

Algorithmen und Komplexität (D-MATH) Klausur Frühling 2012

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 beidseitig bedruckten Aufgabenblättern mit insgesamt 5 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.**
- Alle Mobiltelefone müssen vollständig ausgeschaltet sein.
- **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. Das Aufgabenblatt ist mit abzugeben.**

Viel Erfolg!

	Erreichte Punktzahl (maximal)	Visum
1	(15)	
2	(20)	
3	(25)	
4	(30)	
5	(30)	
Σ	(120)	

Aufgabe 1

(total 15 Punkte)

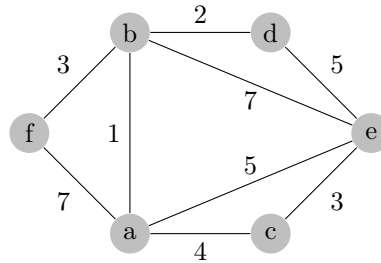
Welche der folgenden Aussagen trifft zu? Begründen Sie Ihre Antwort.

- (a) Quicksort hat eine Worst-Case Laufzeit von $\Omega(n \log n)$.
- (b) Für jeden Graphen G gilt, dass entweder der Graph selbst oder sein Komplement \overline{G} bipartit ist.
- (c) Es gibt einen auf Vergleichen basierenden Suchbaum, der die Operationen INSERT und EXTRACTMIN in $O(1)$ amortisierter Zeit ausführt.
- (d) Aus $f(n) = \Omega(n^3)$ und $f(n) = \mathcal{O}(n^4)$ folgt $(f(n))^2 = \Omega(n^5)$.
- (e) Falls $P \neq NP$ gilt, dann gibt es einen Algorithmus, welcher in $O(|\phi|2^{|\phi|})$ Operationen entscheidet ob eine Formel ϕ erfüllbar ist.

(15 Punkte)

Aufgabe 2

(total 20 Punkte)

Betrachten Sie den folgenden Graphen G :

- (a) Finden Sie einen kürzesten Pfad von f nach e in G . Beschreiben Sie Ihr Vorgehen. (5 Punkte)
- (b) Untersuchen Sie folgenden rekursiven MST-Algorithmus auf Korrektheit. Der Algorithmus unterteilt die Knoten in zwei beliebige Mengen $V = S_1 \cup S_2$ (so dass $G[S_1]$ und $G[S_2]$ verbunden sind) und berechnet rekursiv einen minimalen Spannbaum T_1 von $G[S_1]$ und T_2 von $G[S_2]$. Ausgabe ist $T_1 \cup T_2 \cup \{e\}$ wobei e die kleinste Kante zwischen S_1 und S_2 ist. Für einen Graphen mit nur einem Knoten gibt der Algorithmus die leere Menge zurück. (15 Punkte)

Aufgabe 3

(total 25 Punkte)

Sie wissen aus der Vorlesung, dass 3-SAT NP-vollständig ist. Zeigen Sie, dass das folgende Problem ALMOST 3-SAT NP-vollständig ist.

ALMOST 3-SAT

Gegeben: Variablen x_1, \dots, x_n und eine Boolesche Formel F in konjunktiver Normalform, in der jede Klausel höchstens k Literale enthält.

Frage: Gibt es eine Belegung für F , so dass genau eine Klausel nicht erfüllt ist und alle anderen erfüllt sind.

(25 Punkte)

Aufgabe 4

(total 30 Punkte)

Auf einem Graphen $G = (V, E)$ mit Gewichtsfunktion $w : V \rightarrow \mathbb{R}$ betrachten wir das folgende Bankräuber Problem. Sei $P \subseteq V$ die Menge der Polizeistationen und $B \subseteq V$ die Menge der Banken. Ein Knoten $v \in V$ gilt genau dann als *sicher*, wenn seine Distanz zur nächsten Bank kürzer ist als seine Distanz zur nächsten Polizeistation.

- (a) Beschreiben Sie einen effizienten Algorithmus welcher alle sicheren Knoten berechnet. (15 Punkte)
- (b) Wir betrachten nun folgende Modifikation des Problems. Es gibt nur eine Polizeistation und nur eine Bank. Allerdings kann sich der Räuber nun dreimal so schnell über die Kanten bewegen wie die Polizei. Ein Knoten $v \in V$ ist nur dann sicher, wenn es einen Pfad von v zur Bank gibt, so dass der Räuber jeden Knoten auf dem Pfad vor der Polizei erreichen kann. Berechnen Sie unter diesen Umständen alle sicheren Knoten. (15 Punkte)

Aufgabe 5

(total 30 Punkte)

Gegeben ist eine Reihe x_1, x_2, \dots, x_n von n Münzen wobei x_i den Wert der i -ten Münze beschreibt und n eine gerade Zahl ist. Die Spieler A und B spielen nun folgendes Spiel. Beginnend mit Spieler A , darf abwechslungsweise ein Spieler die Münze am linken oder am rechten Ende der Reihe entfernen. Dabei möchte jeder Spieler am Ende einen möglichst hohen Betrag an Münzen in seinem Besitz haben. Entwerfen Sie einen Algorithmus, welcher mittels dynamischer Programmierung berechnet, welchen Betrag sich Spieler A sichern kann, wenn beide Spieler optimal spielen.

(30 Punkte)