1. Klausur zur Vorlesung
Algorithmentechnik
Wintersemester 2009/2010

Hier Aufkleber mit Name und Matrikelnummer anbringen

Vorname: ____________________

Nachname: ____________________

Matrikelnummer: ____________________

Beachten Sie:

- Bringen Sie den Aufkleber mit Ihrem Namen und Matrikelnummer auf diesem Deckblatt an und beschriften Sie jedes Aufgabenblatt mit Ihrem Namen und Matrikelnummer.
- Schreiben Sie die Lösungen auf die Aufgabenblätter und Rückseiten. Zusätzliches Papier erhalten Sie bei Bedarf von der Aufsicht.
- Zum Bestehen der Klausur sind **20** der möglichen **60** Punkte hinreichend.
- Es sind keine Hilfsmittel zugelassen.

Aufgabe	Mögliche Punkte					Erreichte Punkte				
	a	b	c	d	Σ	a	b	c	d	Σ
1	1	2	2	-	5				-	
2	1	2	3	-	6				-	
3	3	1	-	-	4			-	-	
4	4	2	-	-	6			-	-	
5	1	2	2	-	5				-	
6	3	2	3	2	10					
7	3	4	-	-	7			-	-	
8	2	4	-	-	6			-	-	
9	11x1				11					
Σ					60					

Problem 1: Union-Find $1 + 2 + 2 = 5$ Punkte

Im Folgenden sei Union-Find mit Weighted Union und Pfadkompression implementiert, wie in der Vorlesung vorgestellt.

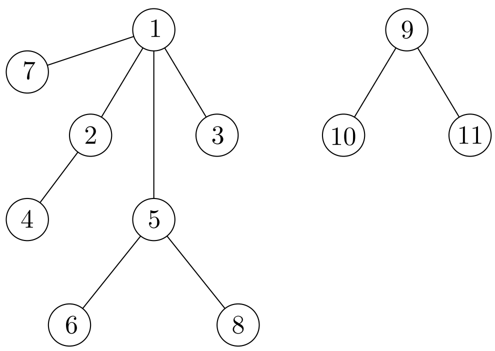
(a) Gegeben sei folgendes Array A, das den Zustand einer Union-Find-Datenstruktur mit 9 Elementen repräsentiert. Zeichnen Sie die in A gegebene Datenstruktur in der Baum-Repräsentation (wie in der Vorlesung vorgestellt).

x	1	2	3	4	5	6	7	8	9
Vor(x)	9	-6	7	2	7	9	-3	2	2

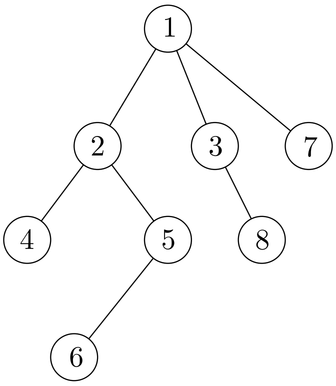
(b) Gegeben sei folgende Union-Find-Datenstruktur in der Baum-Repräsentation sowie die Operationen

(i) UNION(FIND(8),FIND(10)) und
(ii) FIND(11).

Zeichnen Sie den Zustand der Datenstruktur (als Baum-Repräsentation) sowohl nach Ausführung der Operation (i) als auch nach **anschließender** Ausführung der Operation (ii).



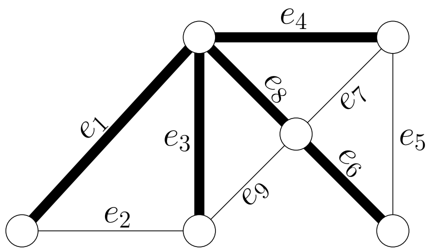
(c) Betrachten Sie den unten abgebildeten Zustand einer Union-Find-Datenstruktur. Geben Sie eine Folge von Operationen MAKESET, UNION und FIND an, die den dargestellten Zustand der Datenstruktur erzeugen *kann*.



MAKESET(1), ..., MAKESET(8)

Problem 2: Kreisbasen $1 + 2 + 3 = 6$ Punkte

(a) Untenstehende Abbildung zeigt den Graphen $G^* := (V^*, E^*)$. Gegeben sei weiterhin der aufspannende Baum $T^* = \{e_1, e_3, e_4, e_6, e_8\}$ (fett eingezeichnet). Geben Sie die durch T^* induzierte Fundamentalbasis an. Stellen Sie die Fundamentalkreise als Teilmengen von E^* dar.

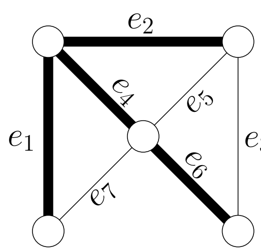


(b) Gegeben sei ein Graph $G := (V, E)$. Weiter sei $T := (V, E_T)$ ein aufspannender Baum von G, und $e \in E_T$ sei eine **Baumkante**. Die Kante e sei in zwei Fundamentalkreisen C_1 und C_2 bzgl. T enthalten.

Sei $T' = (V, E'_T)$ ein weiterer aufspannender Baum von G, der dieselbe Fundamentalbasis induziert wie T. Zeigen Sie, dass e in E'_T enthalten sein muss.

(c) Untenstehende Abbildung zeigt den Graphen $\hat{G} := (\hat{V}, \hat{E})$. Gegeben sei weiterhin der aufspannende Baum $\hat{T} = \{e_1, e_2, e_4, e_6\}$ (fett eingezeichnet) sowie die zugehörige Fundamentalbasis mit $C_3 := \{e_2, e_3, e_4, e_6\}$, $C_5 := \{e_2, e_4, e_5\}$ und $C_7 := \{e_1, e_4, e_7\}$.

Geben Sie alle weiteren aufspannenden Bäume von $\hat{G}$ an, die dieselbe Fundamentalbasis induzieren wie $\hat{T}$. Begründen Sie, weshalb die von Ihnen angegebene Baummenge maximal ist.



Problem 3: Maybe-MST $3 + 1 = 4$ Punkte

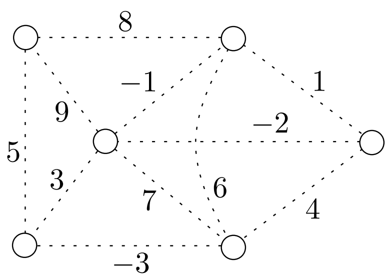
Gegeben sei folgender Algorithmus MAYBE-MST. Die Eingabe sei ein gewichteter Graph $G = (V, E, c)$ mit Gewichtungsfunktion $c : E \to \mathbb{R}$. Desweiteren bezeichne $\min(v) \subseteq E$ die Menge der minimal gewichteten Kanten unter allen zu $v \in V$ inzidenten Kanten.

Algorithmus 1 : MAYBE-MST

Eingabe : Graph $G = (V, E, c)$
Ausgabe : Kantenmenge T

$T \leftarrow \emptyset$
für alle $v \in V$ **tue**
  $T \leftarrow T \cup \min(v)$
Sortiere $E \setminus T$ in nicht absteigender Reihenfolge
Für *sortierte Kanten, wende nacheinander an:*
  Wenn *Endknoten der Kante in verschiedenen Zusammenhangskomponenten von* T *liegen*
    Füge Kante zu T hinzu
Return T

(a) Wenden Sie MAYBE-MST auf den unten abgebildeten Graphen G^* an. Zeichnen Sie die Kanten in T fett durchgezogen. Berechnet MAYBE-MST für G^* einen minimalen aufspannenden Baum?



(b) Zeichnen Sie einen nicht-leeren, zusammenhängenden, gewichteten Eingabegraphen, für den MAYBE-MST **keinen** minimalen aufspannenden Baum berechnet.

Problem 4: Matroide $4 + 2 = 6$ Punkte

Sei $G = (V, E)$ ein einfacher, gerichteter Graph. Für jeden Knoten $v \in V$ sei durch $k_v \in \mathbb{Z}^+$ eine positive, ganze Zahl gegeben.

Wir definieren ein Mengensystem $(M,\mathcal{U})$ durch $M := E$ und

$$\mathcal{U} := \{I \subseteq E \mid \mathrm{indeg}_I(v) \leq k_v \text{ für alle } v \in V\}.$$

Dabei bezeichne $\mathrm{indeg}_I : V \to \mathbb{N}_0$ den *Eingangsgrad* eines Knotens bezüglich der Kantenmenge I, das heißt $\mathrm{indeg}_I(v) := |\{u \in V \mid (u, v) \in I\}|$.

(a) Zeigen Sie, dass $(M,\mathcal{U})$ ein Matroid ist.

Sei nun $G = (V, E)$ ein einfacher, *ungerichteter* Graph. Analog zu (a) sei das Mengensystem $(M,\mathcal{U})$ definiert durch $M := E$ und

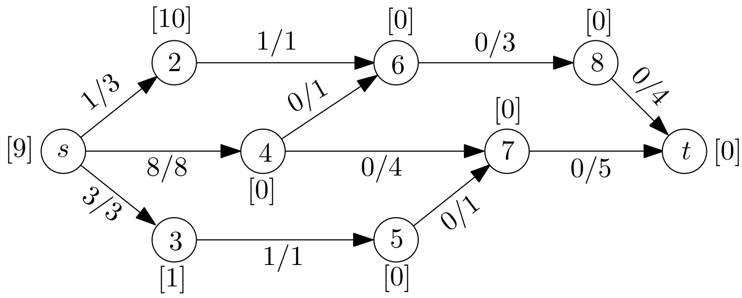
$$\mathcal{U} := \{I \subseteq E \mid \deg_I(v) \leq k_v \text{ für alle } v \in V\}.$$

Dabei ist $\deg_I : V \to \mathbb{N}_0$ der *Grad* eines Knotens bezüglich der Kantenmenge I, das heißt $\deg_I(v) := |\{u \in V \mid \{u, v\} \in I\}|$.

(b) Zeigen Sie durch Angabe eines Gegenbeispiels, dass $(M,\mathcal{U})$ im Allgemeinen kein Matroid ist.

Problem 5: Goldberg-Tarjan-Fluss-Algorithmus $1 + 2 + 2 = 5$ Punkte

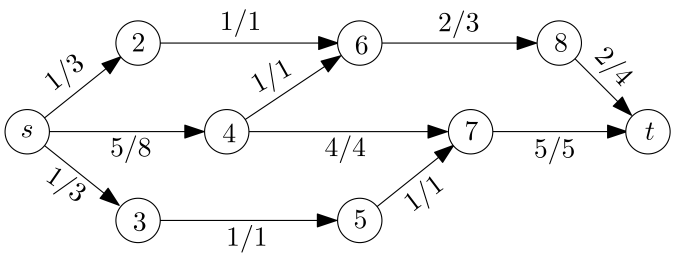
Untenstehendes Netzwerk $(D; s; t; c)$ zeigt einen Zwischenschritt des Goldberg-Tarjan-Algorithmus. Die Distanzlabel $dist(v)$ der Knoten (aus der Prozedur RELABEL im Skript) sind in eckigen Klammern über den Knoten notiert, die Kanten sind im Format $f(e)/c(e)$ beschriftet. Dabei bezeichnet $c(e)$ die Kantenkapazität und $f(e)$ den Fluss auf der Kante e.



(a) Markieren Sie in obiger Abbildung alle aktuell aktiven Knoten.

(b) Ausgehend von obigem Zustand, geben Sie eine mögliche Folge von PUSH- und RELABEL-Aufrufen an, nach deren Ausführung Knoten 4 erstmals nicht mehr aktiv ist (Hinweis: Solange Knoten 4 aktiv ist, wählen Sie diesen als aktuellen aktiven Knoten).

(c) Untenstehende Abbildung zeigt einen maximalen s-t-Fluss auf dem Netzwerk. Zeichnen Sie alle minimalen s-t-Schnitte ein (Schnittgewicht bzgl. Kantenkapazität).



Problem 6: Maximum Matching $3 + 2 + 3 + 2 = 10$ Punkte

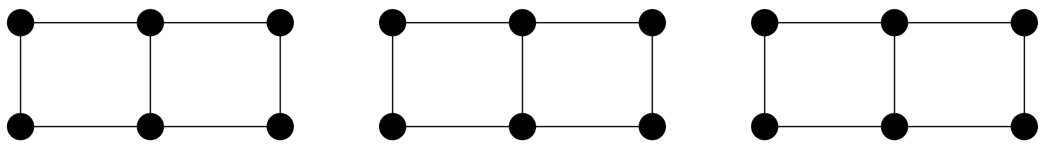
In einem einfachen, ungewichteten Graphen $G = (V, E)$ ist ein *Matching* eine Kantenmenge $M \subseteq E$, so dass keine zwei Kanten aus M zu einem gemeinsamen Knoten inzident sind. Ein Matching M heißt

- **inklusions***maximal*, wenn es keine echte Teilmenge eines anderen Matchings M' ist, d.h. es existiert kein Matching M', so dass $M \subsetneq M'$.
- **kardinalitäts***maximal*, wenn es kein Matching M' mit echt größerer Kantenanzahl gibt, d.h. es existiert kein Matching M', so dass $|M| < |M'|$.

(a) Zeichen Sie in die untenstehenden Graphen Folgendes ein:

- in (i) ein **kardinalitäts**maximales Matching
- in (ii) ein **inklusions**maximales aber **kein kardinalitäts**maximales Matching
- in (iii) ein nicht-leeres, **nicht inklusions**maximales Matching

Zeichnen Sie hierzu die Kanten der Matchings fett ein.



(i) **kardinalitäts**maximal. (ii) **inklusions**maximal, nicht kardinalitätsmaximal. (iii) nicht leer, **nicht** inklusionsmaximal.

Eine Knotenmenge $V' \subseteq V$ heißt *Überdeckung* der Kantenmenge E, wenn für alle $e \in E$ gilt: $\exists u \in V'$ mit u inzident zu e.

(b) Sei M ein beliebiges Matching in $G = (V, E)$.
Sei V' eine beliebige Überdeckung von E.

Zeigen Sie: $|M| \leq |V'|$,
wobei $|M|$ die Anzahl der Kanten in M und $|V'|$ die Anzahl der Knoten in V' bezeichnet.

(c) Sei $M_{\max}$ ein **inklusions**maximales Matching in $G = (V, E)$.
Sei $V'_{\min}$ eine Überdeckung von E mit minimaler Kardinalität.

Zeigen Sie: $|V'_{\min}| \leq 2\,|M_{\max}|$,
wobei $|V'_{\min}|$ die Anzahl der Knoten in $V'_{\min}$ und $|M_{\max}|$ die Anzahl der Kanten in $M_{\max}$ bezeichnet.

(d) Sei I eine Instanz des Matchingproblems.
Sei M_M ein **kardinalitäts**maximales Matching bzgl. I.
Sei $M_{\max}$ ein **inklusions**maximales Matching bzgl. I.

Zeigen Sie: $\mathrm{OPT}(I) \leq 2 \cdot \mathcal{A}(I)$,
mit $\mathrm{OPT}(I) := |M_M|$, $\mathcal{A}(I) := |M_{\max}|$.

(Hinweis: Man kann die Teilaufgaben (b) und (c) nutzen.)

Problem 7: Lineares Programm für kürzeste Wege 3 + 4 = 7 Punkte

Gegeben sei ein einfacher, gerichteter, gewichteter Graph $G = (V, E, w)$ mit einer *echt-positiven* Gewichtsfunktion $w : E \to \mathbb{R}^+$ und zwei Knoten $s, t \in V$ mit $s \neq t$.

Ein s-t-Weg ist ein Weg mit Startknoten s und Endknoten t.
Ein *kürzester s-t-Weg* ist ein s-t-Weg mit minimalem Gewicht unter allen s-t-Wegen.

(a) Gegeben sei ein kürzester s-t-Weg $P = (s = v_1, v_2, \ldots, v_{k-1}, v_k = t)$.
Weiterhin sei für alle Kanten $(u, v) \in E$

$$x_{u,v} := \begin{cases} 1, & (u,v) \in P \\ 0, & \text{sonst.} \end{cases}$$

Zeigen Sie, dass für jeden Knoten $v \in V$ mit $v \neq s, t$ gilt:

$$\sum_{(u,v)\in E} x_{u,v} - \sum_{(v,u)\in E} x_{v,u} = 0$$

(b) Geben Sie ein (ganzzahliges) lineares Programm an, dessen Optimallösung einen kürzesten s-t-Weg P liefert. Wählen Sie für jede Kante $(u, v) \in E$ als Variable $x_{u,v} \in \{0, 1\}$ mit der gleichen Bedeutung wie in Aufgabe (a).

Problem 8: Kernbildung für 3-HITTING SET 2 + 4 = 6 Punkte

Das 3-HITTING SET-Problem ist wie folgt definiert.

Gegeben: Eine Menge V und eine Menge $\Phi \subseteq 2^V$ von Teilmengen von V.
Dabei hat jede Menge in Φ höchstens 3 Elemente.
Ein Parameter k.

Gesucht: Eine Teilmenge $H \subseteq V$ mit $|H| \leq k$,
so dass H mindestens ein Element aus jeder der Mengen aus Φ enthält.

(a) Betrachten Sie folgende Instanz:
$V = \{1, 2, 3, 4, 5\}$, $\Phi = \{\{1, 2, 3\}, \{2, 5\}, \{3, 4, 5\}\}$, $k = 2$.
Geben Sie zwei verschiedene 3-Hitting Sets aus jeweils maximal 2 Elementen an.

(b) Sei $x, y \in V$ mit $x \neq y$,
und sei $\Phi_{x,y} \subseteq \Phi$ die Menge aller Tripel, die x und y enthalten.
Außerdem sei $|\Phi_{x,y}| > k$.

Zeigen Sie: Jedes 3-Hitting Set H für Φ mit $|H| \leq k$
ist auch ein 3-Hitting Set für $\Phi' := (\Phi \setminus \Phi_{x,y}) \cup \{\{x, y\}\}$.

Problem 9: $11 \times 1 = 11$ Punkte

Kreuzen Sie für folgende Aussagen an, ob diese wahr oder falsch sind. Für jede richtige Antwort gibt es einen Punkt, für jede falsche Antwort wird ein Punkt abgezogen. Es wird keine negative Gesamtpunktzahl für diese Aufgabe geben.

Auf ein Fluss-Netzwerk $(D; s; t; c)$ mit ganzzahligen Kapazitäten wird der FORD-FULKERSON-Algorithmus angewendet. Der so berechnete maximale s-t-Fluss ist dann ebenfalls auf jeder Kante ganzzahlig.	☐ Wahr	☐ Falsch
Jedes APAS ist ein FPAS.	☐ Wahr	☐ Falsch
Die Klasse $\mathcal{NC}$ (Nick's Class) ist die Klasse der Probleme, die durch einen parallelen Algorithmus mit polylogarithmischer Laufzeit und polynomieller Prozessorzahl gelöst werden kann.	☐ Wahr	☐ Falsch
Das MAX-2-SAT-Problem ist $\mathcal{NP}$-schwer.	☐ Wahr	☐ Falsch
Jede Fundamentalbasis bezüglich eines MST ist eine minimale Kreisbasis.	☐ Wahr	☐ Falsch
Für ein Problem mit der Rekursionsgleichung $T(n) = T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) + n$ gilt $T(n) \in \Theta(n \cdot \log n)$.	☐ Wahr	☐ Falsch
Sei P ein lineares Programm mit optimalem Lösungsvektor x^*, und D sei das zugehörige duale Programm mit optimalem Lösungsvektor y^*, so gilt $x^* = y^*$.	☐ Wahr	☐ Falsch
In jeder Phase des Stoer & Wagner Schnitt-Algorithmus kann der Startknoten beliebig gewählt werden, ohne dass die Korrektheit des Algorithmus dadurch beeinflusst wird.	☐ Wahr	☐ Falsch
Sei G ein Graph, C ein Vertex Cover und D ein Dominating Set von G. Dann gilt $\|C\| \leq \|D\|$.	☐ Wahr	☐ Falsch
Sei $\mathcal{A}$ ein 2-Approximationsalgorithmus für ein Maximierungsproblem, dann gilt $\mathcal{A}(I) \geq \frac{1}{2}\mathrm{OPT}(I)$. Dabei beschreibt I eine Probleminstanz.	☐ Wahr	☐ Falsch
Sei G ein Graph und T ein aufspannender Baum von G. Dann trennt jeder nichttriviale Schnitt in G ein in T adjazentes Knotenpaar.	☐ Wahr	☐ Falsch