Name: ______________________________ Matrikel-Nr.: ____________________

Prof. Dr. Andreas Podelski
Dr. Matthias Heizmann
Alexander Nutz

22.03.2019

# Klausur zur Vorlesung
# Theoretische Informatik
## WS 2018/2019

Die Klausur besteht aus diesem Deckblatt und zehn Aufgabenblättern. Falls Sie eine Aufgabe nicht auf dem entsprechenden Blatt bearbeiten, machen Sie das bitte deutlich kenntlich. Auf Anfrage erhalten Sie zusätzliches Papier. Tragen Sie auf jedem Blatt bitte Ihren Namen und Ihre Matrikelnummer ein.

Abgesehen von einem beschriebenen DIN A4-Blatt sind keine Hilfsmittel zugelassen. Zur Bearbeitung haben Sie 120 Minuten Zeit. Insgesamt können 120 Punkte erzielt werden.

Nicht lesbare Lösungen/Lösungsversuche werden nicht gewertet. Falls Sie eine Aufgabe mehrmals bearbeiten, machen Sie bitte kenntlich, welche Lösung bewertet werden soll.

| | |
|---|---|
| 1. Reguläre Sprachen | von 8 |
| 2. Nerode-Relation | von 8 |
| 3. Endliche Automaten | von 15 |
| 4. Pumping Lemma | von 12 |
| 5. Grammatiken | von 16 |
| 6. CYK-Algorithmus | von 10 |
| 7. Kellerautomaten | von 8 |
| 8. Turingmaschinen | von 12 |
| 9. Entscheidbarkeit | von 14 |
| 10. Komplexität | von 17 |
| Summe | von 120 |

Name: ________________________________ Matrikel-Nr.: ____________________

**1. Aufgabe** (Reguläre Sprachen) **8**

Betrachten Sie den folgenden regulären Ausdruck $\alpha$ über dem Alphabet $\Sigma = \{a, b\}$.

$$\alpha := (b + a^*ab)^*aa^*$$

(a) Geben Sie zwei verschiedene Wörter an, die in $L(\alpha)$ liegen. 2

(b) Geben Sie zwei verschiedene Wörter an, die nicht in $L(\alpha)$ liegen. 2

(c) Geben Sie einen regulären Ausdruck $\alpha'$ an, sodass die Gleichheit der Sprachen $L(\alpha') = L(\alpha)$ gilt und $\alpha'$ nur einen Stern enthält. 4

........................................ Lösung ........................................

(a) $a, aa, ba, aaa, baa, aba, bba$

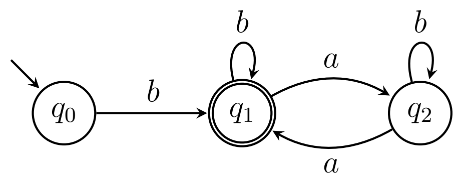
(b) $\varepsilon, b, bb, ab$

(c) $(a + b)^*a$

Name: ______________________________ Matrikel-Nr.: __________________

## 2. Aufgabe (Nerode-Relation) 8

Betrachten Sie den folgenden NEA $\mathcal{A}$ über dem Alphabet $\Sigma = \{a, b\}$.




(a) Wie viele Äquivalenzklassen hat die Nerode-Relation der Sprache $L(\mathcal{A})$?

(b) Geben Sie für jede Äquivalenzklasse genau einen Repräsentanten an.

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Lösung . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

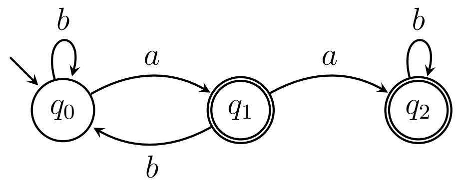
(a) 4

(b) $\varepsilon$, $b$, $ba$, $a$

Name: ______________________________ Matrikel-Nr.: ____________________

## 3. Aufgabe (Endliche Automaten) 15

Betrachten Sie den folgenden NEA $\mathcal{A}$ über dem Alphabet $\Sigma = \{a, b\}$.




(a) Konstruieren Sie einen NEA $\mathcal{A}'$ sodass $\mathcal{A}'$ nur einen akzeptierenden Zustand hat und die Gleichheit $L(\mathcal{A}') = L(\mathcal{A})$ gilt. Beschreiben Sie $\mathcal{A}'$ durch ein Zustandsdiagramm. 5

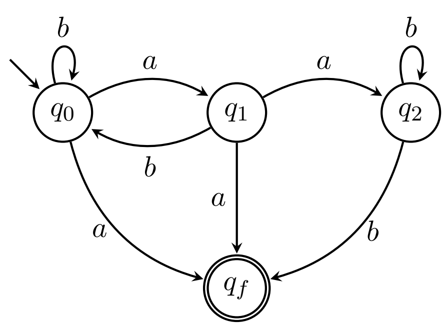
(b) Beweisen Sie die folgende Behauptung: Zu jedem NEA $\mathcal{N} = (\Sigma, Q, \delta, q^{\text{init}}, F)$ mit $\varepsilon \notin L(\mathcal{N})$ gibt es einen NEA $\mathcal{N}_1$ sodass $\mathcal{N}_1$ nur einen akzeptierenden Zustand hat und die Gleichheit $L(\mathcal{N}_1) = L(\mathcal{N})$ gilt. 10

*Zur Erinnerung:* Die akzeptierte Sprache eines NEA $\mathcal{N}$ wurde in der Vorlesung wie folgt definiert.

> Wir nennen eine Folge von Zuständen $q_0 q_1 \dots q_n$ einen *Lauf von* $\mathcal{N}$ *über* $w = a_1 \dots a_n$, falls $q_i \in \delta(q_{i-1}, a_i)$ für alle $i$ mit $1 \leq i \leq n$. Wir nennen einen Lauf *initial*, falls $q_0 = q^{\text{init}}$. Wir nennen einen Lauf *akzeptierend*, falls $q_n \in F$. Ein Wort $w \in \Sigma^*$ wird von $\mathcal{N}$ *akzeptiert*, falls $\mathcal{N}$ einen initialen und akzeptierenden Lauf über $w$ hat. Die von $\mathcal{N}$ akzeptierte Sprache ist die Menge der von $\mathcal{N}$ akzeptierten Wörter, d.h. $L(\mathcal{N}) = \{w \in \Sigma^* \mid \exists \text{ initialer, akzeptierender Lauf von } \mathcal{N} \text{ über } w\}$.

.................................... Lösung ....................................

(a) Idee: Füge zusätzlichen Zustand $q_f$ ein, nur dieser ist akzeptierend, jede Kante die früher in einen akzeptierenden Zustand ging geht nun auch in $q_f$.




(b) Wir Konstruieren zunächst $\mathcal{N}_1 = (\Sigma, Q_1, \delta_1, q_1^{\text{init}}, F_1)$ mit

- $Q_1 = Q \cup \{q_f\}$
- $\delta_1(q, a) = \begin{cases} \delta(q, a) & \text{falls } \delta(q, a) \cap F = \emptyset \\ \delta(q, a) \cup \{q_f\} & \text{sonst} \end{cases}$

Name: ______________________________ Matrikel-Nr.: ________________

- $q_1^{\mathsf{init}} = q^{\mathsf{init}}$
- $F_1 = \{q_f\}$

und zeigen dann die Gleichheit $L(\mathcal{N}_1) = L(\mathcal{N})$.

$$\begin{aligned}
& a_1 \dots a_n \in L(\mathcal{N}) \text{ sodass } n \geq 1 \\
\Leftrightarrow\ & \text{es gibt Zustandsfolge } q_0 q_1 \dots q_n \text{ sodass } n \geq 1, q_0 = q^{\mathsf{init}}, q_n \in F \\
& \qquad\qquad \text{und } q_i \in \delta(q_{i-1}, a_i) \text{ für alle } i \in \{1, \dots, n\} \\
\Leftrightarrow\ & \text{es gibt Zustandsfolge } q_0 q_1 \dots q_n \text{ sodass } n \geq 1, q_0 = q^{\mathsf{init}}, q_n = q_f \\
& \qquad\qquad \text{und } q_i \in \delta_1(q_{i-1}, a_i) \text{ für alle } i \in \{1, \dots, n\} \\
\Leftrightarrow\ & a_1 \dots a_n \in L(\mathcal{N}_1) \text{ sodass } n \geq 1
\end{aligned}$$

Name: ______________________________ Matrikel-Nr.: ____________________

**4. Aufgabe** (Pumping Lemma) $\boxed{\mathbf{12}}$

Zeigen Sie mithilfe des Pumping Lemmas, dass die folgende Sprache $L$ über dem Alphabet $\Sigma = \{a, b, c\}$ nicht regulär ist.

$$L = \{a^m b^k c^k \mid m \geq 0, k \geq 1\}$$

*Bemerkung*: Achten Sie darauf, dass die einzelnen Beweisschritte für den Leser nachvollziehbar sind. (Zum Beispiel sollte ersichtlich sein, ob ein Schritt eine Folgerung oder eine neue Annahme ist.)

.................................... Lösung ....................................

Dieser Beweis ist sehr kurz gehalten und würde gerade noch für die volle Punktzahl ausreichen. Sie finden einen ausführlichen Pumping Lemma Beweis im Skript und einen sehr ausführlichen Pumping Lemma Beweis in der ersten Zwischenklausur.

Sei $n \in \mathbb{N}$ mit $n \geq 1$. Wähle $z = b^n c^n \in L$. Offensichtlich gilt $|z| = 2n \geq n$.
Sei $z = uvw$ mit $u, v, w \in \Sigma^*$ eine beliebige Zerlegung für die $|uv| \leq n$ und $|v| > 0$ gilt.
Somit gilt $v \in L(b^+)$. Wähle $i = 0$ und betrachte $z' := uv^i w$. Es gilt $z' = b^{n-|v|} c^n$ und somit $z' \notin L$
Mit Hilfe des Pumping Lemma schließen wir, dass $L$ nicht regulär ist.

Name: ______________________________ Matrikel-Nr.: __________________

**5. Aufgabe** (Grammatiken) **16**

Betrachten Sie die Grammatik $\mathcal{G} = (\Sigma, N, P, S)$ mit dem Alphabet $\Sigma = \{a, b\}$, Nichtterminalsymbolen $N = \{S, A, B, C, D\}$ und den folgenden Regeln.

$$P = \{ \quad \begin{aligned} & S \to ASB, \\ & AS \to C \\ & C \to D \\ & DB \to \varepsilon \\ & A \to a \\ & B \to b\} \end{aligned}$$

(a) Geben Sie eine Ableitung für das Wort $ab$ an. 3

(b) Geben Sie ein von $ab$ verschiedenes Wort an, das in der Sprache $L(\mathcal{G})$ liegt. 2

(c) Geben Sie ein Wort an, das nicht in der Sprache $L(\mathcal{G})$ liegt. 2

(d) Beantworten Sie die folgenden vier Fragen jeweils mit ja/nein. [1] 2

| | | |
|---|---|---|
| Ist $\mathcal{G}$ eine Typ-0 Grammatik? | ☐ ja | ☐ nein |
| Ist $\mathcal{G}$ eine Typ-1 (kontextsensitive) Grammatik? | ☐ ja | ☐ nein |
| Ist $\mathcal{G}$ eine Typ-2 (kontextfreie) Grammatik? | ☐ ja | ☐ nein |
| Ist $\mathcal{G}$ eine Typ-3 (reguläre) Grammatik? | ☐ ja | ☐ nein |

(e) Beantworten Sie die folgenden vier Fragen jeweils mit ja/nein.[1] 2

| | | |
|---|---|---|
| Ist $L(\mathcal{G})$ eine Typ-0 Sprache? | ☐ ja | ☐ nein |
| Ist $L(\mathcal{G})$ eine Typ-1 (kontextsensitive) Sprache? | ☐ ja | ☐ nein |
| Ist $L(\mathcal{G})$ eine Typ-2 (kontextfreie) Sprache? | ☐ ja | ☐ nein |
| Ist $L(\mathcal{G})$ eine Typ-3 (reguläre) Sprache? | ☐ ja | ☐ nein |

(f) Konstruieren Sie eine kontextfreie Grammatik $\mathcal{G}'$ in Chomsky Normalform sodass $L(\mathcal{G}') = L(\mathcal{G})$ gilt. 5

........................................ Lösung ........................................

(a) $S \vdash ASB \vdash AASBB \vdash ACBB \vdash ADBB \vdash AB \vdash aB \vdash ab$

(b) $\varepsilon, aabb, aaabbb$

(c) $a, b, ba, aa, bb$

(d)

| | | |
|---|---|---|
| Ist $\mathcal{G}$ eine Typ-0 Grammatik? | ☒ ja | ☐ nein |
| Ist $\mathcal{G}$ eine Typ-1 (kontextsensitive) Grammatik? | ☐ ja | ☒ nein |
| Ist $\mathcal{G}$ eine Typ-2 (kontextfreie) Grammatik? | ☐ ja | ☒ nein |
| Ist $\mathcal{G}$ eine Typ-3 (reguläre) Grammatik? | ☐ ja | ☒ nein |

[1] In dieser Aufgabe werden keine Teilpunkte für falsche oder unvollständige Lösungen vergeben.

Name: ______________________________ Matrikel-Nr.: __________________

(e) Ist $L(\mathcal{G})$ eine Typ-0 Sprache? ☒ ja ☐ nein
Ist $L(\mathcal{G})$ eine Typ-1 (kontextsensitive) Sprache? ☒ ja ☐ nein
Ist $L(\mathcal{G})$ eine Typ-2 (kontextfreie) Sprache? ☒ ja ☐ nein
Ist $L(\mathcal{G})$ eine Typ-3 (reguläre) Sprache? ☐ ja ☒ nein

(f) $\mathcal{G}' = (\Sigma, N', P', S')$ mit Nichtterminalsymbolen $N = \{S', S, X_{AS}, A, B\}$ und den folgenden Regeln.

$$\begin{aligned} P = \{ \quad & S' \to S \mid \varepsilon, \\ & S \to X_{AS}B, \\ & X_{AS} \to AS \\ & S \to AB, \\ & A \to a \\ & B \to b\} \end{aligned}$$

Name: ______________________________ Matrikel-Nr.: ____________________

**6. Aufgabe** (CYK-Algorithmus) $\boxed{10}$

Betrachten Sie die kontextfreie Grammatik $\mathcal{G} = (\Sigma, N, P, S)$ mit dem Alphabet $\Sigma = \{a, b, c\}$, Nichtterminalsymbolen $N = \{S, A, B, C\}$ und den folgenden Regeln.

$$
\begin{aligned}
P = \{\ & S \to AB, \\
& A \to a \mid BC \mid CB, \\
& B \to b \mid c, \\
& C \to a \mid b \mid SS \mid BC\ \}
\end{aligned}
$$

Wenden Sie den CYK-Algorithmus an, um das Wortproblem für die Grammatik $\mathcal{G}$ und das Wort $w = bbbc$ zu entscheiden. Gehen Sie dazu wie folgt vor.

(a) Vervollständigen Sie die unten stehende Tabelle. $\boxed{6}$

(b) Liegt $w$ in der Sprache von $\mathcal{G}$? Begründen Sie Ihre Antwort mithilfe der Tabelle. $\boxed{1}$

(c) Geben Sie (ohne weitere Begründung) die Menge aller echten Präfixe von $w$ an, die in $L$ enthalten sind. $\boxed{3}$

(Zur Erinnerung: Für gegebenes Wort $w \in \Sigma^*$ ist $x \in \Sigma^*$ ein *echtes* Präfix, wenn gilt $w = xy$ für ein $y \in \Sigma^*$ das nicht das leere Wort ist.)

| | | | |
|---|---|---|---|
| | | | |
| b | | | |
| | b | | |
| | | b | |
| | | | c |

.................................... Lösung ....................................

(a)

| | | | |
|---|---|---|---|
| $B, C$ | $A, C$ | $S, A, C$ | $S, A$ |
| b | $B, C$ | $A, C$ | $S, A$ |
| | b | $B, C$ | $A$ |
| | | b | $B$ |
| | | | c |

(b) Ja, denn das Startsymbol $S$ taucht in der oberen rechten Ecke auf.

(c) $\{bbb\}$

Begründung (nicht erforderlich): Wenn wir die letzte Spalte der Matrix weglassen erhalten wir die Matrix für das Wort $bbb$ und in dieser "Untermatrix" steht $S$ in der oberen rechten Ecke. Wenn wir weitere Spalten weglassen steht $S$ nicht mehr in der oberen rechten Ecke der "Untermatrix".

Name: ________________________________ Matrikel-Nr.: __________________

**7. Aufgabe** (Kellerautomaten) 8

Betrachten Sie die folgende Sprache $L$ über dem Alphabet $\Sigma = \{a, b\}$.

$$L = \{a^n b^n \mid n \in \mathbb{N} \text{ und } n \geq 1\}$$

Konstruieren Sie einen Kellerautomaten der $L$ akzeptiert und dessen Zustandsmenge nur ein Element enthält.

Vervollständigen Sie dafür den Kellerautomaten $\mathcal{K} = (\Sigma, Q, \Gamma, q^{\text{init}}, Z^{\text{init}}, \delta)$ mit $Q = \{q\}, q^{\text{init}} = q$ und $Z^{\text{init}} = \#$ durch ein geeignetes Kelleralphabet $\Gamma$ und eine geeignete Transitionsfunktion $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \to \mathcal{P}(Q \times \Gamma^*)$. Beschreiben Sie die Transitionsfunktion mithilfe der unten abgebildeten Tabelle.

*Zur Erinnerung:* Wenn mehrere Symbole auf den Keller gelegt werden steht Symbol welches ganz oben auf den Kellerspeicher kommt in letzten Eintrag der Tabelle ganz links.

$$\Gamma = \{\#,$$

| $Q$ | $\Sigma \cup \{\varepsilon\}$ | $\Gamma$ | $Q$ | $\Gamma^*$ |
|---|---|---|---|---|
| | | | | |

Lösung

$$\Gamma = \{\#, S, A, B\}$$

| $Q$ | $\Sigma \cup \{\varepsilon\}$ | $\Gamma$ | $Q$ | $\Gamma^*$ |
|---|---|---|---|---|
| $q$ | $\varepsilon$ | $\#$ | $q$ | $ASB$ |
| $q$ | $a$ | $A$ | $q$ | $\varepsilon$ |
| $q$ | $\varepsilon$ | $S$ | $q$ | $ASB$ |
| $q$ | $\varepsilon$ | $S$ | $q$ | $\varepsilon$ |
| $q$ | $b$ | $B$ | $q$ | $\varepsilon$ |

Name: ______________________________ Matrikel-Nr.: ____________________

## 8. Aufgabe (Turingmaschinen) 12

Betrachten Sie die deterministische Turingmaschine $\mathcal{M} = (\Sigma, Q, \Gamma, \delta, q^{\text{init}}, \sqcup, F)$ mit $\Sigma = \{0, 1\}$, $Q = \{q_s, q_r, p_0, p_1, q_e\}$, $\Gamma = \Sigma \cup \{\sqcup\}$, $q^{\text{init}} = q_s$, $F = \{q_e\}$ und $\delta$ gegeben durch folgende Turingtafel.

| | | | | |
|---|---|---|---|---|
| $q_s$ | 0 | $q_r$ | 0 | $R$ |
| $q_s$ | 1 | $p_1$ | 1 | $R$ |
| $q_r$ | 0 | $q_r$ | 0 | $R$ |
| $q_r$ | 1 | $q_r$ | 1 | $R$ |
| $q_r$ | $\sqcup$ | $q_r$ | $\sqcup$ | $R$ |
| $p_1$ | 0 | $p_0$ | 1 | $R$ |
| $p_1$ | 1 | $p_1$ | 0 | $R$ |
| $p_1$ | $\sqcup$ | $q_e$ | $\sqcup$ | $R$ |
| $p_0$ | 0 | $p_0$ | 1 | $R$ |
| $p_0$ | 1 | $p_1$ | 0 | $R$ |

Sei $w = 110$.

(a) Geben Sie die Haltekonfiguration von $\mathcal{M}$ angesetzt auf das Wort $w$ an. 2

(b) Welches Resultat liefert die von $\mathcal{M}$ berechnete partielle Funktion $f_{\mathcal{M}}$ für das Wort $w$? 2

(c) Geben Sie ein Wort an, das von $\mathcal{M}$ akzeptiert wird. 2

(d) Beschreiben Sie die von $\mathcal{M}$ akzeptierte Sprache mithilfe eines regulären Ausdrucks. 3

(e) Welche partielle Funktion $f$ berechnet $\mathcal{M}$? Geben Sie eine präzise Beschreibung von $f$. Sie dürfen dabei die folgende Funktion benutzen. 3

$$\mathsf{inv} : \Sigma \to \Sigma \qquad \mathsf{inv}(x) = \begin{cases} 0 & \text{falls } x = 1 \\ 1 & \text{falls } x = 0 \end{cases}$$

........................................ Lösung ........................................

(a) $101p_0\sqcup$

(b) 101

(c) $1, 101, 111, 1001, 1011, 1101, 111$

(d) $1\ +\ 1(0+1)^*1$

(e) $f(w) = \begin{cases} a_1\ \mathsf{inv}(a_2)\ldots\mathsf{inv}(a_2) & \text{falls } w = a_1\ldots a_n \text{ mit } a_1 = 1 \\ \varepsilon & \text{falls } w = \varepsilon \\ undef. & \text{sonst} \end{cases}$

Name: ______________________________ Matrikel-Nr.: ________________

**9. Aufgabe** (Entscheidbarkeit) $\boxed{14}$

Betrachten Sie das Alphabet $\Sigma = \{0, 1, \#\}$ und die folgende Sprache.

$$H_2 = \left\{ w\#v \;\middle|\; \begin{array}{l} v \in \{0,1\}^* \text{ enthält mindestens zweimal das Zeichen 1,} \\ w \in \{0,1\}^* \text{ und } \mathcal{M}_w \text{ angesetzt auf } v \text{ hält} \end{array} \right\}$$

Zeigen Sie mithilfe von Reduktion, dass $H_2$ unentscheidbar ist. Verwenden Sie dabei, dass das Halteproblem auf dem leeren Band $H_\varepsilon$ unentscheidbar ist.

.................................... Lösung ....................................

Wir zeigen $H_\varepsilon \leq H_2$ und konstruierne uns dafür eine Reduktionsfunktion $f : \Sigma^* \to \Sigma^*$.
Zunächst definieren wir uns $\mathcal{M}_{11}$ als die TM, die terminiert falls die Eingabe nicht $11$ ist und ansonsten die Eingabe $11$ löscht und dann anhält (die also für Eingabe $11$ ein leeres Eingabeband hinterlässt).
Weiter definieren wir für $w \in \Sigma^*$ die TM $\mathcal{M}'_w$ als die TM die zuerst $\mathcal{M}_{11}$ und anschließend $\mathcal{M}_w$ ausführt. Unsere Reduktionsfunktion $f$ sei nun wie folgt definiert.

$$f(w) = \ulcorner M'_w \urcorner \#11$$

Die Funktion $f$ is offensichtlich total und berechenbar, außerdem gelten die folgenden Äquivalenzen.

$$\begin{aligned} w \in H_\varepsilon \quad &\Leftrightarrow \quad \mathcal{M}_w \text{ angesetzt auf } \varepsilon \text{ hält} \\ &\Leftrightarrow \quad \mathcal{M}'_w \text{ angesetzt auf } 11 \text{ hält} \\ &\Leftrightarrow \quad f(w) \in H_2 \end{aligned}$$

Da $H_\varepsilon$ unentscheidbar ist, ist somit auch $H_2$ unentscheidbar.

Name: ______________________________ Matrikel-Nr.: ____________________

**10. Aufgabe** (Komplexität) **17**

Das Problem ISET (Independent Set) ist wie folgt definiert:

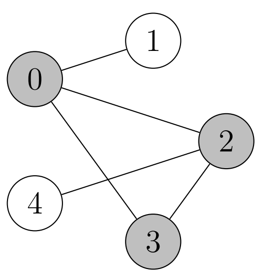
> *Gegeben*: Ein ungerichteter Graph $\mathcal{G} = (V, E)$ und eine natürliche Zahl $k \leq |V|$.
>
> *Frage*: Besitzt $\mathcal{G}$ eine „unabhängige Knotenmenge“ (independent set) der Größe mindestens $k$? Eine unabhängige Knotenmenge ist eine Teilmenge $V' \subseteq V$, sodass für alle Knoten $u, v \in V'$ gilt $(u, v) \notin E$.

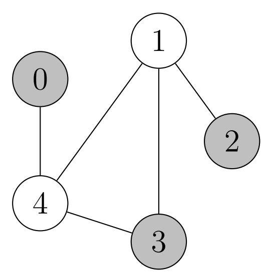
Zeigen Sie, dass ISET $NP$-vollständig ist. Verwenden Sie dazu das $NP$-vollständige Problem CLIQUE, das wir in der Vorlesung wie folgt definiert hatten.

> *Gegeben:* Ein ungerichteter Graph $\mathcal{G} = (V, E)$ und eine Zahl $k \in \mathbb{N}$.
>
> *Frage:* Hat $\mathcal{G}$ eine $k$-Clique? Eine $k$-Clique ist eine $k$-elementige Menge von Knoten, die paarweise durch Kanten verbunden sind, d.h., eine Menge $C \subseteq V$, sodass $|C| = k$ und $\forall u, v \in C : u \neq v \rightarrow (u, v) \in E$.

Es genügt, wenn Sie Ihre Laufzeitabschätzungen grob begründen. Sie müssen weder Pseudocode noch Turingmaschinen explizit angeben.

*Beispiele:* Im linken Graph ist die Knotenmenge $\{0, 2, 3\}$ eine 3-Clique. Der rechte Graph hat mit $\{0, 2, 3\}$ eine unabhängige Knotenmene der Größe 3.




........................................ Lösung ........................................

Für NP-Vollständigkeit von ISET zeigen wir dass (a) ISET $\in NP$ gilt und dass (b) ISET NP-schwer ist.

(a) Man kann eine Teilmenge $V'$ raten und in polynomieller Zeit überprüfen, ob sie eine Lösung für ISET ist. Vergleiche dazu die Größe von $V'$ mit $k$ (in $O(|V'| + k) \subseteq O(|V| + k)$) und prüfe, ob keine Kante zwischen zwei Knoten in $V'$ verläuft (betrachte alle Paare von Knoten in $V'$ (in $O(|V'|^2 \cdot |E|) \subseteq O(|V|^2 \cdot |E|)$).

(b) Wir zeigen dass ISET NP-schwer ist in dem wir das NP-schwere Problem CLIQUE darauf reduzieren (wir zeigen also CLIQUE $\preceq_p$ ISET). Die Reduktionsfunktion $f$ definieren wir wie folgt.

$$f\big(\,(V, E), k\,\big) = \big(\,(V, \{(v, v') \in V \times V \mid (v, v') \notin E\}), k\,\big)$$

In Worten: Wir lassen die Knotenmenge und die Zahl k unverändert und “invertieren” die Kantenmenge.

- Die Reduktionsfunktion ist offensichtlich total.

Name: ______________________________ Matrikel-Nr.: __________________

- Die Reduktionsfunktion ist in polynomieller Zeit berechenbar:

  Wir müssen ein mal über alle Knotenpaare iterieren und prüfen ob die entsprechende Kante in $E$ enthalten ist. Sowohl Knotenmenge als auch $E$ sind durch Länge der Eingabe beschränkt. Gesamte Laufzeit ist dann auf Dreibandmaschine (Ein Band zum Ausgabe schreiben, ein Band um $V$ zwischenzuspeichern, ein Band um $E$ zwischenzuspeichern) durch Polynom vom Grad drei beschränkt.

- Sei $(V, E', k) = f((V, E, k))$

$$\begin{aligned} & (V, E, k) \in \textsc{Clique} \\ \iff & \text{es gibt eine k-elementige Menge } X \subseteq V \text{ sodass } \forall u, v \in X : (u, v) \in E \\ \iff & \text{es gibt eine k-elementige Menge } X \subseteq V \text{ sodass } \forall u, v \in X : (u, v) \notin E' \\ \iff & (V, E', k) \in \textsc{Iset} \end{aligned}$$