

Berlin, 21. Februar 2013

Name:

Matr.-Nr.:

Klausur Informatik-Propädeutikum
(Niedermeier/Hartung/Nichterlein, Wintersemester 2012/13)

1	
2	
3	
4	
5	
6	
7	
8	
9	
Σ	

Bearbeitungszeit: 90 min.
max. Punktezahl: 60 Punkte
min. Punktezahl zum Bestehen: 20 Punkte

Allgemeine Hinweise:

- Es sind keinerlei Hilfsmittel erlaubt.
- Benutzen Sie keinen Bleistift, sondern einen Kugelschreiber oder Füller in der Farbe schwarz oder blau.
- Beschriften Sie jedes Blatt mit Vor- und Nachnamen sowie Matrikelnummer.
- Falls in der Aufgabenstellung nicht explizit ausgeschlossen, sind alle Antworten zu begründen! Antworten ohne Begründung erhalten 0 Punkte.

Viel Erfolg!

Name:

Matr.-Nr.:

Aufgabe 1: Stable Matching (Gale und Shapley)

(2+2+2+2 Punkte)

Sei die Menge der Männer $M = \{A, B, C\}$ und die Menge der Frauen $F = \{1, 2, 3\}$ mit folgenden Präferenzlisten gegeben:

$$\begin{array}{ll} A : 1 > 2 > 3, & 1 : B > C > A \\ B : 2 > 3 > 1, & 2 : C > A > B \\ C : 1 > 2 > 3, & 3 : A > B > C \end{array}$$

- (a) Geben Sie eine „männeroptimale“ stabile Zuordnung an (ohne Begründung).
- (b) Geben Sie eine „frauenoptimale“ stabile Zuordnung an (ohne Begründung).
- (c) Gibt es immer eine stabile Zuordnung, welche sowohl männeroptimal als auch frauenoptimal ist? Hinweis: Sie können z. B. beweisen, dass dies nicht stimmt indem Sie Präferenzlisten angeben für welche dies nicht der Fall sein kann.
- (d) Geben Sie Präferenzlisten mit mindestens jeweils 3 Männern und Frauen an, welche eine stabile Zuordnung zulassen, welche sowohl männer- als auch frauenoptimal ist.

Hinweis: Eine stabile Zuordnung wird **männer-(frauen-)optimal** genannt, wenn es keine andere stabile Zuordnung gibt, welche mindestens einen Mann (Frau) besser stellt, d. h. er (sie) bekommt eine(n) Frau (Mann) mit höherer Präferenz. Der Gale-Shapley Algorithmus liefert immer eine männeroptimale Lösung.

Zur Erinnerung:

Eine Zuordnung der Männer M zu den Frauen W ist eine **stabile** Zuordnung, falls für jeden $m \in M$ und jede $w \in W$ die nicht m zugeordnet ist gilt:

- (a) m zieht die ihm zugeordnete w' gegenüber w vor, oder
- (b) w zieht den ihr zugeordneten m' gegenüber m vor.

Propose&Reject [Gale, Shapley 1962]

- 1 Initialisiere alle $m \in M$ und alle $w \in W$ als „frei“
- 2 **while** $\exists m \in M : m$ ist frei und $\exists w \in W$ der m
noch keinen Antrag gemacht hat
- 3 $w \leftarrow$ erste noch „unbeantragte“ Frau in m 's Präferenzfolge
- 4 **if** w ist frei **then:**
- 5 (m, w) wird Paar, $m \leftarrow$ „verlobt“, $w \leftarrow$ „verlobt“
- 6 **else if** w zieht m ihrem aktuellen „Verlobten“ m' vor **then:**
- 7 (m, w) wird Paar, $m \leftarrow$ „verlobt“, $w \leftarrow$ „verlobt“, $m' \leftarrow$ „frei“
- 8 **else:** w lehnt m ab

Name:

Matr.-Nr.:

Aufgabe 2: **Heuristiken für Vertex Cover**

(2 + 2 + 3 Punkte)

VERTEX COVER wurde in der Vorlesung wie folgt definiert:

VERTEX COVER

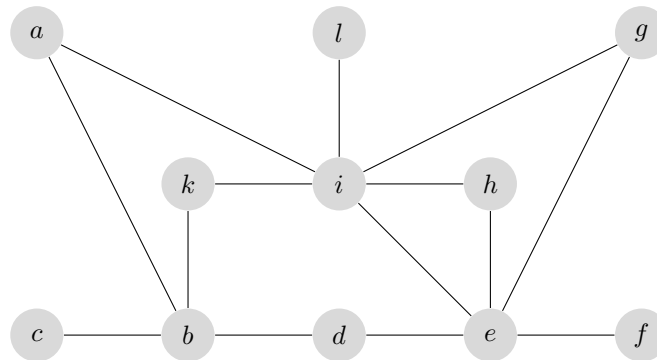
Eingabe: Ein ungerichteter Graph G .

Aufgabe: Finde eine kleinstmögliche Knotenmenge in G , sodass jede Kante mindestens einen dieser Knoten als Endpunkt hat.

Gegeben sind die folgenden beiden Heuristiken für Vertex Cover und folgender Graph G :

Heuristik 1: Solange es eine Kante gibt, wähle irgendeine, nehme beide Endpunkte dieser Kante in die Lösungsmenge und lösche diese zwei Knoten aus dem Graph.

Heuristik 2: Solange es eine Kante gibt, nehme einen Knoten mit der höchsten Anzahl anliegender Kanten in die Lösungsmenge und lösche ihn aus dem Graph.



- (a) Geben Sie für den gegebenen Graph (siehe obiges Bild) jeweils eine von Heuristik 1 und eine von Heuristik 2 erzeugte Lösungsmenge an (ohne Begründung).
- (b) Ist eine der zwei erzeugten Lösungsmengen optimal (d.h. kleinstmöglich)?
- (c) Nur eine der beiden Heuristiken ist eine Faktor-2-Approximation, d.h. die von dieser Heuristik gelieferten Lösungsmengen sind bei beliebigen Eingabegraphen maximal doppelt so groß wie die optimalen Lösungsmengen. Welche der beiden Heuristiken ist die Faktor-2-Approximation?

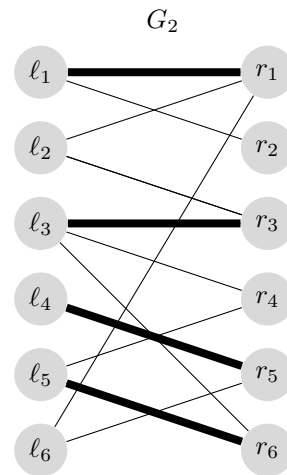
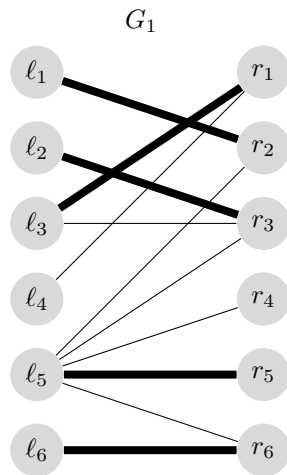
Name:

Matr.-Nr.:

Aufgabe 3: **Bipartites Matching**

(6 Punkte)

Gegeben sind folgende zwei bipartite Graphen G_1 und G_2 .



Die fett gezeichneten Kanten geben ein Matching in den jeweiligen Graphen an. Sind die beiden angegebenen Matchings größtmöglich? Wenn nicht, geben Sie ein größtmögliches Matching an.

Name:

Matr.-Nr.:

Aufgabe 4: **Algorithmische Komplexität**

(4+4 Punkte)

Für aussagenlogische Variablen spricht man von einer Klausel, wenn eine beliebige Anzahl von Literalen (für eine Variable x sind x und \bar{x} die dazugehörigen Literale) durch ein logisches ODER verknüpft wird. Eine aussagenlogische Formel befindet sich in konjunktiver Normalform, falls sie aus durch ein logisches UND verketteten Klauseln besteht.

Beispiel: $\underbrace{(x_1 \vee \bar{x}_2 \vee x_3 \vee x_5)}_{\text{Klausel}} \wedge (\bar{x}_1 \vee \bar{x}_3) \wedge (\bar{x}_2 \vee \bar{x}_3 \vee \bar{x}_4)$

Eine aussagenlogische Formel heißt **erfüllbar**, falls es eine Belegung der Variablen mit 0 oder 1 gibt, sodass die Formel sich zu 1 auswertet. Obige Formel ist erfüllbar z. B. durch die Belegung $x_1 = 1, \bar{x}_3 = 1$ (x_4 und x_5 beliebig).

Betrachten Sie die beiden folgenden Probleme.

SAT

Eingabe: Aussagenlogische Formel F in konjunktiver Normalform.

Frage: Ist F erfüllbar?

3-SAT

Eingabe: Aussagenlogische Formel F in konjunktiver Normalform mit höchstens drei Literalen pro Klausel.

Frage: Ist F erfüllbar?

Beachten Sie, dass beim SAT-Problem jede Klausel beliebig viele Variablen beinhalten kann.

Wir nehmen nun an wir hätten einen Algorithmus $\mathcal{A}^{3\text{-SAT}}$, welcher das 3-SAT-Problem in polynomieller Zeit löst. Wir wollen nun auch das SAT-Problem für eine Formel F in polynomieller Zeit mithilfe des folgenden Algorithmus lösen:

1. Konstruiere eine Formel F' , indem jede Klausel $(\ell_1 \vee \ell_2 \vee \dots \vee \ell_m)$ mit $m \geq 4$ in F ersetzt wird durch

$$(\ell_1 \vee \ell_2 \vee z_1) \wedge (\bar{z}_1 \vee \ell_3 \vee z_2) \wedge \dots \wedge (\bar{z}_{m-3} \vee \ell_{m-1} \vee \ell_m),$$

wobei z_1, z_2, \dots, z_{m-3} neue Variablen sind, welche bislang noch nicht in F enthalten sind (man führt insbesondere für jede zu ersetzende Klausel neue z -Variablen ein).

2. Löse das 3-SAT-Problem für F' mithilfe des Algorithmus $\mathcal{A}^{3\text{-SAT}}$ und gebe das Ergebnis aus.

Beweisen Sie, dass obiger Algorithmus korrekt ist. Gehen Sie dazu wie folgt vor: Sei X die Menge der Variablen in Formel F und Z die Menge der neu hinzugefügten Variablen in Formel F' .

- 1) Beweisen Sie, dass wenn F erfüllbar ist, dann auch F' .
- 2) Beweisen Sie, dass wenn F' erfüllbar ist, dann auch F .

Name:

Matr.-Nr.:

Aufgabe 5: **Ackermannfunktion**

(5 Punkte)

Die Ackermann-Funktion f ist wie folgt rekursiv definiert:

$$\begin{aligned}f(k, 1) &= 2, \\f(1, n) &= n + 2 \text{ für } n > 1, \\f(k, n) &= f(k - 1, f(k, n - 1)) \text{ für } k, n > 1.\end{aligned}$$

Berechnen Sie $f(3, 3)$.

Name:

Matr.-Nr.:

Aufgabe 6: **Persönlichkeiten**

(2+2+2 Punkte)

Nennen Sie drei zentrale Persönlichkeiten der Informatik und beschreiben Sie jeweils kurz (höchstens drei Sätze) deren herausragende wissenschaftlichen Leistungen.

Name:

Matr.-Nr.:

Aufgabe 7: Fehlerkorrigierende Codes

(2+4 Punkte)

- (a) Welche „Abstandsbedingung für Codewörter“ muss ein Code erfüllen, damit k -Bit-Fehler korrigiert werden können?
- (b) Geben Sie eine binäre Codierung der vier Buchstaben a , b , c und d an, sodass:
 - i. Alle Codewörter genau Länge fünf haben.
 - ii. Fehler von genau einem Bit korrigiert werden können.

Zeigen Sie, dass die unter (a) genannte Bedingung auch für den von Ihnen entwickelten Code gilt.

Name:

Matr.-Nr.:

Aufgabe 8: Huffman-Codierung

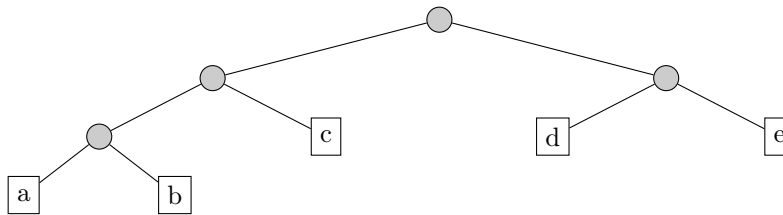
(2+4 Punkte)

Der Huffman-Baum zu einer gegebenen relativen Häufigkeit der einzelnen Zeichen wird wie folgt aufgebaut:

- 1: Erzeuge für jedes Zeichen x einen Teilbaum mit Knoten x und Gewicht $p_x :=$ relative Häufigkeit von x in M
- 2: **while** Es gibt mehr als einen Teilbaum **do**
- 3: Suche zwei Teilbäume mit kleinsten Gewichten p_1 und p_2
- 4: Vereinige die beiden Teilbäume zu einem mit Gewicht $p_1 + p_2$
- 5: **end while**

Gegeben folgende Häufigkeitsverteilung p für die Zeichen ‘a’, ‘b’, ‘c’, ‘d’, ‘e’ mit $p(a) = 1/12$, $p(b) = 1/12$, $p(c) = 1/6$, $p(d) = 1/4$, $p(e) = 5/12$.

- (a) Gehört folgender Huffman-Baum zur gegebenen Häufigkeitsverteilung? Warum? Die Häufigkeiten wurden im Baum einfachheitshalber weggelassen.



- (b) Geben Sie den zur Zeichenkette “fischersfritzelfischtfrischefische” zugehörigen Huffman-Baum an. Geben Sie die jeweiligen Gewichte der einzelnen Knoten an. Hinweis: Die betrachtete Zeichenkette hat 33 Zeichen.

Name:

Matr.-Nr.:

Aufgabe 9: **Zentrale Begriffe**

(2+2+2+2 Punkte)

Zeigen Sie in jeweils maximal drei Sätzen Verknüpfungen der Informatik zwischen folgenden Begriffpaaren auf (ggf. mit Beispiel).

- a) Zufall und Kryptologie
- b) Zufall und algorithmische Komplexität
- c) Abstraktion und Computational Thinking
- d) Fehlerkorrigierende Codes und Datenkomprimierung