Abschlussklausur
Theoretische Informatik 2
28. Juli 2020

Prof. Dr. Roland Meyer
Thomas Haas
TU Braunschweig
Sommersemester 2020

1. **Bitte am Anfang ausfüllen:**

   Vorname:

   Nachname:

   Matrikelnummer:

   Nummer des Sitzplatzes:

   Unterschrift:

2. Die Nummer Ihrer Klausur ist `altePO # 1`. Bitte merken Sie sich die Nummer. Wir werden Ihr Klausurergebnis anonymisiert unter Verwendung dieser Nummer bekanntgeben.
3. Achten Sie darauf, dass Ihre Klausur **vollständig ist** und **getackert bleibt** (20 Blätter).
4. Benutzen Sie **nur das an dieses Blatt angeheftete Papier**. Bei Bedarf können wir weitere Leerblätter austeilen. Wenn der Platz auf der Vorderseite des jeweiligen Aufgabenblatts nicht ausreicht, **machen Sie kenntlich**, wo Sie die Bearbeitung der Aufgabe fortsetzen.
5. Als Hilfsmittel sind **ausschließlich** Sprachwörterbücher sowie ein beidseitig **handschriftlich beschriebenes DIN A4-Blatt** erlaubt. Elektronische Geräte müssen während der Klausur ausgeschaltet bleiben. Täuschungsversuche werden als nicht bestanden gewertet und dem Prüfungsamt gemeldet.
6. Schreiben Sie leserlich und bearbeiten Sie Ihre Klausur mit einem **dokumentenechten Stift** (nicht mit Bleistift, kein Tipp-Ex, kein Tintenkiller) und **nicht in roter oder grüner Farbe**.
7. Wir werden die **Klausurergebnisse** auf unserer Website bekanntgeben:
   `tcs.cs.tu-bs.de/teaching/TheoInf2_SS_2020.html`.
8. Die **Bearbeitungszeit** beträgt **180 Minuten** (+ ggf. Zeit zum Lüften).
9. Mit **40 Punkten** ist die Klausur **sicher bestanden.**

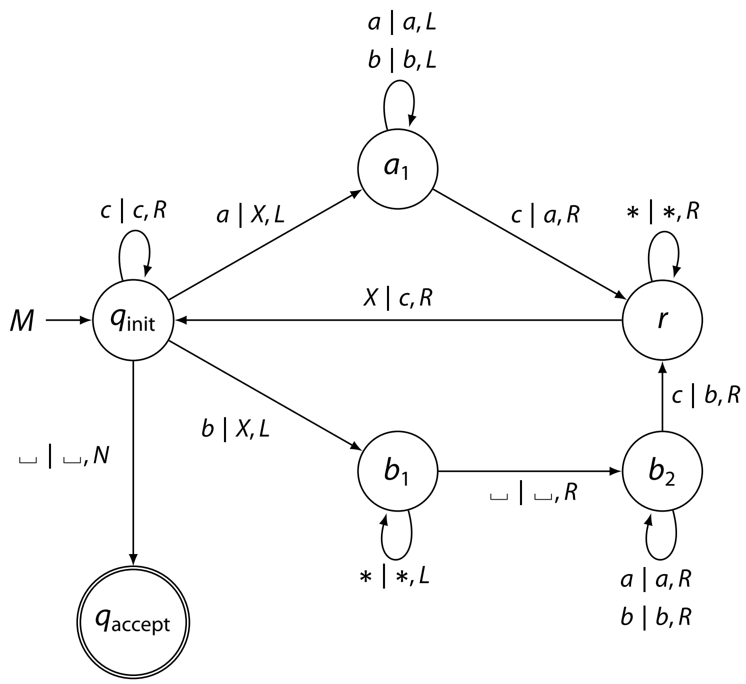
Bepunktung: (wird von den Korrektoren ausgefüllt)

Aufgabe	1	2	3	4	5	6	7	8	9	10	Σ
Max.	10	10	10	10	10	10	10	10	10	10	**100**
Punkte											

1. TM-Analyse

$\boxed{3 + 5 + 2 = 10 \text{ Punkte}}$

Betrachten Sie den 1-Band-Berechner M mit Eingabealphabet $\{a, b, c\}$, Bandalphabet $\{a, b, c, X, \sqcup\}$ und Zustandsmenge $\{q_{\text{init}}, q_{\text{accept}}, a_1, b_1, b_2, r\}$. Hierbei ist q_{init} der Startzustand und q_{accept} der akzeptierende Zustand. M berechnet eine partielle Funktion $f : \{a, b, c\}^* \to_p \{a, b, c\}^*$.



Eine mit $a \mid b, d$ beschriftete Kante von q nach q' steht für die Transition $\delta(q, a) = (q', b, d)$. Kanten mit mehreren Beschriftungen stehen für mehrere Transitionen. Die Notation $* \mid *, d$ ist eine Abkürzung für die Transitionen $s \mid s, d$ mit $s \in \{a, b, c\}$.

a) Bestimmen Sie die Funktionswerte $f(ccba)$, $f(ccab)$ und $f(abc)$.

b) Beschreiben Sie die von M berechnete Funktion f in Worten und geben Sie diese als Pseudo-Code an.

c) Bestimmen Sie, für welche Eingaben die Funktion f definiert ist.

Hinweise: Bei Eingabe w startet die Maschine in der Konfiguration $\ldots\sqcup q_{\text{init}}\ w \sqcup \ldots$, d.h. links und rechts von der Eingabe befinden sich $\sqcup$-Symbole und der Lese-/Schreibkopf zeigt auf den ersten Buchstaben der Eingabe. Der Funktionswert $f(w)$ ist definiert als w', wenn M bei Eingabe w mit Bandinhalt w' akzeptiert. Alle fehlenden Transitionen führen dazu, dass die Maschine die Berechnung beendet und keine Ausgabe erzeugt.

Zu Aufgabe 1:

2. NP-Vollständigkeit

$\boxed{4 + 6 = 10 \text{ Punkte}}$

Betrachten Sie das folgende Problem.

DodgyCycle (DCYCLE)	
Gegeben:	Ein gerichteter Graph $G = (V, E)$ und ein Knoten $v \in V$.
Entscheide:	Gibt es einen Kreis in G, der v nicht enthält und alle anderen Knoten genau einmal enthält?

Zeigen Sie, dass DCYCLE NP-vollständig (bzgl. Polynomialzeit-Reduktionen) ist:

a) „Membership“: DCYCLE $\in$ NP.

b) „Hardness“: DCYCLE ist NP-schwer (bzgl. Polynomialzeit-Reduktionen).

3. NL-Vollständigkeit

$\boxed{4 + 6 = 10 \text{ Punkte}}$

Betrachten Sie das folgende Problem.

Pfad gerader Länge (EVENPATH)	
Gegeben:	Ein gerichteter Graph $G = (V, E)$ und Knoten s, t.
Entscheide:	Gibt es in G einen Pfad von s nach t mit gerader Länge?

Hierbei entspricht die Länge eines Pfades genau der Anzahl an Kanten in dem Pfad.

Zeigen Sie, dass EVENPATH NL-vollständig (bzgl. logspace-many-one-Reduktionen) ist:

a) „Membership“: EVENPATH $\in$ NL.

b) „Hardness“: EVENPATH ist NL-schwer (bzgl. logspace-many-one-Reduktionen).

4. Entscheidbarkeit

$\boxed{4 + 6 = 10 \text{ Punkte}}$

Betrachten Sie das folgende Problem.

SQUARE-BOUNDED-ACCEPT	
Gegeben:	Kodierung w einer DTM mit Eingabealphabet $\{0, 1\}$.
Entscheide:	Gibt es ein Wort x, das von M_w in maximal $\|x\|^2$ Schritten akzeptiert wird?

a) Zeigen Sie, dass SQUARE-BOUNDED-ACCEPT semi-entscheidbar ist.

b) Zeigen Sie, dass SQUARE-BOUNDED-ACCEPT unentscheidbar ist.

5. Konstruktion einer DTM

10 Punkte

Es sei $\Sigma = \{a, b, c\}$ und $w \in \Sigma^*$ ein Wort. Wir bezeichnen mit $|w|_x$ die Anzahl an x in w (für $x \in \Sigma$).

Konstruieren Sie eine **deterministische** Turingmaschine M, welche die Sprache

$$\mathcal{L} = \left\{ w \in \Sigma^* \;\middle|\; |w|_a \cdot |w|_b = |w|_c \right\}$$

entscheidet. Beispielsweise soll M die Wörter ε, *acb* und *abccbacccca* akzeptieren.

- Erklären Sie die Arbeitsweise der Maschine ausführlich. Geben Sie insbesondere die Aufgabe jedes Kontrollzustands der Maschine an.
- Geben Sie die Transitionen der Maschine explizit an, z.B. in Form einer Tabelle oder als Zustandsgraph.
- Sie können wahlweise annehmen, dass das Band auf beiden Seiten der Eingabe mit ␣-Symbolen gefüllt ist, oder dass das Band auf der linken Seite durch ein \$-Symbol beschränkt ist. Geben Sie an, wofür Sie sich entschieden haben und geben Sie an, auf welches Symbol der Lese-/Schreibkopf initial zeigt.

Hinweis: Die Turingmaschine darf mehrere Bänder verwenden.

Zu Aufgabe 5:

6. Berechenbarkeit

4 + 6 Punkte

Es sei $\Sigma = \{0, 1\}$. Betrachten Sie die Funktion *maxSteps* : $\mathbb{N} \to \mathbb{N}$ definiert als

$$maxSteps(n) := \max_{\text{DTM } M}\{Time_M(\varepsilon) \mid M \text{ hat n Zustände, Bandalphabet } \Sigma \text{ und } \varepsilon \in \mathcal{L}(M) \subseteq \Sigma^*\}.$$

Zeigen Sie nun, dass

a) *maxSteps* wohldefiniert ist. Das heißt, dass $maxSteps(n) < \infty$ für alle Eingaben n gilt.

b) *maxSteps* nicht berechenbar ist.

Hinweis: Für Teilaufgabe b) betrachten Sie das Halteproblem auf Epsilon.

7. Quiz

$\boxed{2 + 2 + 3 + 3 = 10 \text{ Punkte}}$

Beantworten Sie die folgenden Fragen. Begründen Sie Ihre Antwort mit einem kurzen Beweis oder einem Gegenbeispiel.

a) Sei $\mathcal{L}_1 \cup \mathcal{L}_2$ entscheidbar. Sind dann $\mathcal{L}_1$ und $\mathcal{L}_2$ immer semi-entscheidbar?

b) Ist die Sprache $\mathcal{L} = \{0^n.1^n \mid n \in \mathbb{N}\}$ in DTIME($\mathcal{O}(1)$)?

c) Wenn die partielle Funktion $f_p : \Sigma^* \to_p \mathbb{N}$ berechenbar ist, dann ist auch die totale Erweiterung $f_{total} : \Sigma^* \to \mathbb{N}$ mit

$$f_{total}(x) := \begin{cases} f_p(x), & \textit{falls } f_p(x) \textit{ definiert ist.} \\ 0, & \textit{sonst} \end{cases}$$

immer berechenbar?

d) Ist die Funktion $f : \{0,1\}^* \to \{0,1\}$ mit $f(w) = 1$ genau dann, wenn M_w auf allen Eingaben x in maximal $|w|$ Schritten hält, berechenbar?

8. Starke NTMs

5 + 5 Punkte

Eine NTM M ist eine starke NTM (SNTM), wenn sie folgende Eigenschaften hat:

1. Sie hat 3 gesonderte Zustände q_{accept}, q_{reject} und q_{maybe}.
2. Jede Berechnung endet in einem der 3 gesonderten Zustände.
3. Zu jedem Wort gibt es eine Berechnung, die entweder in q_{accept} oder in q_{reject} landet.
4. Zu jedem Wort w gilt, dass entweder alle Berechnungen in q_{accept} oder q_{maybe} landen, oder alle Berechnungen in q_{reject} oder q_{maybe} landen. Im ersten Fall wird w akzeptiert, im zweiten abgelehnt.

Zeigen Sie, dass die Klasse SNP aller Probleme, die von einer starken NTM in Polynomialzeit entschieden werden können, genau der Klasse NP $\cap$ coNP entspricht. Gehen Sie dazu in zwei Schritten vor:

a) Zeigen Sie die Inklusion SNP $\subseteq$ NP $\cap$ coNP.

b) Zeigen Sie die Inklusion NP $\cap$ coNP $\subseteq$ SNP.

Hinweis: Beide Inklusionsrichtungen können durch Konstruktion geeigneter (S)NTMs gezeigt werden.

Zu Aufgabe 8:

9. Polylog

8 + 2 Punkte

Betrachten Sie die folgende Komplexitätsklasse

$$\text{PolyL} := \bigcup_{k=1}^{\infty} \text{DSPACE}(log(n)^k)$$

Es ist bekannt, dass die Inklusionen $\text{DSPACE}(log(n)^k) \subsetneq \text{DSPACE}(log(n)^{k+1})$ für alle $k \in \mathbb{N}$ echt sind.

a) Zeigen Sie, dass es keine PolyL-vollständigen Probleme bzgl. logspace-Reduktionen gibt.

b) Folgern Sie nun, dass PolyL $\neq$ P gilt.

10. Entscheidbarkeit II

10 Punkte

Betrachten Sie das folgende Problem.

ALMOSTUNIVERSAL

Gegeben: Kodierung w einer DTM mit Eingabealphabet $\{0, 1\}$.

Entscheide: Gibt es ein Wort $x \in \{0, 1\}^*$, sodass $\mathcal{L}(M_w) = \{0, 1\}^* \setminus \{x\}$ gilt?

Beweisen Sie, dass ALMOSTUNIVERSAL weder semi-entscheidbar noch co-semi-entscheidbar ist.

1. Zusatzblatt

2. Zusatzblatt

3. Zusatzblatt

4. Zusatzblatt

5. Zusatzblatt

6. Zusatzblatt