Berlin, 24. Februar 2014

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

Endkontrolle Informatik-Propädeutikum

(Ermel/Hartung/Nichterlein/Niedermeier, Wintersemester 2013/14)

1	
2	
3	
4	
5	
6	
Σ	

Bearbeitungszeit: 60 min.
max. Punktezahl: 60 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: **Euklidisches TSP** (10 Punkte)

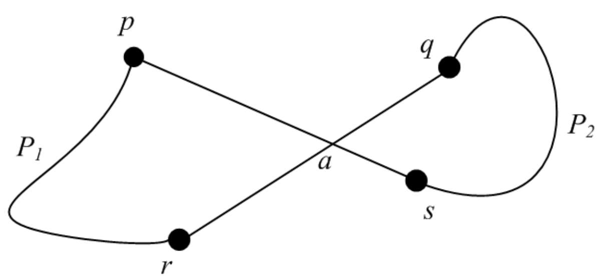
Das TRAVELLING SALESPERSON Problem wurde in der Vorlesung wie folgt definiert:

> **TSP**
> **Eingabe:** n Punkte mit paarweisen Abständen $d_{i,j}$
> **Aufgabe:** Finde eine kürzeste Rundtour, die alle Punkte genau einmal besucht.

Ein Spezialfall ist das *Euklidische TSP*, bei dem alle Punkte in der (Euklidischen) Ebene liegen und ihr paarweiser Abstand sich aus den Entfernungen ergibt. Insbesondere gilt hier die Dreiecksungleichung.

Zeigen Sie, dass es in einer optimalen Rundtour beim Euklidischen TSP keine Überschneidungen gibt.

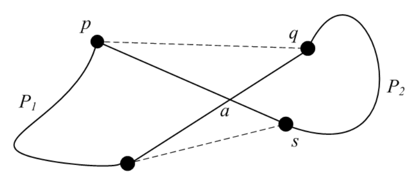
Gehen Sie dazu wie folgt vor: Sei eine Rundtour $T = (r, P_1, p, s, P_2, q, r)$ (mit geeigneten Pfaden P_1, P_2) gegeben, in der sich die Kanten ps und qr in a überschneiden. Der Einfachheit halber nehmen wir an, dass a mit keinem der anderen Punkte p, q, r, s zusammenfällt und dass p, q, r, s ein Viereck aufspannen, d.h. keine drei Punkte von p, q, r, s liegen auf einer Geraden.



Zeigen Sie nun, dass es eine andere Rundtour T' ohne Überschneidungen gibt, die die beiden Pfade P_1 und P_2 enthält, und die kürzer ist als T.

——————Lösung——————

Wir wählen die Rundtour $T' = (r, P_1, p, q, P_2^-, s, r)$ (wobei P_2^- der Pfad P_2 in umgekehrter Kantenreihenfolge sei).



Es gilt:

$$\begin{aligned} c(T') - c(T) &= d(p,q) + d(s,r) - (d(p,a) + d(a,s)) - ((d(q,a) + d(a,r)) \\ &= d(p,q) - \underbrace{(d(p,a) + d(a,q))}_{>d(p,q)} + d(s,r) - \underbrace{(d(s,a) + d(a,r))}_{>d(s,r)} \\ &< 0 \end{aligned}$$

Da die zweite Rundtour T' kürzer ist als die erste, kann T nicht optimal gewesen sein.

Punktevergabe: 2 Punkte für die richtige Rundtour, 8 Punkte für den Beweis mit Verwendung der Dreiecksungleichung.

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

Aufgabe 2: **Zentrale Begriffe** (3+3+3+3 Punkte)

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

a) Zufall und Kryptologie

b) Heuristiken und NP-schwere Probleme.

c) Abstraktion und Computational Thinking

d) Zero-Knowledge-Beweise und NP-vollständige Probleme

————Lösung————

a) Zufall wird in der Kryptologie bei der Schlüsselerzeugung benutzt.
Bsp.: Bereitstellung der Primzahlen p und q für das RSA-Verschlüsselungsverfahren erfolgt durch 1) Pseudozufallszahlengenerator liefert Zufallszahl, 2) Primzahl-Test prüft, ob es sich um eine Primzahl handelt.

b) NP-schwere Probleme sind berechnungsschwer und haben hohe Worst-Case-Komplexität. Oft kann man sie durch Verwendung von Heuristiken gut und schnell in der Praxis lösen. Heuristiken verzichten in der Regel auf die beweisbare Optimalität der gefundenen Lösung und/oder auf beweisbar schnelle Laufzeit des Algorithmus für *jede* Eingabe.
Bsp. Intervall-Scheduling, Vertex Cover (Greedy-Heuristiken), Travelling Salesperson (TSP), Euklidisches TSP, Simulated Annealing, Heuristische Suchalgorithmen (z.B. A*-Algorithmus)

c) Computational Thinking umfasst Denkprozesse, die zu Problemmodellierung und ihrer Lösung gebraucht werden. Dabei spielt der Denkprozess *Abstraktion* eine entscheidende Rolle. Computational Thinking als wesentlichen Grundbaustein der Informatik zu betrachten, ist ein Paradigmenwechsel von den *metal tools* (Transistoren, Hardware) zu unseren *mental tools* (Abstraktionsvermögen und Methoden der Abstraktion, z.B. Rekursion.
Bsp. für Abstraktion als Graphen: Königsberger Brückenproblem, Landkartenfärben, Zuordnungsprobleme (Stable Matching), Vertex Cover, Independent Set, Dominating Set, ...

d) Bei Zero-Knowledge-Beweisen wird als Beweis für bestimmtes Wissen eine Lösung präsentiert, die effizient verifizierbar ist. Bei NP-vollständigen Problemen kann man gefundene Lösungen auch effizient verifizieren. Das Finden der Lösung ist aber bei NP-vollständigen Problemen nicht effizient durchführbar.
Bsp. Alice beweist Bob, dass sie die Lösungsformel für Polynome 3. Grades kennt, indem sie Bob für ein Polynom seiner Wahl die Nullstellen mitteilt. Das kann Bob überprüfen und muss ihr dann glauben, dass sie es kann, obwohl er nicht weiß, wie sie es gemacht hat.

Punktevergabe: Je 1 Punkt für Verknüpfung im allgemeinen und 2 Punkte für ein oder mehrere passende Beispiele

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

Aufgabe 3: **Modellieren mit Graphen** (4+4 Punkte)

Modellieren Sie die folgenden Fragestellungen als Graphprobleme. Unter welchem Namen tauchen Ihre Graphprobleme in der Vorlesung auf?

a) Sie geben eine Party und wollen möglichst viele, aber mindestens k Ihrer n Freunde einladen. Allerdings gibt es unter diesen Freunden einige, die sich gegenseitig nicht leiden können. Um keine schlechte Stimmung aufkommen zu lassen, können Sie von zwei Freunden, die sich gegenseitig unsympathisch sind, nicht beide einladen. Ihnen ist bekannt, welche Ihrer n Freunde sich gegenseitig unsympathisch sind. Die Aufgabe ist es, mindestens k Ihrer n Freunde auszuwählen und einzuladen.

b) In einem Rechnernetz soll es bestimmte ausgezeichnete Rechner geben (Gateways). Verschiedene Rechnernetze kommunizieren nur über ihre Gateways miteinander. Deshalb soll gelten: Jeder Rechner, der kein Gateway ist, soll direkt mit einem Gateway verbunden sein, um Nachrichten über das Gateway nach außen übertragen zu können. Das Problem ist nun, möglichst wenige, aber höchstens k Rechner im Netz als Gateways zu bestimmen.

——————Lösung——————

a) Die Problemstellung ist äquivalent zum Independent Set Problem (eine Untermenge von k Knoten, die nicht paarweise durch Kanten verbunden sind). Dabei werden Freunde als Knoten modelliert und die Knoten sind durch eine Kante verbunden, falls sie sich nicht leiden können.

b) Das Problem ist identisch zum Dominating Set Problem (Untermenge D von Knotenmenge V, so dass jeder Knoten in V entweder in D ist oder einen Nachbarn in D hat). Dabei werden die Rechner als Knoten modelliert, und jeder Knoten ist entweder selbst Gateway (also in D oder mit einem verbunden.

Punktevergabe: Je ein Punkt für das richtige Graphproblem und 2 Punkte für eine vernünftige Erklärung.

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

Aufgabe 4: **Heuristiken für Vertex Cover** (4 + 4 + 4 Punkte)

Vertex Cover wurde in der Vorlesung wie folgt definiert:

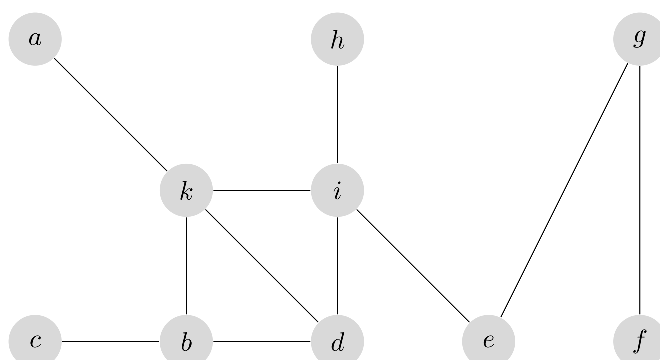
> **Vertex Cover**
> **Eingabe:** Ein ungerichteter Graph G.
> **Aufgabe:** Finde eine kleinstmögliche Knotenmenge S in G, sodass jede Kante mindestens einen der Knoten in S als Endpunkt hat.

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

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

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

Graph G:



(a) Geben Sie für den Graphen G 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?

——————Lösung——————

(a) Heuristik 1: z. B. $\{i, k, g, f, b, c\}$ oder $\{i, d, g, e, k, b\}$ (nicht eindeutig)
Heuristik 2: $\{i, b, k, g\}$

(b) Die Lösung von Heuristik 2 ist optimal: Die Knoten a, c, f, h haben alle jeweils genau einen Nachbarn und diese Nachbarn sind verschieden $\rightsquigarrow$ mindestens 4 Knoten werden in einer Lösung benötigt.

(c) Nur Heuristik 1 ist eine Faktor-2-Approximation: für jede der betrachteten Kanten muss mindestens einer der beiden anliegenden Knoten in einer Lösung enthalten sein - die Heuristik nimmt beide $\rightsquigarrow$ Faktor-2-Approximation.

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

Aufgabe 5: **Persönlichkeiten** (6 Punkte)

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

——————Lösung——————

Siehe Folien. Z.B. Zuse, Turing, Gödel, von Neumann, Huffman, Babbage, Lovelace, Shapley, Knuth, Ackermann, Post, Church, Rice, Rechenberg, Markov, Shannon, Shamir, Phil Zimmermann, Kemeny, Condorcet, Jeannette Wing

Punktevergabe: je 1 Punkt für passende Persönlichkeit und 2 Punkte für passende Leistungen.

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

Aufgabe 6: **Kemeny-Verfahren** (4+4+4 Punkte)

Gegeben sei eine Wahl mit n Wählern und m Kandidaten, wobei jeder Wähler eine Präferenzliste über die m Kandidaten hat. Der **Kendall-Tau-Abstand** zwischen zwei Präferenzlisten ist die Anzahl der unterschiedlich geordneten Kandidatenpaare. Die **Kemeny-Punktzahl** einer Präferenzliste ist die Summe der Kendall-Tau-Abstände zu allen Präferenzlisten. Eine **Kemeny-Konsensliste** ist eine Präferenzliste mit kleinstmöglicher Kemeny-Punktzahl.

Laut **Condorcet-Verfahren** gewinnt ein Kandidat, wenn er gegen jeden anderen Kandidaten die Mehrheit im direkten Vergleich erhält.

Sei eine Wahl mit 6 Wählern und 3 Kandidaten a, b, c gegeben. Die Präferenzlisten dieser 6 Wähler sehen so aus:

Wähler 1: $c \succ b \succ a$
Wähler 2: $b \succ a \succ c$
Wähler 3: $b \succ c \succ a$
Wähler 4: $a \succ b \succ c$
Wähler 5: $a \succ b \succ c$
Wähler 6: $a \succ b \succ c$

Die Präferenzlisten sind so zu verstehen: z.B. findet Wähler 1 den Kandidaten c am besten, gefolgt von b, und mag a am wenigsten.

a) Hat obige Wahl einen Condorcet-Gewinner?

b) Die kleinstmögliche Kemeny-Punktzahl einer Präferenzliste ist 6. Geben Sie eine mögliche Kemeny-Konsensliste für die obige Wahl an.

c) Hat eine Wahl mit zwei Kandidaten immer einen Condorcet-Gewinner?

——————Lösung——————

a) Obige Wahl hat keinen Condorcet-Gewinner, denn die Kandidaten a und b schlagen jeweils den Kandidaten c im direkten Vergleich, aber Kandidaten a und b sind untereinander ausgeglichen.

b) $a \succ b \succ c$ oder $b \succ a \succ c$.

c) Nein, nicht immer. Wenn die Anzahl der Wähler gerade ist, dann kann es sein, dass die Anzahl der Wähler die einen Kandidaten besser als den anderen finden, gleich der Anzahl der Wähler ist, die genau entgegengesetzter Meinung sind. Dann hat diese Wahl keinen Condorcet-Gewinner.
