Berlin, 11. Oktober 2017

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

Nachklausur Grundlagen der Algorithmik

(Nichterlein/Niedermeier, Sommersemester 2017)

Aufgabe Nr.:	1	2	3	4	5	**Summe**
Punktzahl:	8	10	10	11	11	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.**

Viel Erfolg!

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

Aufgabe 1: **Eigenschaften von Netzwerkflüssen** (8 Punkte)

Gegeben ist ein gerichteter Graph $G = (V, E)$ mit geradzahligen Kantenkapazitäten $c\colon E \to \{0, 2, 4, \ldots\}$ und zwei Knoten $s, t \in V$. Zeigen Sie, dass der größtmögliche s-t-Fluss einen geradzahligen Wert hat.

Hinweis: Nutzen Sie eine aus der Vorlesung bekannte Charakterisierung des Wertes eines größtmöglichen Flusses. Alle aus der Vorlesung bekannten Sätze brauchen nicht bewiesen zu werden.

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

Aufgabe 2: **Reduktion auf Max Flow** (10 Punkte)

Betrachten Sie die folgende Problemstellung.

Ein Restaurant hat ℓ Gerichte $g_1, \ldots, g_\ell$ auf der Speisekarte. Aufgrund von Lieferengpässen sind für jedes Gericht g_i nur noch Vorräte für e_i Essen gelagert. In dieser Situation hat sich eine Gruppe von n hungrigen Stammkunden $s_1, \ldots, s_n$ angemeldet. Es ist abzusehen, dass nicht jeder Stammkunde ein Essen seiner Wahl bekommt.

Die Stammkunden und das Restaurant einigen sich auf folgendes Vorgehen: Jeder Stammkunde s_i schickt vorab eine Menge $A_i \subseteq \{g_1, \ldots, g_\ell\}$ an Gerichten, die ihm zusagen. Das Restaurant gibt dann an jeden Stammkunden s_i entweder ein Gericht aus der Menge A_i oder ein Gutschein im Wert von 20 Euro.

Ihre Aufgabe ist es, dem Restaurant Geld zu sparen in dem Sie die eine Zuteilung der Gerichte finden, sodass möglichst wenige Gutscheine ausgegeben werden müssen.

Modellieren Sie obiges Problem als ein Maximum Flow-Problem, sodass die Anzahl der Knoten und die Anzahl der Kanten im Flussnetzwerk polynomiell in n und ℓ ist. Geben Sie hierfür die Knoten, Kanten und Kantenkapazitäten Ihres Flussnetzwerks an und begründen Sie, wie die Verteilung der Gerichte und die Anzahl der ausgegebenen Gutscheine bestimmt werden kann.

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

Aufgabe 3: **Lineares Programmieren** (10 Punkte)

Geben Sie für die beiden folgenden Probleme 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.

(a) **Partial Vertex Cover** (5 P)

Eingabe: Ein ungerichteter Graph G und eine natürliche Zahl $k > 0$.

Aufgabe: Finde k Knoten in G, sodass möglichst viele Kanten in G mindestens einen dieser Knoten als Endpunkt hat.

(b) **Triangle Deletion** (5 P)

Eingabe: Ein ungerichteter Graph G.

Aufgabe: Finde möglichst wenige Knoten in G, sodass nach deren Löschung keine Dreiecke mehr im Graph enthalten sind.

Hinweis: Ein Dreieck besteht aus drei Knoten die jeweils paarweise mit einer Kante verbunden sind.

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

Aufgabe 4: **Spezialfälle NP-schwerer Probleme** (11 Punkte)

Das NP-schwere Problem VERTEX COVER ist wie folgt definiert.

VERTEX COVER
Eingabe: Ein ungerichteter Graph G und eine natürliche Zahl $k > 0$.
Frage: Gibt es k Knoten in G, sodass jede Kante in G mindestens einen dieser Knoten als Endpunkt hat?

Argumentieren Sie, wie die folgenden drei Varianten von VERTEX COVER in Polynomzeit gelöst werden können.

(a) Der Eingabegraph G hat höchstens 100 Knoten. (3 P)

(b) Der Eingabegraph G ist ein vollständiger Graph (eine Clique). (4 P)

(c) Für die Lösungsgröße gilt $k = \log n$. (Hierbei ist n die Anzahl der Knoten in G.) (4 P)

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

Aufgabe 5: **Datenreduktion** (11 Punkte)

Das MATCHING-Problem ist wie folgt definiert:

MATCHING
Eingabe: Ein ungerichteter Graph G und eine natürliche Zahl $k > 0$.
Frage: Gibt es k Kanten in G, sodass kein Knoten Endpunkt von mehr als einer dieser Kanten ist?

Nachfolgend sind einige (vermeintlich korrekte) Datenreduktionsregeln für MATCHING angegeben. Eine Datenreduktionsregel ist korrekt, wenn die resultierende Instanz eine „Ja"-Instanz ist genau dann, wenn auch die ursprüngliche Instanz eine „Ja"-Instanz ist.

Welche der vorgeschlagenen Regeln sind korrekt? Begründen Sie Ihre Antworten.

(a) Gibt es in G einen Knoten v ohne Nachbarn, so lösche v. (3 P)

(b) Gibt es in G einen Knoten v mit genau einem Nachbarn w, so lösche v und w und verringere k um 1. (4 P)

(c) Gibt es in G ein Dreieck u, v, w mit $\deg_G(v) = \deg_G(w) = 2$, so lösche v und w und verringere k um 1. (4 P)

Hinweis: Ein Dreieck besteht aus drei Knoten die jeweils paarweise mit einer Kante verbunden sind. Für einen beliebigen Knoten v gibt $\deg_G(v)$ den Knotengrad von v in G an.