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

Multiple-Choice-Test zu Grundlagen der Algorithmik (A)
TU Berlin, 26.05.2018

(Niedermeier/Bentert, Sommersemester 2018)

Arbeitszeit: 20 Minuten, Gesamtpunktzahl: 25

Hinweis: Je Aufgabe ist **mindestens** eine Antwortmöglichkeit korrekt.
Sobald eine **falsche** Antwortmöglichkeit angekreuzt wurde, gibt es **Null** Punkte für die betroffene Aufgabe.
Viel Erfolg!

Aufgabe 1: **Stable Matching** (5 Punkte)

Gegeben seien die Mengen $M = \{A, B, C\}$ und $W = \{X, Y, Z\}$ mit folgenden Präferenzen:

$$A : X < Y < Z, \qquad X : C < A < B,$$
$$B : X < Y < Z, \qquad Y : B < C < A,$$
$$C : X < Y < Z, \qquad Z : A < B < C.$$

Dabei bedeutet z.B. „$X : A < B < C$“, dass X findet, dass A besser als B ist und B besser als C ist. Welche der folgenden Aussagen sind wahr? (*Erinnerung*: Stabil heißt, dass es kein Paar $m \in M$ und $w \in W$ gibt, sodass sich m und w jeweils den ihnen zugewiesenen Partnern vorziehen.)

- [ ] Die Zuordnung $(X, C), (Y, A), (Z, B)$ ist stabil.
- [X] Die Zuordnung $(X, C), (Y, B), (Z, A)$ ist stabil.
- [X] Die Zuordnung $(X, C), (Y, A), (Z, B)$ ist nicht stabil.
- [ ] Die Zuordnung $(X, C), (Y, B), (Z, A)$ ist nicht stabil.

Aufgabe 2: **Pseudocode** (4 Punkte)

Welches Problem wird von folgendem Algorithmus gelöst? (Genau eine Antwort ist korrekt!)

Hinweis: Die Problemkandidaten sind wie folgt definiert (die Eingabe ist jeweils ein ungerichteter Graph $G = (V, E)$ und eine natürliche Zahl $k > 0$). Die Frage ist jedes Mal ob eine Knotenmenge S der Größe k von G existiert, sodass die folgenden problemspezifischen Kriterien erfüllt sind.

Problemname	Kriterium
k-Clique	Alle Knoten in S sind in G paarweise untereinander mit einer Kante verbunden (sind adjazent).
k-Independent Set	Der von S induzierte Subgraph enthält keine Kante.
k-Path	Der von S induzierte Subgraph ist ein Pfad, d. h. S besteht aus k Knoten $\{v_1, v_2, \dots, v_k\}$ und genau den Kanten $\{\{v_i, v_{i+1}\} \mid 1 \leq i < k\}$.
k-Cycle	Der von S induzierte Subgraph ist ein Kreis, d. h. S ist zusammenhängend und alle Knoten haben Grad 2.

Hinweise:

Ein von einer Knotenmenge S induzierter Subgraph von G ist ein Graph, der aus den Knoten in S und allen Kanten aus G, die beide Endpunkte in S haben, besteht.

Die Funktion $\deg_S(v)$ gibt die Anzahl der Nachbarn von v in S an.

```
input : Ein ungerichteter Graph G = (V, E) und eine natürliche Zahl k > 0.
foreach S ⊆ V mit |S| = k do
    if inducedSubgraph(S) ist zusammenhängend then
        foreach u ∈ S do
            foreach v ∈ S \ {u} do
                if deg_S(u) = 1 and deg_S(v) = 1 then
                    answer ← true
                    foreach w ∈ S \ {u, v} do
                        if deg_S(w) ≠ 2 then answer ← false
                    if answer = true then return true
return false
```

- [ ] k-Independent Set
- [X] k-Path
- [ ] k-Cycle
- [ ] k-Clique

Aufgabe 3: **Laufzeitanalyse** (6 Punkte)

Welche der folgenden Angaben sind korrekte Laufzeitabschätzungen des Algorithmus aus der vorherigen Aufgabe? (Es gilt: $n = |V|$ und $m = |E|$.)

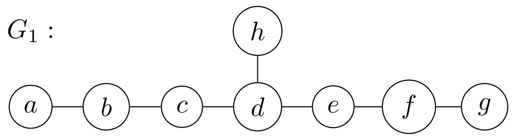
Hinweis: Es können mehrere Antworten korrekt sein.

- [ ] $O(n + m)$
- [X] $O(n^{k+3})$
- [X] $O(2^n \cdot k^3)$
- [ ] $O(n \log n)$
- [X] $O(2^n \cdot n^3)$
- [X] $O(n^k)$

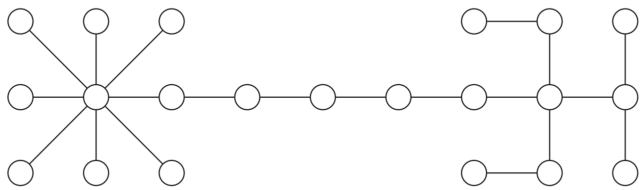
Aufgabe 4: Beispiel für 4-Club auf Bäumen (4 Punkte)

Gegeben ist ein Baum $G = (V, E)$. (Ein Baum ist ein zusammenhängender kreisfreier Graph.) Die Aufgabe besteht darin, eine möglichst große Menge S von Knoten zu finden, sodass S ein 4-Club induziert, d.h. der von S induzierte Graph Durchmesser vier hat. Der Durchmesser eines Graphen ist die maximale Anzahl von Kanten eines kürzesten Pfades zwischen zwei Knoten. Der Durchmesser des folgenden Graphen G_1 ist sechs, da der kürzeste Pfad zwischen a und g aus sechs Kanten besteht.

G_1 :



Was ist in folgendem Beispiel die Größe des größtmöglichen 4-Clubs?



- [ ] 9
- [ ] 10
- [X] 11
- [ ] 21

Hinweis: Ein von einer Knotenmenge S induzierter Subgraph von G ist ein Graph, der aus den Knoten in S und allen Kanten aus G, die beide Endpunkte in S haben, besteht.

Aufgabe 5: Algorithmus für 4-Club auf Bäumen (5 Punkte)

Vervollständigen Sie folgenden Greedy-Algorithmus für 4-Club auf Bäumen, indem Sie die korrekten Lückenfüller auswählen. Die Funktionen $N_S(v)$ und $\deg_S(v)$ geben die Menge und die Anzahl der Nachbarn von v in S an. Es gilt $N_{G_1}(d) = \{c, h, e\}$ und $\deg_{G_1}(d) = 3$.

Input: Ein ungerichteter Baum $G = (V, E)$.
Output: Die Größe des größten 4-Clubs in G.

```
answer ← 0
foreach v ∈ V do
    c ← 0
    foreach w ∈ N_G(v) do
        c ← c+ [ A ].
    if c > answer then answer ← [ B ]
return answer + 1
```

- [X] A: $\deg_G(w)$ — B: c
- [ ] A: $\deg_G(w) - 1$ — B: $c + \deg_G(v)$
- [ ] A: $\deg_G(w)$ — B: $c + \deg_G(v)$
- [ ] A: $\deg_G(w) - 1$ — B: c