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

Multiple-Choice-Test zu Algorithmentheorie
TU Berlin, 25.07.2020

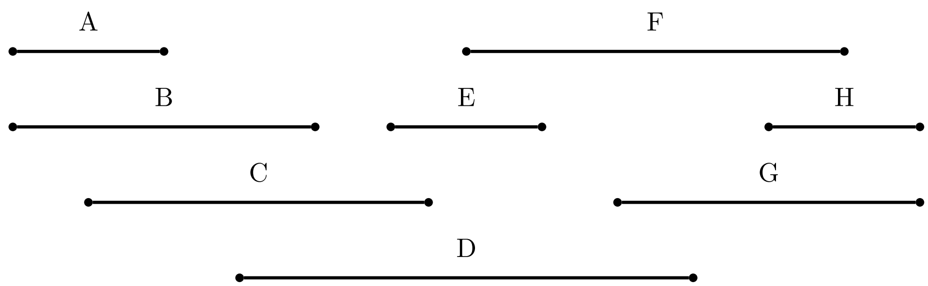
(Niedermeier/Nichterlein/Bentert/Heeger, Sommersemester 2020)

Arbeitszeit: 45 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: **Intervall Scheduling** (3 Punkte)

Betrachten Sie folgende Instanz von Intervall Scheduling.



Welche der folgenden Aussagen sind korrekt?

- [ ] Jede optimale Lösung enthält das Intervall E.
- [X] Jede optimale Lösung enthält das Intervall D oder das Intervall E (oder beide).
- [ ] Es gibt eine optimale Lösung, die Intervall D und Intervall E enthält.
- [X] Jede optimale Lösung enthält genau drei Intervalle.
- [ ] Jede optimale Lösung enthält genau vier Intervalle.
- [X] Es gibt mindestens fünf optimale Lösungen.

Aufgabe 2: **Huffman-Kodierung** (2 Punkte)

Wieviele Bits werden benötigt, um das Wort *mississippi* mit der Huffman-Kodierung und den folgenden Häufigkeiten h der Buchstaben zu kodieren?

$h(m) = 1/11, h(i) = 4/11, h(s) = 4/11$ und $h(p) = 2/11$

[ ] 20 [X] 21 [ ] 19 [ ] 11

Aufgabe 3: **Zählen von Inversionen** (3 Punkte)

Bestimme die Anzahl an Inversionen von

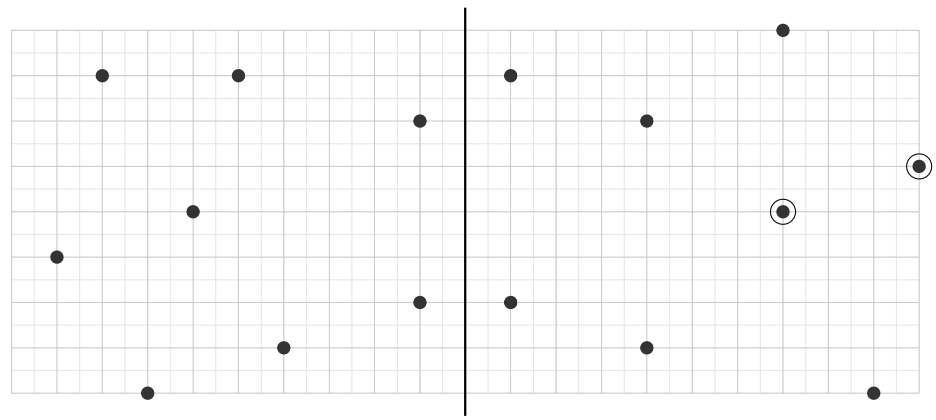
$$1,\ 4,\ 2,\ 6,\ 5,\ 8,\ 10,\ 13,\ 11,\ 9,\ 12,\ 14,\ 15,\ 3,\ 7,\ 16$$

im Vergleich zur Ordnung $1, 2, \ldots, 16$.

[ ] 16 [ ] 21 [X] 26 [ ] 31

Aufgabe 4: **Closest Pair** (2+1 Punkte)

Betrachte die folgende Instanz von Closest Pair. Die vertikale Linie symbolisiert die Aufteilung der Punkte, die von dem Divide and Conquer-Algorithmus im ersten Divide-Schritt vorgenommen wird. Beide Teile werden rekursiv gelöst.



(a) Auf der rechten Seite wurden die beiden Punkte mit minimaler Distanz markiert. Markieren Sie auf linken Seite die beiden Punkte mit minimaler Distanz.

(b) Wie viele Punktpaare im "Grenzstreifen" müssen verglichen werden?

☐ 1 ☒ 2 ☐ 3 ☐ 4

Aufgabe 5: **Repräsentative Probleme** (3 Punkte)

Ein Unternehmen vermietet einen Sitzungssaal. Das Unternehmen hat mehrere Anfragen für Buchungen des Saals für verschiedene Zeiträume. Der Gewinn, den das Unternehmen mit einer Buchung macht, ist dabei direkt proportional zur Dauer der Buchung.

Welches Problem muss das Unternehmen lösen, um seinen Gewinn zu maximieren? (Genau eine Antwort ist korrekt!)

☐ Independent Set
☐ Competitive Facility Location
☐ Bipartites Matching
☒ Gewichtetes Intervall Scheduling
☐ Intervall Scheduling

Hinweis:

Problemname	Aufgabe
Bipartites Matching	Finde größtmögliche „unabhängige“ (keine gemeinsamen Endpunkte) Kantenmenge in einem bipartiten Graphen.
Independent Set	Finde größtmögliche „unabhängige“ (paarweise nicht benachbart) Knotenmenge in einem Graphen.
Intervall Scheduling	Gegeben eine Menge von Intervallen, finde größtmögliche Menge nichtüberlappender Intervalle.
Gewichtetes Intervall Scheduling	Gegeben eine Menge von Intervallen mit positiven Gewichten, finde eine Menge nichtüberlappender Intervalle mit größtmöglichen Gewicht.
Competitive Facility Location	Zwei Spielerinnen wählen abwechselnd Knoten aus einem gegebenen knotengewichteten Graphen und jeder gewählte Knoten wird samt seiner Nachbarn gelöscht. Spielerin 1 beginnt und will erreichen, dass Spielerin 2 so wenig Punkte wie möglich bekommt. Jede Spielerin bekommt Punkte entsprechend des Gewichts des gewählten Knoten.

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

Welche der folgenden Aussagen sind für folgenden Algorithmus korrekt?

```
Data: Ein Graph G = (V, E) und eine natürliche Zahl k.
Result: Ja oder Nein.
foreach V' ⊆ V with |V'| = k do
    s ← 0
    foreach v ∈ V' do
        foreach w ∈ V \ V' do
            if {v, w} ∈ E then
                s ← s + 2
        foreach u ∈ V' \ {v} do
            if {v, u} ∈ E then
                s ← s + 1
    if s = 2|E| then
        return Ja
return Nein
```

- [ ] Der Algorithmus löst INDEPENDENT SET.
- [X] Der Algorithmus löst VERTEX COVER.
- [ ] Die Laufzeit des Algorithmus ist in $O(|V|^5)$.
- [ ] Die Laufzeit des Algorithmus ist in $O(2^k \cdot k^3)$.
- [X] Die Laufzeit des Algorithmus ist in $O(2^{|V|} \cdot |V|^3)$.
- [X] Die Laufzeit des Algorithmus ist in $O(2^{|V|} \cdot k \cdot |V|)$.

Hinweise: Sie können davon ausgehen, dass eine Abfrage $\{v, u\} \in E$ in konstanter Zeit beantwortet werden kann. Bei INDEPENDENT SET wird in einem Graph nach k Knoten gesucht, die paarweise nicht durch eine Kante verbunden sind. Bei VERTEX COVER wird in einem Graph nach k Knoten gesucht, sodass jede Kante im Graph mindestens einen dieser k Knoten als Endpunkt hat.

Aufgabe 7: **Stable Matching** (4 Punkte)

Gegeben seien die Mengen $\{A, B, C\}$ der Männer und $\{X, Y, Z\}$ der Frauen mit folgenden Präferenzen:

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

Dabei bedeutet z.B. „ $A : Z < X < Y$ “, dass A findet, dass Z besser als X ist und X besser als Y ist. Welche der folgenden Aussagen sind wahr?

- [ ] Die Zuordnung $(A, Z), (B, X), (C, Y)$ ist stabil.
- [X] Die Zuordnung $(A, Y), (B, X), (C, Z)$ ist stabil.
- [ ] Im Allgemeinen gilt, dass das vom Propose&Reject-Algorithmus berechnete Matching von der Reihenfolge der Proposals abhängen kann.
- [X] In der frauenoptimalen Zuordnung kann eine Frau dem Mann zugewiesen werden, den sie am schlechtesten findet.

Erinnerung: Eine Zuordnung heißt stabil, wenn es kein Paar (M, W) gibt, sodass sich sowohl M als auch W gegenüber dem in der gegebenen Zuordnung zugeteilten Partner bevorzugen.

Aufgabe 8: **Algorithmenanalyse** (4 Punkte)

Wir erinnern an das Master-Theorem aus der Vorlesung:

Satz 1 (Master Theorem)**.** *Sei $T(n) = aT(\lceil n/b \rceil) + O(n^d)$ für Konstanten $a > 0, b > 1$ und $d \geq 0$, dann ist*

$$T(n) = \begin{cases} O(n^d) & \text{falls } d > \log_b a, \\ O(n^d \log n) & \text{falls } d = \log_b a, \\ O(n^{\log_b a}) & \text{falls } d < \log_b a. \end{cases}$$

Wir nennen eine Matrix $A \in \mathbb{N}^{n \times n}$ *sortiert* wenn sowohl jede Zeile als auch jede Spalte der Matrix aufsteigend sortiert ist, d.h., für $1 \leq i, j \leq n$ gilt, dass $a_{i',j} \leq a_{i,j}$ für alle $i' \leq i$, und dass $a_{i,j'} \leq a_{i,j}$ für alle $j' \leq j$.

Folgender Algorithmus bestimmt ob eine sortierte Matrix A eine Zahl $x \in \mathbb{N}$ enthält:

Algorithm 1: Pseudocode der Funktion `find`(A, x).

1 **Eingabe:** Eine sortierte Matrix $A \in \mathbb{N}^{n \times n}$, eine Zahl $x \in \mathbb{N}$.
2 **Ausgabe:** Wahr oder Falsch.
3 $i \leftarrow \lfloor n/2 \rfloor$
4 $A_{1,1} \leftarrow$ die Teilmatrix von A mit Zeilen $1, \ldots, i$ und Spalten $1, \ldots, i$
5 $A_{1,2} \leftarrow$ die Teilmatrix von A mit Zeilen $1, \ldots, i$ und Spalten $i+1, \ldots, n$
6 $A_{2,1} \leftarrow$ die Teilmatrix von A mit Zeilen $i+1, \ldots, n$ und Spalten $1, \ldots, i$
7 $A_{2,2} \leftarrow$ die Teilmatrix von A mit Zeilen $i+1, \ldots, n$ und Spalten $i+1, \ldots, n$
8 **if** $n = 0$ **then return** Falsch.
9 **if** $a_{i,i} = x$ **then return** Wahr.
10 **else if** $a_{i,i} > x$ **then return** `find`$(A_{1,1}) \vee$ `find`$(A_{1,2}) \vee$ `find`$(A_{2,1})$
11 **else if** $a_{i,i} < x$ **then return** `find`$(A_{1,2}) \vee$ `find`$(A_{2,1}) \vee$ `find`$(A_{2,2})$

Welche der folgenden oberen Schranken für den Algorithmus sind korrekt?

☐ $O(\log n)$	☐ $O(n \log n)$	☐ $O(n^{\log_3 2})$
☐ $O(n)$	X $O(n^{\log_2 3})$	X $O(n^2)$

Hinweise:

Es kann angenommen werden, dass die Zeilen 3-9 in `find` in $O(1)$ Zeitschritten ausgeführt werden können. Es können mehrere Antwortmöglichkeiten korrekt sein. Es gibt für dieses Problem noch einen besseren Algorithmus, dieser ist für diese Aufgabe jedoch irrelevant.