# Algorithmen und Datenstrukturen 2

WiSe 2022/2023 · Prof. Dr. M. Hoefer
21. Februar 2023 (Erstklausur)
Klausurnummer: **0000**

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

**Nachname** (Druckschrift): ____________________

**Vorname** (Druckschrift): ____________________

**Matrikelnummer:** ____________________

**Studiengang:** ____________________

Beachten Sie die folgenden Hinweise:

- Schreiben Sie Ihre persönlichen Daten **nur auf dieses Titelblatt**. **Merken oder notieren** Sie sich Ihre Klausurnummer **0000**, da nur unter dieser Nummer die Ergebnisse veröffentlicht werden.
- Legen Sie Ihre **Goethe-Card** deutlich sichtbar an Ihren Platz, damit wir während der Klausur Ihre Identität überprüfen können.
- Überprüfen Sie, ob die Klausur aus **?** durchnummerierten **Vorder- und Rückseiten** besteht. Beachten Sie, dass die Aufgabenstellungen sowohl auf den Vorder- als auch auf den Rückseiten stehen.
- Sie dürfen nur **dokumentenechte Stifte** in den Farben blau/schwarz zum Ausfüllen verwenden. Insbesondere ist die Nutzung von Tintenlöschern und Tipp-Ex untersagt.
- **Zugelassene Hilfsmittel:**
  **1 Blatt DIN A4 mit handschriftlichen Notizen (beidseitig).**
  Das Mitbringen nicht zugelassener Hilfsmittel stellt eine Täuschung dar und führt zum Nichtbestehen der Klausur. **Schalten Sie daher bitte alle elektronischen Geräte, insbesondere Handys und Smartwatches, vor Beginn der Klausur aus.**
- Sie müssen **alle Klausurunterlagen** (v.a. die Klausur, Zusatzpapier, DIN A4-Blatt mit handschriftlichen Notizen) nach der Bearbeitungszeit abgeben.
- Sollte der Platz unter einer Aufgabe nicht ausreichen, **nutzen Sie bitte die Zusatzblätter am Ende**. Weitere Zusatzblätter sind auf Nachfrage erhältlich. Vermerken Sie auf lose Zusatzblätter unbedingt Ihre Klausurnummer!
- Wenn sich Ihre Lösung zu einer Aufgabe teilweise oder ganz auf Zusatzblättern befindet, vermerken Sie dies entsprechend bei der Aufgabe und auf dem Zusatzblatt.
- Werden zu einer Aufgabe zwei oder mehr Lösungen angegeben, so gilt die Aufgabe als nicht gelöst. Entscheiden Sie sich also immer für **eine** Lösung. Begründungen sind nur dann notwendig, wenn die Aufgabenformulierung dies explizit verlangt.
- Die Klausur ist mit Sicherheit bestanden, wenn mindestens **50%** der Höchstpunktzahl erreicht werden (ohne Berücksichtigung der Bonifikation aus den Übungen).
  Die Bearbeitungszeit beträgt **180 Minuten**.

**Viel Erfolg!**



1001222100003

a) Betrachten Sie eine Turingmaschine $T$ mit Zustandsmenge $Q = \{q_0, q_1, q_2\}$, Eingabealphabet $\Sigma = \{0, 1\}$, Startzustand $q_0$, Arbeitsalphabet $\Gamma = \{0, 1, \mathtt{B}\}$ und Menge $F = \{q_2\}$ aller akzeptierenden Zustände. Für $x \in \{0, 1\}$ ist die Zustandsüberführungsfunktion $\delta$ gegeben durch

$$\begin{array}{ll} \delta(q_0, \mathtt{B}) = \perp & \delta(q_0, x) = (q_1, x, \mathtt{rechts}) \\ \delta(q_2, \mathtt{B}) = \perp & \delta(q_1, x) = (q_1, x, \mathtt{rechts}) \\ \delta(q_2, x) = \perp & \delta(q_1, \mathtt{B}) = (q_2, \mathtt{B}, \mathtt{links}) \end{array}$$

i) **(1 Punkt)** Welche der folgenden Eingaben wird von $T$ *nicht* akzeptiert?

| | |
|---|---|
| ☐ 00110 | ☐ 1 |
| ☐ $\epsilon$ | ☐ 11111 |

ii) **(1 Punkt)** Die Eingabe $w = w_1 \cdots w_n$ sei in den Zellen $1, ..., n$ von $T$ gespeichert. Über welcher Zelle befindet sich der Lesekopf von $T$ am Ende der Berechnung?

Zelle: ____________________

⇑⇑⇑ ______ / 2 Punkt(e) ⇑⇑⇑

b) Betrachten Sie das Lastverteilungsproblem mit $m = 2$ Prozessoren und $n = 5$ Aufgaben mit Rechenzeiten

$$t_1 = 2, \;\; t_2 = 4, \;\; t_3 = 5, \;\; t_4 = 6, \;\; t_5 = 7 \;.$$

i) **(1 Punkt)** Geben Sie eine Aufteilung der Aufgaben auf die Prozessoren an, so dass der resultierende Makespan *minimal* ist.

Aufgaben für Prozessor 1: ________________________

Aufgaben für Prozessor 2: ________________________

ii) **(1 Punkt)** Geben Sie den resultierenden Makespan an, wenn die Aufgaben mit der *On-line* Heuristik aus der Vorlesung zugeteilt werden.

Makespan: ____________________

iii) **(1 Punkt)** Geben Sie den resultierenden Makespan an, wenn die Aufgaben mit der *Off-line* Heuristik aus der Vorlesung zugeteilt werden.

Makespan: ____________________

$\Uparrow\Uparrow\Uparrow$ ______ / 3 Punkt(e) $\Uparrow\Uparrow\Uparrow$

c) Sei $A$ ein $\mathcal{NP}$-vollständiges Entscheidungsproblem.

Welche der folgenden Aussagen ist korrekt?

☐ Es gilt $L \leq_p A$ für alle $\mathcal{NP}$-harten Probleme $L$.

☐ Für alle Sprachen $L \in \mathcal{P}$ gilt $L \leq_p A$.

☐ Falls $A \leq_p B$ für ein Problem $B$, dann ist $B$ ebenfalls $\mathcal{NP}$-vollständig.

☐ Wenn $A \subseteq B$ für eine Sprache $B \in \mathcal{P}$, dann gilt auch $A \in \mathcal{P}$.

$\Uparrow\Uparrow\Uparrow$ ______ / 2 Punkt(e) $\Uparrow\Uparrow\Uparrow$

a) Sei `A[1...n]` ein Array mit `A[`$i$`]` $= i \cdot (n - i)$ für $i \in \{1, \ldots, n\}$.

Geben Sie jeweils asymptotisch exakt in Abhängigkeit von $n$ an, welche Laufzeit der angegebene Sortieralgorithmus auf dem Array `A` hat.

Bubble Sort: $\Theta\left(\underline{\hspace{4cm}}\right)$

Selection Sort: $\Theta\left(\underline{\hspace{4cm}}\right)$

Mergesort: $\Theta\left(\underline{\hspace{4cm}}\right)$

Radix Sort mit Basis $b = n$: $\Theta\left(\underline{\hspace{4cm}}\right)$

⇑⇑⇑ ______ / 4 Punkt(e) ⇑⇑⇑

b) Gegeben sei das Array

`A` $=$ 2 , 1 , 4 , 3 , 6 , 5 , 8 , 7

i) **(2 Punkte)** Nehmen Sie an, `A` wird mit *Bubble Sort* sortiert. Geben Sie das Array `A` nach Ende der ersten Phase an.

Array `A`: ________________________

ii) **(2 Punkte)** Nehmen Sie an, `A` wird mit *Mergesort* sortiert. Geben Sie an, wie oft die Funktion `merge(...)` aufgerufen wird.

Anzahl Aufrufe: ________________

iii) **(2 Punkte)** Nehmen Sie an, `A` wird mit *Quicksort* mit `pivot(links,rechts)=rechts` sortiert. Geben Sie an, wie oft die Funktion `partition(...)` aufgerufen wird.

Anzahl Aufrufe: ________________

⇑⇑⇑ _____ / 6 Punkt(e) ⇑⇑⇑

c) Ist die folgende Aussage korrekt?

*Für jedes vergleichsorientierte Sortierverfahren gibt es Eingaben der Länge 5, so dass das Verfahren mindestens 7 Vergleiche benötigt, um die Eingabe zu sortieren.*

Begründen Sie Ihre Antwort.

⇑⇑⇑ _____ / 2 Punkt(e) ⇑⇑⇑


1 0 0 7 2 2 2 1 0 0 0 0 5

## Aufgabe 3: Sortieren II

7 Punkt(e)

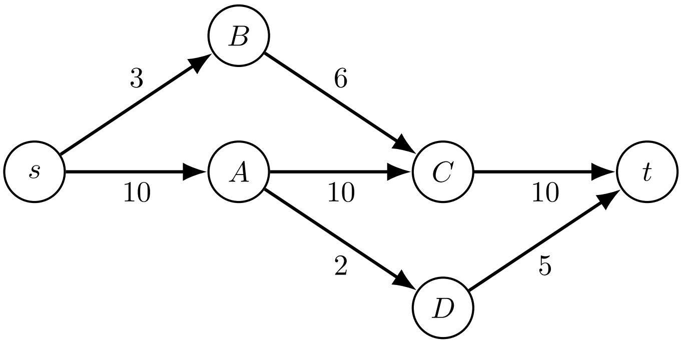
Gegeben seien zwei Arrays `A[1...n]` und `B[1...n]` von natürlichen Zahlen. Die Zahlen in `A` sind aus der Menge $\{1, 2, \ldots, n^3\}$, die Zahlen in `B` sind aus der Menge $\left\{2^i \mid i \in \{0, 1, \ldots, n\}\right\}$.

Beschreiben Sie ein Verfahren, welches ein aufsteigend sortiertes Array `C[1...2n]` der Zahlen aus `A` und `B` berechnet. Die asymptotische worst-case Laufzeit soll $\mathcal{O}(n)$ nicht überschreiten. Sie dürfen alle aus der Vorlesung bekannten Verfahren und deren Eigenschaften ohne weitere Erläuterung verwenden.

Begründen Sie auch, warum Ihr Verfahren die Laufzeitschranke einhält.

⇑⇑⇑ ______ / 7 Punkt(e) ⇑⇑⇑

Betrachten Sie das folgende Flussnetzwerk $G = (V, E, c, s, t)$ mit $V = \{A, B, C, D, s, t\}$:




Die Kantenbeschriftungen geben für jede Kante $e \in E$ ihre Kapazität $c(e)$ an.

a) Berechnen Sie mittels des Ford-Fulkerson Algorithmus einen maximalen Fluss in $G$.

Wählen Sie dafür in jeder Runde $i$ des Algorithmus einen augmentierenden Pfad $p_i$ und erhöhen Sie den Fluss entlang dieses Pfades so viel wie möglich. Tragen Sie die von Ihnen gewählten augmentierenden Pfade zusammen mit dem Wert $f_i$ des $s$-$t$-Flusses am Ende jeder Runde in die nachfolgende Tabelle ein (eine Runde entspricht einer Zeile).

| Runde $i$ | Augmentierender Pfad $p_i$ | Flusswert $f_i$ |
|---|---|---|
| 1 | | |
| | | |
| | | |
| | | |

⇑⇑⇑ ______ / 9 Punkt(e) ⇑⇑⇑



1 009222 1 00009

b) Zeichnen Sie ein Flussnetzwerk $H$, für das der minimale $s$-$t$-Schnitt nicht eindeutig ist.

Geben Sie auSSerdem zwei verschiedene minimale $s$-$t$-Schnitte in $H$ an.

⇑⇑⇑ ______ / 3 Punkt(e) ⇑⇑⇑

c) Sei nun $G = (V, E, c, s, t)$ ein beliebiges Flussnetzwerk.

Beweisen Sie die folgende Aussage:

*Wenn* $(S_1, V \setminus S_1)$ *und* $(S_2, V \setminus S_2)$ *zwei minimale s-t-Schnitte sind, dann ist* $(S, V \setminus S)$ *mit* $S = S_1 \cap S_2$ *ebenfalls ein minimaler s-t-Schnitt.*

*Hinweis: Max-Flow-Min-Cut Theorem.*

⇑⇑⇑ ______ / 2 Punkt(e) ⇑⇑⇑



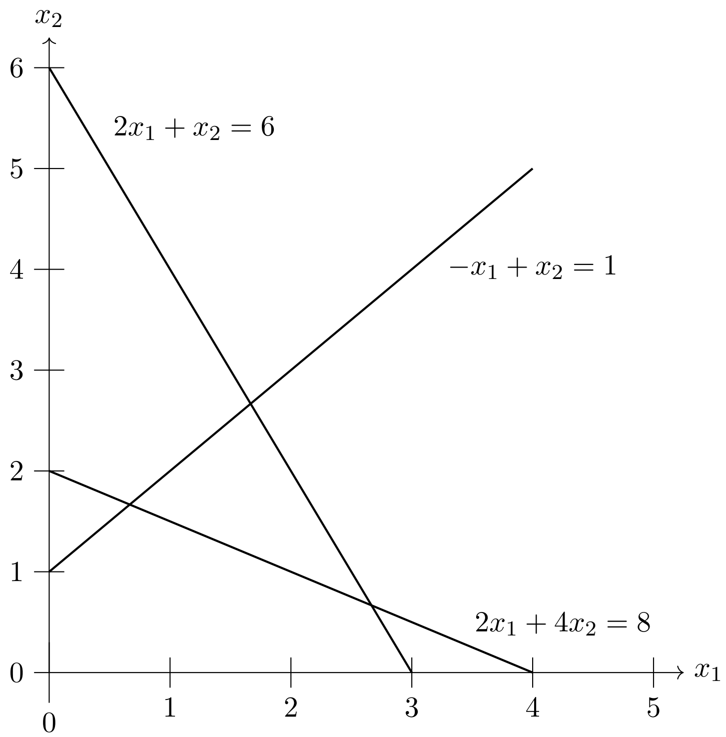
1 010222 100005

a) Betrachten Sie das folgende lineare Programm:

$$\begin{array}{rrcl} \text{Max.} & 12x_1 + 13x_2 & & \\ \text{s.d.} & 2x_1 + 4x_2 & \leq & 8 \\ & 2x_1 + x_2 & \leq & 6 \\ & -x_1 + x_2 & \leq & 1 \\ & x_1, x_2 & \geq & 0 \end{array}$$

i) **(2 Punkte)** Schraffieren Sie in der nachfolgenden Abbildung das Lösungspolytop.

ii) **(1 Punkt)** Markieren Sie in derselben Abbildung eine optimale Lösung.




⇑⇑⇑ ______ / 3 Punkt(e) ⇑⇑⇑

b) Gegeben sei das folgende lineare Programm (P):

$$\begin{array}{rrcl} \text{Max.} & 7x_1 - 14x_2 & & \\ \text{s.d.} & x_1 - 4x_2 - 2x_3 & \leq & 1 \\ & 2x_1 + 2x_2 + 3x_3 & \leq & 9 \\ & x_1, x_2, x_3 & \geq & 0 \end{array} \qquad \text{(P)}$$

i) **(3 Punkte)** Bestimmen Sie das zu (P) duale lineare Programm (D). Gehen Sie dabei nach dem Rezept aus der Vorlesung vor.

ii) **(3 Punkte)** Seien $(\boldsymbol{x}^*)^T = (3, 0, 1)$ und $(\boldsymbol{y}^*)^T = (3, 2)$.

Zeigen Sie, dass $\boldsymbol{x}^*$ eine optimale Lösung für (P) und $\boldsymbol{y}^*$ eine optimale Lösung für (D) ist.

⇑⇑⇑ ______ / 6 Punkt(e) ⇑⇑⇑

## Aufgabe 6: P und NP

$4 + 4 + 4 = 12$ Punkt(e)

Die Sprache `Trio` sei wie folgt definiert:

$$\texttt{Trio} := \{G \mid G \text{ ist ein ungerichteter Graph und enthält eine Clique der GröSSe 3}\}$$

a) Zeigen Sie, dass `Trio` $\in \mathcal{P}$.

$\Uparrow\Uparrow\Uparrow$ ______ / 4 Punkt(e) $\Uparrow\Uparrow\Uparrow$

b) Ist die folgende Aussage korrekt?

*Wenn* $L \leq_p$ `Trio` *für eine Sprache* $L$ *gilt, dann ist* $L \in \mathcal{P}$.

Begründen Sie Ihre Antwort.

*Hinweis: Sie dürfen die Aussage in a) hier auch ohne Beweis verwenden.*

$\Uparrow\Uparrow\Uparrow$ ______ / 4 Punkt(e) $\Uparrow\Uparrow\Uparrow$

c) Ist die folgende Aussage korrekt?

$$\textit{Wenn } \mathcal{P} \neq \mathcal{NP}, \textit{ dann gilt nicht } \texttt{KNF-SAT} \leq_p \texttt{Trio}.$$

Begründen Sie Ihre Antwort.

*Hinweis: Sie dürfen die Aussage in a) hier auch ohne Beweis verwenden.*

⇑⇑⇑ ______ / 4 Punkt(e) ⇑⇑⇑



1 0 1 4 2 2 2 1 0 0 0 0 3

Ein Flugzeughersteller hat eine Menge $T$ neuer Flugzeugtypen konstruiert, die im Rahmen eines Testprogramms getestet werden sollen. Der Hersteller verfügt über $n$ Piloten, wobei Pilot $i \in \{1, \ldots, n\}$ die Flugzeugtypen in der Menge $F_i \subseteq T$ fliegen kann. Zur Durchführung des Testprogramms soll ein möglichst kleines Team von Piloten zusammengestellt werden, so dass für jeden Flugzeugtyp mindestens *zwei* Piloten im Team sind, die diesen fliegen können.

Formal wird die Sprache `Flugzeugtest` wie folgt definiert:

$$\texttt{Flugzeugtest} := \Big\{(T, F_1, \ldots, F_n, k) \mid \text{es gibt } I \subseteq \{1, \ldots, n\} \text{ mit } |I| = k \text{ und für alle } t \in T \text{ gibt es mindestens zwei } i \in I \text{ mit } t \in F_i \Big\}$$

a) Betrachten Sie folgende Instanz des Problems. Die Flugzeugtypen sind:

$$T = \Big\{\text{AeroTuring, JetFord, SkyMonds, AirKarp}\Big\}$$

Es stehen vier Piloten zur Wahl, die folgende Flugzeugtypen fliegen können:

$$\begin{aligned}
F_1 &= \{\text{AeroTuring, SkyMonds, AirKarp}\},\\
F_2 &= \{\text{JetFord, SkyMonds}\},\\
F_3 &= \{\text{JetFord, AirKarp}\},\\
F_4 &= \{\text{AeroTuring, JetFord, AirKarp}\}
\end{aligned}$$

Geben Sie eine minimale Auswahl $I \subseteq \{1, 2, 3, 4\}$ von Piloten an, so dass für jeden Flugzeugtyp mindestens *zwei* Piloten vorhanden sind.

$I = \Big\{$ ______________________ $\Big\}$

⇑⇑⇑ ______ / 3 Punkt(e) ⇑⇑⇑

b) Zeigen Sie, dass `Flugzeugtest` $\in \mathcal{NP}$.

⇑⇑⇑ ______ / 2 Punkt(e) ⇑⇑⇑



1 0 1 5 2 2 2 1 0 0 0 0 0

c) Zeigen Sie, dass `Flugzeugtest` $\mathcal{NP}$-hart ist.

⇑⇑⇑ ______ / 6 Punkt(e) ⇑⇑⇑

a) Die Sprache $L_1$ sei wie folgt definiert:

$$L_1 := \{\langle M \rangle \mid M \text{ ist eine Turingmaschine und } M \text{ hält } \textit{nur} \text{ auf dem leeren Wort}\}$$

Zeigen Sie, dass die Sprache $L_1$ unentscheidbar ist.

⇑⇑⇑ ______ / 4 Punkt(e) ⇑⇑⇑

b) Betrachten Sie die folgende Sprache:

$$L_2 := \big\{\langle M\rangle \mid M \text{ ist eine Turingmaschine und der Lesekopf von } M \\ \text{liest auf Eingabe } \epsilon \text{ keine Zelle auf dem Band mehrfach}\big\}$$

Ist die Sprache $L_2$ entscheidbar? Beweisen Sie Ihre Antwort.

⇑⇑⇑ ______ / 3 Punkt(e) ⇑⇑⇑

c) Ist die Sprache $\overline{L_2}$ rekursiv aufzählbar? Beweisen Sie Ihre Antwort.

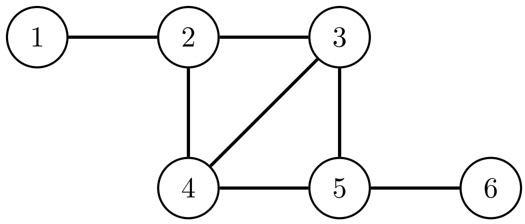
⇑⇑⇑ ______ / 2 Punkt(e) ⇑⇑⇑

Gegeben sei ein ungerichteter Graph $G = (V, E)$ mit $V = \{1, \dots, n\}$ und eine natürliche Zahl $k < n$. Eine Kante $e \in E$ wird durch eine Menge $S \subseteq V$ von Knoten überdeckt, wenn $S \cap e \neq \emptyset$. In $G$ soll nun eine Menge $S \subseteq V$ von $|S| = k$ Knoten gefunden werden, so dass möglichst viele Kanten durch $S$ überdeckt werden.

Betrachten Sie den folgenden Approximationsalgorithmus für das Problem:

$S := \emptyset$;
**for** $i = 1 \dots k$ **do**
  Wähle einen Knoten $v \in V \setminus S$ mit maximalem Grad. Wenn es mehrere solche gibt, wähle den mit kleinstem Index; // grad$(v) = |\{e \in E : v \in e\}|$
  $S := S \cup \{v\}$; // füge Knoten $v$ zu $S$ hinzu
Gebe $S$ aus;

a) Gegeben sei folgende Instanz $I$ des Problems mit $k = 2$:




i) **(1 Punkt)** Geben Sie die vom Algorithmus berechnete Knotenmenge $S_{\text{alg}}$ an.

$S_{\text{alg}} = \{$ ____________________ $\}$

ii) **(1 Punkt)** Geben Sie eine optimale Knotenmenge $S_{\text{opt}}$ an.

$S_{\text{opt}} = \{$ ____________________ $\}$

iii) **(1 Punkt)** Geben Sie den Approximationsfaktor des Algorithmus auf $I$ an.

Approximationsfaktor: ____________

⇑⇑⇑ ______ / 3 Punkt(e) ⇑⇑⇑

b) Zeigen Sie, dass der Algorithmus *nicht* $\delta$-approximativ ist für $\delta < 3/2$.

⇑⇑⇑ ______ / 3 Punkt(e) ⇑⇑⇑

c) Zeigen Sie, dass der Algorithmus stets eine 2-approximative Lösung ausgibt.

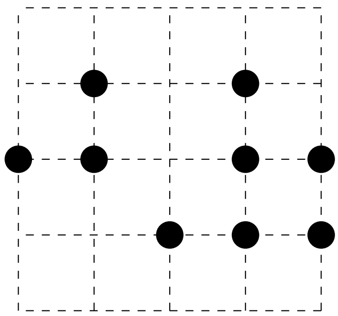
⇑⇑⇑ ______ / 3 Punkt(e) ⇑⇑⇑

Das Gelände der Ferienanlage Schöneruh kann durch einen quadratischen Gittergraphen $G_d$ mit $d$ Knoten entlang jeder Seite modelliert werden. Dabei entsprechen alle Kanten Wegen und alle Knoten Kreuzungen von Wegen. An bestimmten Kreuzungen (nicht unbedingt an allen) stehen insgesamt $k > 0$ Hütten, in denen Gäste untergebracht sind.

Da die Gäste richtig entspannen wollen, möchten sie einander auf keinen Fall begegnen. Im Problem `FreiePfade` sollen daher Pfade von jeder Hütte zum Rand der Anlage bestimmt werden, so dass keine zwei Pfade eine gemeinsame Kreuzung nutzen. Die Pfade dürfen nur entlang der Kanten im Gitter verlaufen.

a) Betrachten Sie die beiden folgenden Eingaben für `FreiePfade` mit $d = 5$ und $k = 9$ Hütten. Die schwarzen Punkte markieren die Positionen $h_1, ..., h_9$ der Hütten. Geben Sie jeweils an, ob es eine Lösung für die Eingabe gibt oder nicht. Falls ja, zeichnen Sie eine entsprechende Lösung ein; falls nicht, begründen Sie dies kurz anhand der Abbildung.

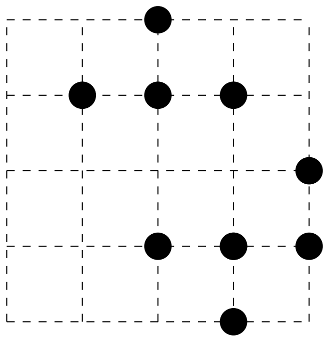
i) **(2 Punkte)**



☐ Ja, es gibt eine Lösung (eingezeichnet).

☐ Nein, es gibt keine Lösung. Begründung:

ii) **(2 Punkte)**



☐ Ja, es gibt eine Lösung (eingezeichnet).

☐ Nein, es gibt keine Lösung. Begründung:

⇑⇑⇑ ______ / 4 Punkt(e) ⇑⇑⇑

b) Gegeben sei ein Gittergraph $G_d$ zusammen mit den Positionen von $k$ Hütten $h_1, ..., h_k$.

Beschreiben Sie, wie mittels eines Flussnetzwerks in polynomieller Laufzeit bestimmt werden kann, ob eine Lösung für `FreiePfade` existiert oder nicht. Beschreiben Sie dabei das Flussnetzwerk explizit. Begründen Sie auch die Korrektheit Ihrer Idee.

⇑⇑⇑ ______ / 6 Punkt(e) ⇑⇑⇑