Institut für Theoretische Informatik
Prof. Dr. Angelika Steger und Dr. Johannes Lengler

24. Januar 2019

Algorithmen und Komplexität (D-MATH)
Klausur Winter 2019

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 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. 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	(20)	
2	(12)	
3	(10)	
4	(18)	
5	(12)	
Σ	(72)	

Aufgabe 1

(total 20 Punkte)

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

(a) $2^{2n} = \mathcal{O}(3^n)$.

wahr ☐ falsch ☐

(b) Gegeben eine Netzwerk $N = (V, E, \ell)$. Angenommen e ist die einzige Kante mit minimalem Kantengewicht in E, das heisst $\ell(e) < \ell(e')$ für alle Kanten $e' \neq e$. Dann ist e in jedem minimalen Spannbaum enthalten.

wahr ☐ falsch ☐

(c) Ein $(3, 5)$-Baum mit n Schlüsseln hat Höhe $\mathcal{O}(\log n)$.

wahr ☐ falsch ☐

(d) Wird bei einem (a, b)-Baum eine DELETE-Operation ausgeführt, so ist immer eine Rebalancierung notwendig.

wahr ☐ falsch ☐

(e) Fibonacci-Heaps eignen sich gut, um die asymptotische Laufzeit des Algorithmus von Prim auf $\mathcal{O}(|V| \log |V| + |E|)$ zu reduzieren.

wahr ☐ falsch ☐

(f) Union-Find Strukturen eignen sich gut, um die assymptotische Laufzeit des Algorithmus von Dijkstra auf $\mathcal{O}(|V| \log |V| + |E|)$ zu reduzieren.

wahr ☐ falsch ☐

(g) Ein Fibonacci-Heap mit n Schlüsseln hat maximal Höhe $\mathcal{O}(\log n)$.

wahr ☐ falsch ☐

(h) Die Boolesche Formel $F = (x_1 \vee x_2) \wedge (\bar{x}_1 \vee x_3) \wedge (\bar{x}_2 \vee \bar{x}_3) \wedge (x_2 \vee x_3) \wedge (\bar{x}_1 \vee x_2)$ ist erfüllbar.

wahr ☐ falsch ☐

(i) Sei L eine $\mathcal{NP}$-vollständige Sprache. Es gilt für alle Sprachen L', dass $L' \leq_P L$.

wahr ☐ falsch ☐

(j) Jeder Graph in dem alle Knoten Grad höchstens 2 haben ist bipartit.

wahr ☐ falsch ☐

(k) Sei $G = (V, E)$ ein Graph. Sei $\mathcal{U}$ die Menge, welche aus den Teilmengen $A \subseteq E$ mit der Eigenschaft, dass der Graph (V, A) keinen Kreis enthält, besteht. Dann ist $\mathcal{M} = (E, \mathcal{U})$, ein Matroid.

wahr ☐ falsch ☐

(l) Sei $N = (V, E, \ell)$ ein Netzwerk mit positiven Kantengewichten und sei T ein minimaler Spannbaum von N. Für jedes Paar von Knoten $u, v \in V$ gilt, dass der eindeutige $u - v$ Pfad in T ein kürzester $u - v$ Pfad in N ist.

wahr ☐ falsch ☐

(m) Es ist bekannt, dass das Problem 2-COL in $\mathcal{P}$ liegt.

wahr ☐ falsch ☐

(n) Jede Boolesche Funktion $f : \{0,1\}^n \to \{0,1\}$ kann als Boolsche Formel in Konjunktiver Normalform dargestellt werden.

wahr ☐ falsch ☐

(o) Es gibt genau 2^n Boolesche Funktionen $f : \{0,1\}^n \to \{0,1\}$.

wahr ☐ falsch ☐

(p) Es gibt mehr unberechenbare als berechenbare Funktionen $f : \{0,1\}^* \to \{0,1\}$.

wahr ☐ falsch ☐

(q) Sei L eine Sprache mit 3-SAT $\leq_p L$. Dann ist L $\mathcal{NP}$-vollständig.

wahr ☐ falsch ☐

(r) Seien L und L' zwei Sprachen. Aus $L \leq_p L'$ folgt $L' \leq_p L$.

wahr ☐ falsch ☐

(s) Wenn ein $\mathcal{NP}$-vollständiges Problem in polynomieller Zeit gelöst werden kann, so können alle Probleme in $\mathcal{NP}$ in polynomieller Zeit gelöst werden.

wahr ☐ falsch ☐

(t) Es sei $f : \{1, 2, \ldots, n\} \to \mathbb{N}$ eine monoton steigende Funktion, welche in konstanter Zeit berechnet werden kann. Für jedes x ist in $\mathcal{O}(\log n)$ Zeit prüfbar, ob es ein $k \in \{1, 2, \ldots, n\}$ gibt mit $f(k) = x$.

wahr ☐ falsch ☐

($20 \cdot 1 = 20$ Punkte)

Aufgabe 2

(total 12 Punkte)

Sei $A = (a_{i,j})_{1 \le i,j \le n}$ eine binäre $n \times n$ Matrix, das heisst, A enthält nur Nullen und Einsen. Gesucht ist die Grösse der grössten quadratischen Teilmatrix, welche nur Einsen enthält. Wir nennen eine solche Teilmatrix mit rechter unterer Ecke $a_{i,j}$ und linker oberer Ecke $a_{i-m,j-m}$, welche nur aus Einsen besteht (das heisst, dass $a_{k,\ell} = 1$ für alle $i - m \le k \le i$ und $j - m \le \ell \le j$), eine *volle quadratische Teilmatrix* der Grösse $m + 1$.

a) Sei $f(i,j)$ die Grösse der grössten vollen quadratischen Teilmatrix, deren rechte untere Ecke $a_{i,j}$ ist. Das Programm ORAKEL berechnet den Wert $f(i,j)$ in konstanter Zeit, wenn es als Input die Werte $a_{i,j}$, $f(i-1,j)$, $f(j,i-1)$ und $f(i-1,j-1)$ erhält. Erstellen Sie ein dynamisches Programm mit Laufzeit $\mathcal{O}(n^2)$, welches mithilfe des Programms ORAKEL die Grösse der grössten vollen quadratischen Teilmatrix in A bestimmt. *(8 Punkte)*

b) Finden Sie eine Rekursionsformel mit der $f(i,j)$ in konstanter Zeit aus den Werten $a_{i,j}$, $f(i-1,j)$, $f(j,i-1)$ und $f(i-1,j-1)$ berechnet werden kann und begründen Sie die Korrektheit der Formel. *(4 Punkte)*

Aufgabe 3

(total 10 Punkte)

Zeigen Sie, dass das folgende Entscheidungsproblem $\mathcal{NP}$-vollständig ist.

Grad-Beschränkter-Spannbaum

Input: Graph $G = (V, E)$ und eine natürliche Zahl k.

Frage: Gibt es einen Spannbaum T von G, sodass alle Knoten maximal Grad k in T haben? Ein Knoten hat Grad k in T, wenn er im Graphen T adjazent zu k Knoten ist.

Sie dürfen dabei annehmen, dass folgende Entscheidungsprobleme $\mathcal{NP}$-vollständig sind.

SAT

Eingabe: Boolesche Formel F mit n Variabeln $x_1, \ldots, x_n$.

Frage: Besitzt F eine erfüllende Belegung?

Hamiltonpfad

Eingabe: Graph $G = (V, E)$.

Frage: Existiert ein Hamiltonpfad in G? Ein Hamiltonpfad ist ein Pfad, der jeden Knoten genau einmal besucht.

3-Col

Eingabe: Graph $G = (V, E)$.

Frage: Ist G 3-färbbar?

(10 Punkte)

Aufgabe 4

(total 18 Punkte)

Ein Young-Tableau ist eine $m \times n$ Matrix, in der die Einträge in jeder Reihe (von links nach rechts) und in jeder Spalte (von oben nach unten) monoton wachsend sind. Einträge in einem Young-Tableau können auch leer sein, wobei die leeren Einträge mit ∞ bezeichnet werden (wir nehmen dabei an dass eine Folge wie $1, 6, \infty, \infty$ monoton wachsend ist). Ein Beispiel für ein 4×5 Young-Tableau ist angegeben.

3	4	8	10	16
5	7	12	17	∞
6	13	18	19	∞
11	21	25	∞	∞

Im folgenden sei A ein $m \times n$ Young-Tableau.

a) Entwerfen Sie die Operation Extract-Min(A). Diese Operation soll A so modifizieren, dass ein Young-Tableau entsteht, welches das kleinste Element von A nicht enthält, ansonsten aber genau dieselben Elemente wie A enthält. Die Operation Extract-Min(A) soll Laufzeit $\mathcal{O}(n + m)$ haben. *(5 Punkte)*

b) Entwerfen Sie die Operation Insert(A, x), wobei wir annehmen, dass A mindestens einen leeren Eintrag enthält. Diese Operation soll A so modifizieren, dass ein Young-Tableau entsteht, welches genau dieselben Elemente wie A und zusätzlich das Element x enthält. Die Operation Insert(A, x) soll Laufzeit $\mathcal{O}(n + m)$ haben. *(5 Punkte)*

c) Benutzen Sie obige zwei Operationen, um n^2 Zahlen in Zeit $\mathcal{O}(n^3)$ zu sortieren. Dabei dürfen keine anderen Sortieralgorithmen verwendet werden. *(2 Punkte)*

d) Entwerfen Sie die Operation Find(A, x), welche in Zeit $\mathcal{O}(n + m)$ feststellt, ob x in A enthalten ist. *(6 Punkte)*

Aufgabe 5

(total 12 Punkte)

Gegeben sei ein Graph $G = (V, E)$, welcher mit Adjazenzlisten abgespeichert ist. Entwerfen Sie einen Algorithmus welcher in Zeit $\mathcal{O}(|V| + \sum_{v \in V} \deg(v)^2)$ die Anzahl Dreiecke in G bestimmt.

(12 Punkte)

Hinweis: Für einen Algorithmus und Laufzeitanalyse, welche Laufzeit $\mathcal{O}(|V| \cdot (|V| + |E|))$ zeigt, können 8 von 12 Punkte erreicht werden.