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, Prof. Dr. Thomas Holenstein, Ralph Keusch

28. Januar 2016

Algorithmen und Komplexität (D-MATH)
Klausur Winter 2016

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	(14)	
2	(18)	
3	(15)	
4	(15)	
5	(14)	
6	(24)	
Σ	(100)	

Aufgabe 1

(total 14 Punkte)

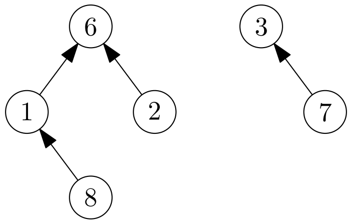
a) Zeichnen Sie den Graphen, der durch die Adjazenzmatrix

$$A = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

gegeben ist.

(4 Punkte)

b) Wir betrachten die folgende `UnionFind`-Struktur:



Führen Sie darauf die folgenden Operationen der Reihe nach nacheinander aus, inklulsive `Path-Compression`:

- `Insert`(5)
- `Union`(3, 5)
- `Find`(8)
- `Union`(`Find`(7), 6)

Zeichnen Sie jeweils die neu entstandene Struktur nach jeder der 4 Anweisungen.

(10 Punkte)

Aufgabe 2

(total 18 Punkte)

Welche der folgenden Aussagen treffen zu, und welche nicht? Begründen Sie jeweils Ihre Antwort. Für jede richtige Antwort erhalten Sie 1 Punkt, weitere 2 Punkte werden jeweils für eine korrekte Begründung verteilt. Für falsche Antworten gibt es keine Minuspunkte.

(a) $1 = \omega(e^{-\sqrt{n}})$.

(b) Es seien A, B, C, D, E fünf $n \times n$-Matrizen. Dann kann das Produkt $ABCDE$ in Laufzeit $o(n^3)$ berechnet werden.

(c) 3-Sat $\leq_p$ ErfüllbarerSchaltkreis.

(d) Sei $G = (V, E)$ ein zusammenhängender Graph mit $|V| \geq 4$, seien $s, t \in V$ und sei w eine positive Gewichtsfunktion, so dass alle Kanten $e \in E$ unterschiedliche Gewichte $w(e)$ aufweisen. Dann ist die schwerste Kante in G in keinem kürzesten s-t-Pfad enthalten.

(e) Durch Verwenden einer geeigneten Datenstruktur kann der Algorithmus von Prim in Laufzeit $O(|V| \cdot \log |E|)$ implementiert werden.

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

(6 · 3 = 18 Punkte)

Aufgabe 3

(total 15 Punkte)

Es ist bekannt, dass das Problem HamiltonCycle $\mathcal{NP}$-vollständig ist:

HamiltonCycle

Eingabe: $G = (V, E)$

Frage: Enthält G einen Kreis, der jeden Knoten genau einmal enthält?

Wir betrachten nun das folgende Problem auf Graphen:

LargeCycle

Eingabe: $G = (V, E)$

Frage: Enthält G einen Kreis mit genau $\lceil \frac{|V|}{3} \rceil$ vielen Knoten?

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

(15 Punkte)

Aufgabe 4

(total 15 Punkte)

Gegeben sei ein unsortiertes Array A mit n unterschiedlichen natürlichen Zahlen. Wir nehmen an, dass $A[1] < A[2]$ und $A[n-1] > A[n]$ gilt.

a) Zeigen Sie, dass mindestens ein lokales Maximum existiert, d.h. zeigen Sie, dass es im Array A ein $i \in \{2, \ldots, n-1\}$ gibt mit $A[i] > A[i-1]$ und $A[i] > A[i+1]$.

(3 Punkte)

b) Beschreiben Sie präzise einen Algorithmus, der in Laufzeit $O(\log n)$ ein solches lokales Maximum findet. Beweisen Sie, dass Ihr Algorithmus korrekt ist und dass er die verlangte Laufzeit erreicht.

(12 Punkte)

Aufgabe 5

(total 14 Punkte)

Es sei $\lambda \in \{0,1\}^*$ das leere Wort. Wir betrachten die Funktion $H_\lambda : \{0,1\}^* \to \{0,1\}$, die wie folgt definiert ist:

$$H_\lambda(A) := \begin{cases} 1, & \text{wenn } A \text{ ein Programm beschreibt und } A(\lambda) \text{ endliche Laufzeit hat.} \\ 0, & \text{sonst.} \end{cases}$$

Beweisen Sie, dass H_λ nicht berechenbar ist!

(14 Punkte)

Aufgabe 6

(total 24 Punkte)

Gegeben seien ein Baum $T = (V, E)$ und eine Gewichtsfunktion $f : V \to \mathbb{N}$. Für eine Teilmenge $W \subseteq V$ von Knoten sei der Wert von W definiert als $f(W) := \sum_{v \in W} f(v)$. Weiter heisst eine Teilmenge $W \subseteq V$ genau dann *unabhängig*, wenn W keine Kanten enthält, wenn also für alle $u, v \in W$ gilt $\{u, v\} \notin E$.

a) Beschreiben Sie einen Algorithmus, der in Laufzeit $O(|V|)$ den maximalem Wert $f(W)$ aller unabhängigen Mengen $W \subseteq V$ berechnet. Beweisen Sie, dass Ihr Algorithmus korrekt ist und die verlangte Laufzeit erreicht. Sie dürfen annehmen, dass der Baum als Adjazenzliste gegeben ist.

(16 Punkte)

b) Erweitern Sie Ihren Algorithmus, damit er in Laufzeit $O(|V|)$ nicht nur den Wert $f(W)$, sondern auch die unabhängige Menge W selbst ausgibt.

(8 Punkte)