

Name: \_\_\_\_\_ Matrikel-Nr.: \_\_\_\_\_

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

05.03.2019

Zweite Zwischenklausur zur Vorlesung  
Theoretische Informatik  
WS 2018/2019

Diese Zwischenklausur wird weder korrigiert noch mit Übungspunkten bewertet. Dieses Dokument orientiert sich in Schwierigkeit der Aufgaben und Layout an Haupt- und Nachklausur. Haupt- und Nachklausur werden allerdings jeweils 120 Minuten dauern. Einziges zugelassenes Hilfsmittel in Haupt- und Nachklausur ist ein DIN A4-Zettel (beidseitig) mit beliebigem handschriftlichen Inhalt zugelassen.

Auch wenn es keine offizielle Korrektur dieser Zwischenklausur gibt können Sie mir bis zum Mittag des 13.03.2019 Ihre Lösungen via E-Mail an [heizmann@informatik.uni-freiburg.de](mailto:heizmann@informatik.uni-freiburg.de) oder durch Abgabe in meinem Büro (Gebäude 052, Raum 00-019) zukommen lassen.

Ich werde mindestens die ersten fünf Lösungen die ich erhalte korrigieren (möglicherweise nicht mehr!) und die korrigierten Lösungen in der Ilias Lernplattform für alle Vorlesungsteilnehmer zugänglich machen. Geben Sie also nur dann eine Lösung ab wenn Sie mit dieser Veröffentlichung einverstanden sind. Sie dürfen selbstverständlich Ihre Lösung in anonymisierter Form abgeben oder Ihren Namen nur auf das Deckblatt schreiben.

Haben Sie keine Hemmungen eine schlechte Lösung abzugeben! Sie erhalten ein realistisches Resultat nur wenn Sie diese Zwischenklausur unter Klausurbedingungen in der angegebenen Zeit bearbeiten und die so erzielte Lösung abgeben.

Name: \_\_\_\_\_ Matrikel-Nr.: \_\_\_\_\_

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

23.12.2018

Zweite Zwischenklausur zur Vorlesung  
Theoretische Informatik  
WS 2018/2019

Die Klausur besteht aus diesem Deckblatt und vier 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 42 Minuten Zeit. Insgesamt können 42 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. Turingmaschinen	von 12
2. Semi-Entscheidbarkeit	von 13
3. Komplexität	von 17
Summe	von 42

**1. Aufgabe**

(Turingmaschinen)

**12**

Betrachten Sie die deterministische Turingmaschine  $\mathcal{M} = (\Sigma, Q, \Gamma, \delta, q^{\text{init}}, \sqcup, F)$  mit  $\Sigma = \{0, 1\}$ ,  $Q = \{q_0, q_1, q_2, q_\ell, q_e\}$ ,  $\Gamma = \Sigma \cup \{*, \sqcup\}$ ,  $q^{\text{init}} = q_0$ ,  $F = \{q_e\}$  und  $\delta$  gegeben durch folgende Turingtafel.

$q_0$	0	$q_1$	0	$R$
$q_1$	0	$q_2$	*	$R$
$q_2$	0	$q_2$	*	$R$
$q_2$	1	$q_2$	1	$R$
$q_2$	$\sqcup$	$q_\ell$	$\sqcup$	$L$
$q_\ell$	0	$q_\ell$	*	$L$
$q_\ell$	1	$q_\ell$	1	$L$
$q_\ell$	*	$q_\ell$	*	$L$
$q_\ell$	$\sqcup$	$q_e$	$\sqcup$	$N$

Sei  $w = 0010$ .

- (a) Geben Sie die Haltekonfiguration von  $\mathcal{M}$  angesetzt auf das Wort  $w$  an. 3
- (b) Wird das Wort  $w$  von  $\mathcal{M}$  akzeptiert? Begründen Sie Ihre Antwort kurz. 1
- (c) Welche Ausgabe liefert  $\mathcal{M}$  angesetzt auf das Wort  $w$ ? 1
- (d) Welche Sprache  $L$  akzeptiert  $\mathcal{M}$ ?  
Geben Sie eine präzise Beschreibung von  $L$  an (z.B. in Mengenschreibweise). 3
- (e) Welche partielle Funktion  $f$  berechnet  $\mathcal{M}$ ?  
Geben Sie eine präzise Beschreibung von  $f$  analog zu folgender Beschreibung einer Funktion  $g$  an. Sie dürfen die Funktion  $g$  in Ihrer Beschreibung verwenden. 4

$$g(w) = \begin{cases} \varepsilon & w = \varepsilon \\ *g(v) & \exists v : w = 0v \\ 1g(v) & \exists v : w = 1v \end{cases}$$

..... Lösung .....

- (a)  $\varepsilon q_e \sqcup * * 1 *$  oder  $(\varepsilon, q_e, \sqcup * * 1 *)$
- (b) Ja, da  $q_e \in F$ .
- (c)  $f(w) = 1$  (nach Projektion auf  $\Sigma$ )
- (d)  $L = \{w \mid \exists v : w = 00v\}$
- (e)  $f(w) = \begin{cases} 1^{\#_1(w)} & \exists v : w = 00v \\ w & \text{sonst} \end{cases}$

**2. Aufgabe**

(Semi-Entscheidbarkeit)

13

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

$$H_{110} = \{w \in \Sigma^* \mid \mathcal{M}_w \text{ angesetzt auf die Eingabe } 110 \text{ hält nicht.}\}$$

Zeigen Sie mithilfe von Reduktion, dass  $H_{110}$  nicht semi-entscheidbar ist.

*Hinweis:*<sup>1</sup> Aus der Vorlesung bekannte Beispiele für nicht semi-entscheidbare Sprachen sind: Das Komplement des speziellen Halteproblems  $\overline{K}$ , das Komplement des allgemeinen Halteproblems  $\overline{H}$ , das Komplement des Halteproblems auf dem leeren Band  $\overline{H}_\varepsilon$ , das universelle Halteproblem  $H_{\forall w}$  und das Komplement des universellen Halteproblems  $\overline{H}_{\forall w}$ .

..... Lösung .....

Wir zeigen  $\overline{H}_\varepsilon \leq H_{110}$  und konstruieren uns dafür eine Reduktionsfunktion  $f : \Sigma^* \rightarrow \Sigma^*$ .

Zunächst definieren wir uns  $\mathcal{M}_{110}$  als die TM, die terminiert falls die Eingabe nicht 110 ist und ansonsten die Eingabe 110 löscht und dann anhält (die also für Eingabe 110 ein leeres Eingabeband hinterlässt).

Weiter definieren wir für  $w \in \Sigma^*$  die TM  $\mathcal{M}'_w$  als die TM die zuerst  $\mathcal{M}_{110}$  und anschließend  $\mathcal{M}_w$  ausführt.

Unsere Reduktionsfunktion  $f$  sei nun die Funktion, die  $w \in \Sigma^*$  auf den Code von  $\mathcal{M}'_w$  abbildet. Die Funktion  $f$  ist offensichtlich total und berechenbar, außerdem gelten die folgenden Äquivalenzen.

$$\begin{aligned} w \in \overline{H}_\varepsilon &\Leftrightarrow \mathcal{M}_w \text{ angesetzt auf } \varepsilon \text{ hält nicht} \\ &\Leftrightarrow \mathcal{M}'_w \text{ angesetzt auf } 110 \text{ hält nicht} \\ &\Leftrightarrow f(w) \in H_{110} \end{aligned}$$

Da  $\overline{H}_\varepsilon$  nicht semi-entscheidbar ist, ist somit auch  $H_{110}$  nicht semi-entscheidbar.

---



---

<sup>1</sup>In der Klausur würde es bei einer vergleichbaren Aufgabe solch einen Hinweis möglicherweise nicht geben.

**3. Aufgabe**

(Komplexität)

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

*Gegeben:* Ein ungerichteter Graph  $G = (V, E)$  und eine natürliche Zahl  $k \leq |V|$ .  
*Frage:* Besitzt  $G$  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.

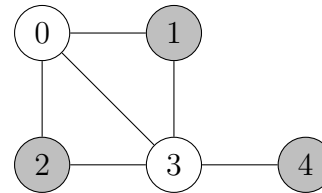
Verwenden Sie dazu das NP-vollständige Problem KNÜB (Knotenüberdeckung bzw. Vertex Cover), das wir auf Übungsblatt 14 wie folgt definiert hatten.

*Gegeben:* Ein ungerichteter Graph  $G = (V, E)$  und eine natürliche Zahl  $k \leq |V|$ .  
*Frage:* Besitzt  $G$  eine „überdeckende Knotenmenge“ (vertex cover) der Größe höchstens  $k$ ? Eine überdeckende Knotenmenge ist eine Teilmenge  $V' \subseteq V$ , sodass für alle Kanten  $(u, v) \in E$  gilt:  $u \in V'$  oder  $v \in V'$ .

Es genügt, wenn Sie Ihre Laufzeitabschätzungen grob begründen. Sie müssen weder Pseudocode noch Turingmaschinen explizit angeben. Sie dürfen das folgende Lemma verwenden:

**Lemma 1:** Eine Teilmenge  $V' \subseteq V$  ist genau dann eine unabhängige Knotenmenge, wenn das Komplement  $\bar{V}' := V \setminus V'$  eine überdeckende Knotenmenge ist.

*Beispiel:* Im rechts dargestellten Graphen ist  $\{0, 3\}$  eine überdeckende Knotenmenge der Größe 2 und  $\{1, 2, 4\}$  eine unabhängige Knotenmenge 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  $O(|V'|^2 + |E|) \subseteq O(|V|^2 + |E|)$ )).

- (b) Wir zeigen dass ISET NP-schwer ist in dem wir das NP-schwere Problem KNÜB darauf reduzieren (wir zeigen also  $KNÜB \preceq_p ISET$ ).

Die Reduktionsfunktion  $f$  definieren wir wie folgt. Zu gegebenem Graphen  $(V, E)$  und Konstante  $k$  konstruieren wir einen Graphen  $(V_2, E_2)$  und eine Konstante  $k_2$ . Wähle dabei  $V_2 = V$ ,  $E_2 = E$  und  $k_2 = |V| - k$ . Falls die Eingabe kein Tripel  $(V, E, k)$  beschreibt, so

Name: \_\_\_\_\_ Matrikel-Nr.: \_\_\_\_\_

kann man diese auch als Eingabe für KNÜB verwenden. Diese Konstruktion ist offensichtlich total und polynomiell, da man nur die Eingabe überprüfen muss (in polynomieller Zeit machbar) und ggf.  $k_2$  berechnen muss (in  $O(|V| + k)$ ).

Es bleibt zu zeigen:  $w \in \text{KNÜB} \iff f(w) \in \text{ISET}$ .

Falls  $w$  kein Tripel  $(V, E, k)$  beschreibt, gilt die Äquivalenz offensichtlich. Sei also  $w$  ein Tripel  $(V, E, k)$ . Dann gilt:

$$\begin{aligned} & (V, E, k) \in \text{KNÜB} \\ \iff & \text{es gibt eine überdeckende Knotenmenge } V' \subseteq V \text{ mit } |V'| = k \\ \stackrel{\text{Lemma}}{\iff} & \text{es gibt eine unabhängige Knotenmenge } V' \subseteq V \text{ mit } |V'| = |V| - k \\ \iff & \text{es gibt eine unabhängige Knotenmenge } V' \subseteq V_2 \text{ mit } |V'| = k_2 \\ \iff & (V_2, E_2, k_2) \in \text{ISET} \\ \iff & f(w) \in \text{ISET} \end{aligned}$$

---