Klausur

Name: ____________________

Matrikelnummer: ____________________

Die Klausur besteht aus mehreren Aufgaben, die Sie in beliebiger Reihenfolge lösen können. Wir empfehlen Ihnen, vor der Bearbeitung erst alle Aufgaben durchzulesen und mit den Ihnen einfach erscheinenden anzufangen. Wenn Sie mit einer Aufgabe nicht weiter kommen, erledigen Sie erst die anderen Aufgaben und probieren es später nochmal. Des Weiteren beachten Sie folgende Anforderungen:

- Notieren Sie auf allen Blättern Ihren Namen und, falls nicht vorgedruckt, die Aufgabennummer. Jede neu bearbeitete Aufgabe fängt auf einem separaten Blatt an, d.h. es existiert kein Blatt, auf dem mehr als eine Aufgabe bearbeitet worden ist. Zusätzliches Papier können Sie stets anfordern.
- Als Hilfsmittel ist neben dokumentenechtem Schreibgerät und einem Wörterbuch auch ein einzelner, doppelseitiger DIN A4 "Spickzettel" erlaubt.
- Nicht erlaubt sind: Bleistifte (nicht dokumentenecht) und Schreibgeräte mit roter Farbe. Mit nicht erlaubten Arbeitsmitteln ausgefüllte Aufgaben werden nicht bewertet. Weiterhin sind keinerlei elektronische Hilfsmittel, wie Taschenrechner oder Mobiltelefone erlaubt. Zuwiderhandlung wird als Täuschungsversuch gewertet.
- Die Teilnahme an der Klausur steht unter dem Vorbehalt, dass die Voraussetzungen für eine Zulassung vorlagen.

Viel Erfolg!

Punktespiegel, **nicht** vom Prüfling auszufüllen!

Aufgabe	mögliche Punkte	erreichte Punkte	Korrektor
1	8		
2	14		
3	6		
4	7		
5	4		
6	6		
7	9		
8	9		
9	6		
Summe	69		
	Note		

Name: ____________________ Punkte:

Aufgabe 1 Kurze Aufgaben, kurze Antworten (8 Punkte)

a) (1P) Bestimmen Sie die asymptotische Laufzeit mit Hilfe des Master-Theorems.

$$\begin{aligned} T(n) &= 9 \cdot T(\tfrac{n}{2}) + n^3 \\ T(1) &= 1 \end{aligned}$$

b) (2P) Bestimmen Sie die asymptotische Laufzeit mit Hilfe des Akra-Bazzi-Theorems. Sie haben die Vermutung, dass die Laufzeit in etwa quadratisch sein könnte.

$$\begin{aligned} T(n) &= 2 \cdot T(\tfrac{n}{2}) + 8 \cdot T(\tfrac{n}{4}) + 3 \cdot n^2 \\ T(1) &= 1 \end{aligned}$$

c) (2P) Lösen Sie mit der Methode von Karazuba die Berechnung von $12 \cdot 23$.

d) (2P) Sie haben eine Balkenwaage und 14 Kugeln. 13 Kugeln haben das gleiche Gewicht, nur eine hat ein anderes Gewicht (ob leichter leichter oder schwerer ist nicht bekannt). Wie viele Blätter hat ihr Entscheidungsbaum maximal, wenn Sie dreimal wiegen dürfen? Ist es möglich, mit dreimaligem Wiegen die eine andere Kugel zu finden? [1]

e) (1P) Wie würden Sie in einem planaren Graph das all-pairs-shortest-Path-Problem lösen, wenn Sie eine Laufzeit in $o(|V|^3)$ erreichen möchten?

[1] Begründen Sie Ihre Antworten. Es ist **nicht** nach einer genauen Wiege-Strategie gefragt, sondern nur, ob es prinzipiell möglich ist.

Name: Punkte:

Aufgabe 2 Ja/Nein bzw. Wahr/Falsch (14 Punkte)

Beantworten Sie die folgenden Fragen mit Ja oder Nein bzw. Aussagen mit Wahr oder Falsch. Begründen Sie Ihre Antwort! Auf unbegründete Antworten werden keine Punkte vergeben.

a) (1P) $2^{n+k} \in O(2^n)$ für $k \in \mathbb{N}$ konstant.

b) (2P) $2^{kn} \in O(2^n)$ für $k \in \mathbb{N}$ konstant.

c) (2P) $\log(n!) \in O(n \log n)$

d) (2P) Gegeben sei ein gerichteter Graph $G = (V,E)$ mit Kantengewichten $w : E \to \mathbb{R}$. Sei $\ell > 0$ und $G' = G$ mit Kantengewichten $w' : E \to \mathbb{R}$, $w'(u,v) \mapsto w(u,v) \cdot \ell$. Sind die kürzesten Wege in G und G' gleich?

Name:

e) (1P) Wie d), jedoch sei $w'(u,v) \mapsto w(u,v) + \ell$.

f) (1P) In einem bipartiten Graphen $G = (S \cup T,E)$ mit $|S| < |T|$ gebe es ein maximales Matching M. Wenn zu S ein Knoten v hinzugefügt wird, der mit allen Knoten aus T verbunden ist, gibt es dann ein Matching M' mit $|M'| > |M|$?

g) (1P) Wie f), nur ist jetzt $|S| \geq |T|$.

h) (1P) Gegeben ist eine Folge $a_1, \dots ,a_n$ von natürlichen Zahlen, für die es ein j mit $1 < j < n$ gibt, so dass $a_1 > a_2 > \cdots > a_j$ und $a_j < a_{j+1} < \cdots < a_n$. Kann man das Maximum der Folge in konstanter Zeit, also $O(1)$, bestimmen?

Name:

i) (2P) Für alle Schnitte eines Graphen gibt es eine eindeutige Kante mit dem kleinsten Gewicht $\Rightarrow$ der minimale Spannbaum des Graphen ist eindeutig.

j) (1P) In i) gilt auch $\Leftarrow$.

Name: Punkte:

Aufgabe 3 Faire unfaire Münze (6 Punkte)

Wir haben eine unfaire Münze gegeben, bei der mit Wahrscheinlichkeit p das Wurfergebnis *Zahl* auftritt und mit Wahrscheinlichkeit $(1-p)$ das Ergebnis *Kopf*, wobei $0 < p < \frac{1}{2}$ gilt. Um mit dieser Münze zu einer fairen Enscheidung zu kommen, verwenden wir folgenden Algorithmus:

```
do
    w1 = Werfe Münze;
    w2 = Werfe Münze;
while (w1 == w2);
if ( w1 == Kopf)
    return "ja";
else
    return "nein";
```

a) (2P) Beweisen Sie, dass der Rückgabewert des Algorithmus eine faire Münze simuliert, also gilt:

$$P(\text{Rückgabe "ja"}) = P(\text{Rückgabe "nein"})$$

b) (1P) Sei X die Zufallsvariable, die beschreibt wie viele Schleifendurchläufe nötig sind. Bestimmen Sie $P(X = i), i \in \mathbb{N}$.

c) (3P) Bestimmen Sie die erwartete Laufzeit des Algorithmus (als Anzahl der Schleifendurchläufe) in Abhängigkeit von p. Finden Sie eine Formel ohne Summenzeichen.

Name: Punkte:

Aufgabe 4 Erzeugende Funktionen (7 Punkte)

Es sei

$$f_n = \begin{cases} 1 & \text{für } n = 0 \\ 1 & \text{für } n = 1 \\ f_{n-1} + 6 \cdot f_{n-2} & \text{für } n > 1 \end{cases}$$

Zeigen Sie

a) (4P)

$$F(z) = \frac{3}{5}\frac{1}{1-3z} + \frac{2}{5}\frac{1}{1+2z}$$

b) (3P)

$$f_n = \frac{1}{5}\left(3^{n+1} + (-1)^n \cdot 2^{n+1}\right)$$

Name: Punkte:

Aufgabe 5 Amortisierte Analyse (4 Punkte)

Eine Queue mit den Operationen `add`, und `del`, lässt sich auch mit zwei Stacks S und T implementieren. Die Operationen der Queue werden dann mit den Operationen der beiden Stacks wie folgt implementiert:

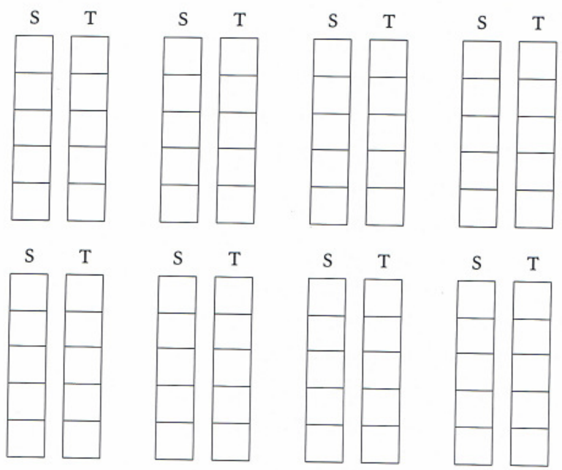
```
add(k):            del:
                     if ( T.is_empty )
  S.push(k)            while ( S.not_empty )
                         T.push(S.pop)
                     return (T.pop)
```

a) (2P) Verdeutlichen Sie die Funktionsweise dieser Implementierung, indem Sie jeweils die Zustände nach den folgenden Operationen illustrieren:

`add`(1), `add`(2), `del`, `add`(3), `add`(4), `del`, `del`, `add`(5), `del`, `del`

b) (2) Wir nehmen an, dass `push` und `pop` jeweils den Aufwand 1 haben. Wie hoch sind die tatsächlichen Kosten von `add` und `del`? Wie viel müssen Sie beim Ausführen von `add` und `del` "bezahlen", damit spätere Kosten mit ausgeglichen sind?

zu a):

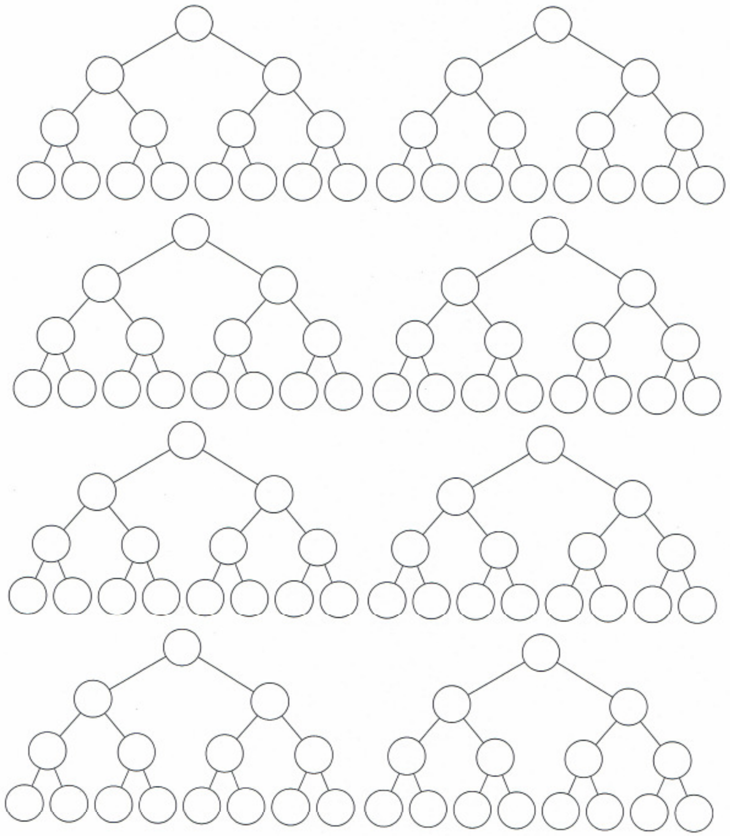


Aufgabe 6 AVL-Bäume (6 Punkte)

Fügen Sie nacheinander folgende Zahlen in einen anfangs leeren AVL-Baum ein:

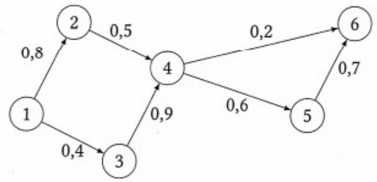
$$2, 3, 7, 4, 6, 5, 1$$

Geben Sie das Aussehen des Baumes nach jedem Einfügen und nach jeder Rotation an, d.h. falls eine Doppelrotation nötig ist, um den Baum wieder zu balancieren, sind es drei Diagramme.



Aufgabe 7 Sichere Übertragung (9 Punkte)

Gegeben sei das folgende Kommunikationsnetzwerk als Graph $G = (V,E)$, sowie eine Funktion $p : E \to [0,1]$, welche die Wahrscheinlichkeit angibt, dass eine Nachricht korrekt, also ohne Fehler, über die jeweilige Kante übertragen wird. Die Wahrscheinlichkeit, dass eine Nachricht von A nach B korrekt übertragen wird, ergibt sich aus dem **Produkt** der Kantenwahrscheinlichkeiten, die auf dem Weg liegen.



Beispiel: Angenommen, man möchte eine Nachricht von 4 nach 6 übertragen. Die direkte Übertragung hätte Wahrscheinlichkeit $p(4,6) = 0{,}2$, über den Knoten 5 wäre die Wahrscheinlichkeit $0{,}6 \cdot 0{,}7 = 0{,}42$.

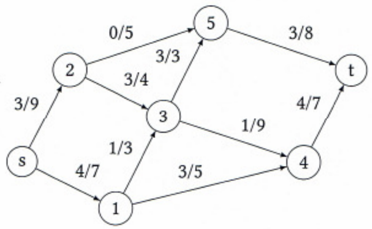
Sei s der Startknoten, d ein Feld der Größe $|V|$. Gegeben Sei folgender Algorithmus:

```
for all ( v ∈ V )
    d[v] = 0
d[s] = 1
for ( i=1; i < |V|; ++i )
    for all ( (u,v) ∈ E )
        if ( d[v] < d[u] · p(u,v) )
            d[v] = d[u] · p(u,v)
```

a) (2P) Sei $s = 1$ der Startknoten. Notieren Sie einmalig die Reihenfolge, in der bei Ihnen die innere Schleife die Kanten abläuft und bestimmen Sie das Feld d mit dem Algorithmus. Geben Sie dabei den Inhalt des Feldes d zu Beginn und nach jeder Iteration der äußeren Schleife an.
b) (2P) Sei $P(s,v)$ ein sicherster Weg von s nach $v \in V$, also ein Weg, der unter allen Wegen von s nach v die höchste Wahrscheinlichkeit hat. Begründen Sie, weshalb alle Teilwege von P auch sicherste Wege sind.
c) (3P) Zeigen Sie, dass der Algorithmus in $d[v]$ die Wahrscheinlichkeit für eine Übertragung auf dem sichersten Weg von s nach v speichert.
d) (1P) Weshalb kann ein sicherster Weg keine Zyklen haben?
e) (1P) Wie könnte man den sichersten Weg schneller bestimmen?

Aufgabe 8 Fluss maximieren (9 Punkte)

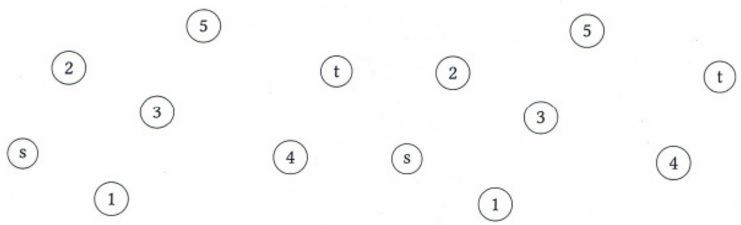
Gegeben Sei folgender Graph $G = (V,E)$, sowie eine Kapazitätsfunktion $c : E \to \mathbb{N}$ und ein Fluss f, deren Werte jeweils in der Form f/c an den Kanten stehen.



a) (1P) Bestimmen Sie das Restnetzwerk $G_f(V,E_f)$.
b) (2P) Bestimmen Sie das geschichtete Restnetzwerk $G_f^L(V,L_f)$ mit $L_f = \{(u,v) \in E_f : \delta(s,v) = \delta(s,u) + 1\}$.
Dabei ist δ, die tatsächliche Länge des kürzesten Weges.
c) (3) Bestimmen Sie einen blockierenden Fluss f_B in G_f^L mit der Methode von Dinic. Wege dürfen Sie durch "hinsehen" bestimmen.
d) (1P) Augmentieren Sie den Fluss in G mit Hilfe des gefundenen blockierenden Flusses f_B.
e) (1P) Ist f nach der Augmentierung maximal? (Begründung!)
f) (1P) Finden Sie einen minimalen S-T-Schnitt.

Anbei finden Sie zur Erleichterung vorgezeichnete Knoten in mehrfacher Ausführung, bei G_f^L werden Sie auch nicht alle Knoten benötigen.

zu a):



Aufgabe 9 Terminplanung (6 Punkte)

Auf dem Campus gibt es im Studentenwohnheim eine internationale WG ausländischer Studierender. In ihr wohnen n Studierende verschiedener Nationen und anläßlich des neuen Semesters haben sie beschlossen, Gerichte aus allen Herkunftsländern auszuprobieren. In den folgenden n Tagen soll jeder Studierende genau einmal für die anderen kochen, wobei es bei der Planung aber noch andere Termine zu berücksichtigen gibt. Wenn wir die Studierenden mit $S := \{s_1, \ldots, s_n\}$ bezeichnen und die Termine mit $T := \{t_1, \ldots, t_n\}$, dann gibt es für jeden Studierenden s_i eine Menge $F_i \subset T$ von Terminen, an denen s_i keine Zeit zum Kochen hat. Wir suchen nun einen möglichen Essensplan p, also eine Zuordnung von Studierenden zu Terminen, so dass für keinen Studierenden ein Terminkonflikt entsteht, also für alle s_i gilt $p(s_i) \notin F_i$.

a) (3P) Formulieren Sie das Problem als Matching, d.h. skizzieren und beschreiben Sie den Graph und wie man ein Matching dann als Lösung des Problems verwenden kann.
b) (3P) Anna, eine der Bewohnerinnen der WG, hat schon einen Essensplan ausgearbeitet. Leider gibt es bei diesem Plan das Problem, dass zwei Studierende s_i und s_j beide für den Tag t_k eingeteilt wurden und am Tag t_ℓ niemand kocht, während der Rest des Plans in Ordnung ist.
Wie können Sie aus diesem "fast möglichen" Essensplan in Zeit $O(n^2)$ einen möglichen Essenplan erstellen, bzw. feststellen, falls es keinen möglichen gibt?