt es ein RAM-Programm A mit der Eigenschaft $A(x) = L(x)$ für alle $x \in \{0, 1\}^*$.

wahr ☐ falsch ☐

(10 · 1 = 10 Punkte)

Aufgabe 2

(total 8 Punkte)

Welche der folgenden Aussagen treffen zu, und welche nicht? Begründen Sie jeweils Ihre Antwort. Pro richtige Begründung gibt es jeweils 2 Punkte.

(a) Besitzt ein Netzwerk $N = (V, E, \ell)$ genau einen minimalen Spannbaum, so existiert für jede nicht-leere Teilmenge $W \subsetneq V$ genau eine Kante zwischen W und $V \setminus W$ mit minimalem Gewicht.

(b) Wenn Breitensuche und Tiefensuche auf C_n (Kreis bestehend aus n Knoten) ausgeführt werden, dann stimmen die beiden erhaltenen BFS- und DFS-Spannbäume überein.

(c) Sei $G = (V, E)$ ein Graph mit n Knoten und der Eigenschaft, dass jeder Knoten v in einem Dreieck enthalten ist, das heisst für alle $v \in V$ existieren $u, w \in V$ so dass $\{u, v\}, \{v, w\}, \{w, u\} \in E$. Dann gilt $|E| \geq |V|$.

(d) $\textsc{Knapsack} \leq_p \textsc{ErfüllbarerSchaltkreis}$

($4 \cdot 2 = 8$ Punkte)

Aufgabe 3

(total 12 Punkte)

(a) Finden Sie einen Algorithmus, der in Laufzeit $O(|V|^3)$ entscheidet, ob es für einen gegebenen Graphen $G = (V, E)$ eine zulässige 3-Färbung mit den Farben rot, blau, grün gibt, so dass höchstens ein Knoten Farbe blau erhält. Zeigen Sie kurz Korrektheit und Laufzeit Ihrer Idee.

(6 Punkte)

(b) Entwerfen Sie einen Algorithmus der als Eingabe ein Array $A = [a_1, \dots, a_n]$, bestehend aus n Zahlen, und eine weitere Zahl $1 \leq k \leq n$ bekommt, und dann diejenigen k Zahlen a_i ausgibt, die am nächsten beim Median von A liegen. Die Laufzeit soll $O(n)$ betragen.

Hinweis: Sie dürfen annehmen, dass n ungerade ist und somit der Median eindeutig definiert ist.

(6 Punkte)

Aufgabe 4 (total 10 Punkte)

In der Ferienserie haben wir gesehen, dass das Problem VertexCover $\mathcal{NP}$-vollständig ist:

VertexCover

Eingabe: Ein Graph $G = (V, E)$ und eine Zahl $k \in \mathbb{N}$

Frage: Gibt es eine Teilmenge S von höchstens k Knoten, so dass jede Kante $e \in E$ mindestens einen Knoten in S besitzt?

Wir betrachten nun das folgende Sammler-Problem:

FootballCardCollector

Eingabe: Eine Zahl $k \in \mathbb{N}$, eine Menge von n Fussballern, sowie Päckchen $P_1, \ldots, P_m$, die jeweils einige Karten von Fussballern enthalten.

Frage: Gibt es eine Möglichkeit, alle Fussballer mindestens einmal zu erhalten, indem man höchstens k Päckchen kauft?

Wir nehmen an, dass der Inhalt von allen Päckchen P_i bekannt ist. Es ist erlaubt, dass nicht alle Päckchen gleich viele Karten enthalten.

Beweisen Sie, dass das Problem FootballCardCollector $\mathcal{NP}$-vollständig ist.

(10 Punkte)

Aufgabe 5

(total 10 Punkte)

Annika lernt n Tage für eine Mathematik-Prüfung. An jedem Tag i entscheidet sie am Morgen, ob sie viel, wenig, oder gar nicht lernt. Falls Annika an einem Tag i viel lernt, so ist sie am nächsten Tag $i+1$ müde und muss dann Pause machen.

Gegeben seien zwei Arrays a und b, die jeweils n natürliche Zahlen enthalten. Wenn Annika am Tag i nicht lernt, hat sie Profit $p_i = 0$. Falls sie am Tag i ein wenig lernt, erreicht sie Profit $p_i = a_i$. Lernt Annika am Tag i viel, so gewinnt sie schliesslich Profit $p_i = b_i$. Der totale Profit ist $P = \sum_{i=1}^{n} p_i$.

Verwenden Sie das Prinzip der Dynamischen Programmierung, um einen Algorithmus zu entwerfen, der in Laufzeit $\mathcal{O}(n)$ entscheidet, wieviel Annika an welchem Tag lernen muss, um den Profit P zu maximieren.

(10 Punkte)

Aufgabe 6

(total 10 Punkte)

Gegeben sei ein gerichteter Graph $G = (V, A)$ ohne Self-Loops, der als Adjazenzmatrix abgespeichert ist. Für zwei Knoten u, v ist es erlaubt, dass sowohl (u, v) als auch (v, u) im Graph G enthalten sind. Ein Knoten v heisst *berühmt*, wenn für alle Knoten $u \in V \setminus \{v\}$ gilt

$$(u, v) \in A \quad \text{und} \quad (v, u) \notin A.$$

Offensichtlich existiert höchstens ein berühmter Knoten in G. Finden Sie einen Algorithmus, der in Laufzeit $\mathcal{O}(n)$ entscheidet, ob es einen berühmten Knoten v gibt, und v ausgibt, falls er existiert. Zeigen Sie kurz, dass Ihr Algorithmus korrekt ist und lineare Laufzeit aufweist. *(10 Punkte)*