Berlin, 8. Oktober 2018

Name: .... **Matr.-Nr.: ....**

Nachklausur Grundlagen der Algorithmik

(Niedermeier/Bentert, Sommersemester 2018)

Aufgabe Nr.:	1	2	3	4	5	**Summe**
Punktzahl:	10	10	10	10	10	50
Davon erreicht:						

Einlesezeit: 15 Minuten
Bearbeitungszeit: 60 Minuten
Max. Punktezahl: 50 Punkte

Allgemeine Hinweise:

- Es sind keinerlei Hilfsmittel erlaubt.
- Benutzen Sie keinen Bleistift, sondern einen Kugelschreiber in der Farbe Schwarz oder Blau.
- Beschriften Sie jedes Blatt mit Vor- und Nachnamen sowie Matrikelnummer.
- **Falls in der Aufgabenstellung nicht explizit ausgeschlossen, so sind alle Antworten zu begründen! Antworten ohne Begründung erhalten 0 Punkte.**
- Sätze aus der Vorlesung dürfen ohne Beweis verwendet werden.

Viel Erfolg!

Aufgabe 1: **Modellieren mit Maximum Flow** (10 Punkte)

Gegeben ist folgendes Problem.

THREE PER LINE

Eingabe: Gegeben ist ein $n \times n$ großes Schachbrett und eine Menge $M = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ von m zulässigen Feldern.

Frage: Ist es möglich, $3n$ Felder aus M so auszuwählen, dass in jeder Zeile und in jeder Spalte des Spielbretts genau drei Felder ausgewählt sind?

Modellieren Sie THREE PER LINE als ein MAXIMUM FLOW-Problem, sodass die Anzahl der Knoten und die Anzahl der Kanten im Flussnetzwerk polynomiell in $n + m$ ist. Geben Sie hierfür die Knoten, Kanten und Kantenkapazitäten Ihres Flussnetzwerks an und begründen Sie, wie die Antwort bestimmt werden kann.

Hinweis: Das Tupel (x_i, y_i) steht für das Feld in der x_i-ten Zeile und der y_i-ten Spalte des Sprielbretts.

Aufgabe 2: **Lineares Programmieren** (10 Punkte)

Geben Sie für das folgende Problem eine Formulierung als ganzzahliges lineares Programm (ILP) an (ohne Begründung).

Definieren Sie dazu die verwendeten Variablen, alle Nebenbedingungen sowie die Zielfunktion und geben Sie an, ob diese minimiert oder maximiert wird.

Max-SAT

Eingabe: Eine aussagenlogische Formel F in konjunktiver Normalform mit n Variablen und m Klauseln.

Aufgabe: Finden Sie eine Belegung der Variablen in F, sodass möglichst viele Klauseln in F erfüllt werden.

Aufgabe 3: **Diverses zu NP** (10 Punkte)

Kreuzen Sie die korrekten Aussagen an (ohne Begründung).

- ☐ Ist ein Problem A NP-schwer, so existiert eine nichtdeterministische Turingmaschine M_A, die A in polynomieller Zeit (in der Eingabegröße) löst.
- ☐ Sind zwei Probleme $A, B \in \mathrm{P}$, so gibt es eine Polynomzeitreduktion von A auf B.
- ☐ Ist ein Problem A in NP, so kann SAT in polynomieller Zeit (in der Eingabegröße) auf A reduziert werden.
- ☐ Ist ein Problem A in NP, so kann A in polynomieller Zeit (in der Eingabegröße) auf SAT reduziert werden.

Hinweise:

Eine nichtdeterministische Turingmaschine M_A, die A in polynomieller Zeit (in der Eingabegröße) löst, kann als ein Polynomzeit-Verifizierer $C\colon \{0,1\}^* \times \{0,1\}^* \to \{\text{JA, NEIN}\}$ interpretiert werden, der für eine gegebene binär kodierte Eingabe s und ein binär kodiertes Zertifikat t mit $|t| \leq |s|^{O(1)}$ in polynomieller Zeit (in $|s| + |t|$) entscheidet, ob $C(s,t) = \text{JA}$ (also ob $s \in A$ gilt).

SAT

Eingabe: Eine aussagenlogische Formel F in konjunktiver Normalform mit n Variablen und m Klauseln.

Frage: Ist F erfüllbar?

Aufgabe 4: **Parametrisierter Algorithmus** (10 Punkte)

Zeigen Sie, dass Clique „fixed-parameter tractable" bezüglich des Parameters „maximaler Knotengrad" ist. Geben Sie hierzu einen Algorithmus an, der Clique auf einem Graph mit n Knoten, m Kanten und maximalem Knotengrad ℓ in $f(\ell) \cdot (n+m)^{O(1)}$ Zeitschritten löst.

Hinweise:

Clique

Eingabe: Ein ungerichteter Graph $G = (V, E)$ mit n Knoten und m Kanten und eine natürliche Zahl k.

Frage: Gibt es eine Menge $K \subseteq V$ aus genau k Knoten in G, die alle paarweise miteinander durch eine Kante verbunden sind?

Der maximale Knotengrad eines Graph ist die maximale Anzahl von Knoten, die zu einem einzelnen Knoten benachbart (mit diesem durch eine Kante verbunden) sind.

Aufgabe 5: **Approximation** (10 Punkte)

Beweisen oder widerlegen Sie folgende Aussage:

Es existiert ein Algorithmus A, der in polynomieller Zeit (in der Eingabegröße) eine 1/2-Approximation für Max-SAT berechnet.

Hinweise:

Max-SAT

Eingabe: Eine aussagenlogische Formel F in konjunktiver Normalform mit n Variablen und m Klauseln.

Aufgabe: Finden Sie eine Belegung der Variablen in F, sodass möglichst viele Klauseln in F erfüllt werden.

Ein Algorithmus ist eine 1/2-Approximation für Max-SAT, wenn er für jede Formel F in der maximal opt viele Klauseln gleichzeitig erfüllt werden können, eine Belegung berechnet, die mindestens opt/2 Klauseln erfüllt.

Betrachten Sie zunächst was bei der Belegung passiert, die alle Variablen auf „true“ setzt.