Abschlussklausur
Theoretische Informatik 2
25. März 2021

Prof. Dr. Roland Meyer
Thomas Haas
TU Braunschweig
Wintersemester 2020/2021

1. Lesen und unterschreiben Sie die **Erklärung nach Abschluss der Online-Prüfung**, bevor Sie die Klausur zu bearbeiten beginnen. Laden Sie diese unbedingt in den **Abgabeordner** hoch, Ihre Klausur gilt sonst als **nicht bestanden**.

   Falls Sie die Erklärung weder händisch noch digital unterschreiben können, schreiben Sie eine E-Mail an *t.haas@tu-bs.de* und *a.soleinsky@tu-bs.de*. Geben Sie in dieser E-Mail **alle Informationen an**, die in der Erklärung abgefragt werden.

2. Schreiben Sie **leserlich** und **nummerieren** Sie die Seiten Ihrer Abgabe.

3. **Laden** Sie bis zum Ende der Klausur Ihre **Abgabe** in den dafür vorgesehenen **Abgabeordner hoch**. Sie können Ihre **Abgabe** dazu **abfotografieren**, **einscannen** oder **direkt als PDF** per Tablet oder ähnlichem erstellen. Abgaben als **.pdf, .jpg** oder sonstigen **Standardformaten** sind möglich.

4. Der Dateiname Ihrer Klausur soll MatNr_Nachname sein. Der Dateiname Ihrer Erklärung nach Abschluss der Online-Prüfung soll MatNr_Nachname_Erklaerung sein.

   Beispiel: 4444444_Mustermann_Max.pdf und
   4444444_Mustermann_Max_Erklaerung.pdf

   Falls Sie mehrere Dateien abgeben, fügen Sie entsprechende Suffixe in der Form _Suffix hinzu (z.B. 4444444_Mustermann_Max_2.pdf für die zweite Seite der Abgabe).

5. Sie dürfen das **Skript** und ihre **eigenen Aufzeichnungen** verwenden. Das Heranziehen **fremder Hilfe** (z.B. andere Studenten oder Internetforen) ist **untersagt**.

6. Wenn Sie im Laufe der Klausur **Fragen** haben, steht Ihnen folgender BBB-Raum zur Verfügung:
   `https://webconf.tu-bs.de/tho-6n6-t7e`
   Stellen Sie Ihre Frage über den öffentlichen Chat. **Nach Beantwortung** Ihrer Frage, **verlassen** Sie bitte den BBB-Raum wieder.

7. Im Falle von Auftreten **technischer Probleme**, machen Sie bitte **Beweisfotos** und melden Sie anschließend die Probleme telefonisch unter +49 531-391-9522.

8. Wir werden den Termin für die Klausureinsicht auf unserer Website bekanntgeben:
   `tcs.cs.tu-bs.de/teaching/TheoInf2Klausur_WS_20202021.html`.

9. Die **Bearbeitungszeit** beträgt **240 Minuten**. 60 Minuten Extrazeit für den technischen Zusatzaufwand sind bereits eingerechnet. Laden Sie die Klausur **rechtzeitig** hoch!

10. Mit **40 Punkten** ist die Klausur **sicher bestanden.**

1. TM-Konstruktion

10 Punkte

Betrachten Sie die Sprache $\mathcal{L} = \{w.u.w \in \{a,b\}^* \mid w,u \in \{a,b\}^*, |w| > 0\}$.

Konstruieren Sie eine NTM M, die diese Sprache akzeptiert.

- 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 $\sqcup$-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.

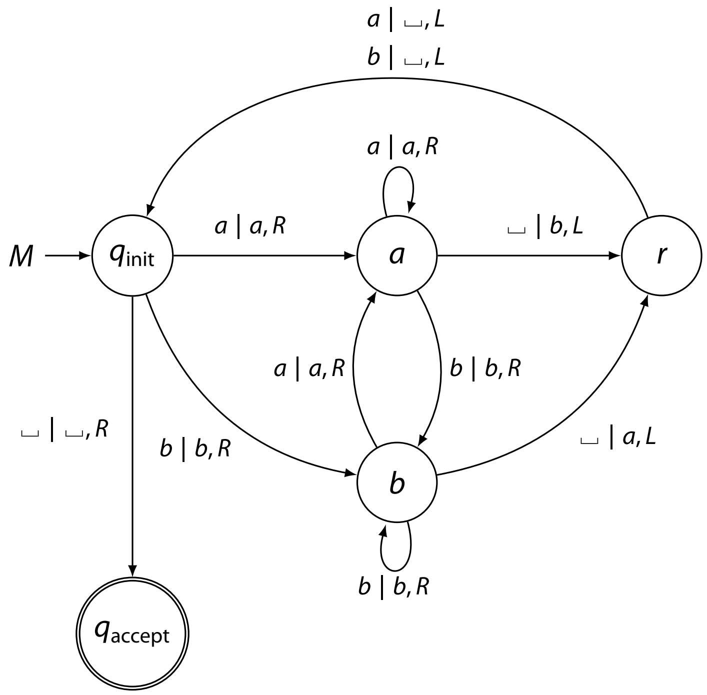
Hinweise: Machen Sie von Nichtdeterminismus Gebrauch. Sie dürfen auch mehrere Bänder verwenden.

Zu Aufgabe 1:

2. TM-Analyse

$5 + 2 + 3 = 10$ Punkte

Betrachten Sie die 1-Band TM M mit Eingabealphabet $\Sigma = \{a, b\}$, Bandalphabet $\Gamma = \{a, b, \sqcup\}$ und Zustandsmenge $Q = \{q_{\text{init}}, q_{\text{accept}}, a, b, r\}$. Hierbei ist q_{init} der Startzustand und q_{accept} der akzeptierende Zustand.



a) Beschreiben Sie die Arbeitsweise von M in Worten und geben Sie diese als Pseudo-Code an.

b) Interpretieren Sie M nun als Entscheider. Was ist die von M akzeptierte Sprache?

c) Interpretieren Sie M nun als Berechner. Was ist die von M berechnete Funktion f?

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 2:

3. NP-Vollständigkeit

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

Betrachten Sie das folgende Problem.

Weak Hamilton-Cycle (WHC)	
Gegeben:	Ein gerichteter Graph $G = (V, E)$
Entscheide:	Gibt es einen Kreis in G, der jeden Knoten v mindestens 1 mal, aber maximal 2 mal enthält.

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

a) „Membership“: WHC $\in$ NP.

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

Bemerkung: Der Start- btw. Endknoten des Kreises wird nur ein mal gezählt.

4. NL-Vollständigkeit

5 + 5 = 10 Punkte

Betrachten Sie das folgende Problem.

Strict One-way-reachability (SOWR)

Gegeben: Ein gerichteter Graph $G = (V, E)$ und Knoten s, t.

Entscheide: Gibt es in G einen Pfad von s nach t, aber keinen Pfad von t nach s?

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

a) „Membership“: SOWR $\in$ NL.

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

5. Entscheidbarkeit

10 Punkte

Betrachten Sie die Sprache aller Kodierungen von Turing-Maschinen, deren Sprache unter Spiegelungen invariant sind.

$$\mathcal{L} = \left\{ w \in \{0,1\}^* \mid \mathcal{L}(M_w) = \mathcal{L}(M_w)^{rev} \right\}.$$

Hierbei bezeichnet $\mathcal{L}^{rev}$ die Spiegelsprache von $\mathcal{L}$.

Beweisen Sie, dass $\mathcal{L}$ nicht entscheidbar ist. Benutzen Sie **nicht** den Satz von Rice.

6. Berechenbarkeit

10 Punkte

Es sei $\Sigma = \{0, 1\}$. Betrachten Sie die Funktion *rejectionDegree* : $\Sigma^* \times \Sigma^* \rightarrow \mathbb{N} \cup \{\infty\}$ definiert als

$$\mathit{rejectionDegree}(w, x) := \text{Anzahl der Berechnungen von } M_w \text{ auf } x\text{, die } q_{\text{reject}} \text{ erreichen.}$$

Falls w keine Turing-Maschine kodiert oder es unendlich viele solcher Berechungspfade auf x gibt, liefert die Funktion den Wert ∞ (als ein spezielles Symbol). Wir gehen davon aus, dass die TM (und somit auch der Berechnungspfad) stoppt, sobald q_{reject} erreicht wird.

Zeigen Sie nun, dass die Funktion *rejectionDegree* **nicht berechenbar** ist.

Hinweis: Nehmen Sie an, dass Sie die Funktion berechnen könnten und führen Sie dies zu einem Widerspruch.

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) Es gibt eine NP-vollständige Sprache $\mathcal{L}$, deren Komplement $\overline{\mathcal{L}}$ nicht in PSPACE liegt.

b) Der Satz von Rice lässt sich auf $\mathcal{L} = \{w \in \{0,1\}^* \mid \mathcal{L}(M_w) \text{ ist semi-entscheidbar}\}$ anwenden.

c) Der Satz von Rice lässt sich auf $\mathcal{L} = \{w \in \{0,1\}^* \mid \mathcal{L}(M_w) \text{ ist co-semi-entscheidbar}\}$ anwenden.

d) Angenommen $P \neq NP$, dann gibt es eine kontextfreie Sprache $\mathcal{L}$ mit nicht kontextfreiem Komplement $\overline{\mathcal{L}}$, sodass $\overline{\mathcal{L}}$ NP-schwer ist.

Hinweis: Betrachten Sie den CYK-Algorithmus.

8. Lazy NTM

10 Punkte

Sei *M* eine NTM und w ein Wort. Wir nennen einen Berechnungspfad von *M* auf *w* **haltend**, wenn er in q_{accept} oder in q_{reject} endet.

Eine **lazy NTM** *M* ist wie eine übliche NTM definiert, besitzt aber eine besondere Akzeptanzbedingung: Ein Wort *w* wird von *M* akzeptiert, wenn es unter den **kürzesten haltenden** Berechungspfaden einen akzeptierenden gibt. Falls es gar keine entscheidenen Berechnungspfade gibt, dann wird das Wort wie üblich abgelehnt.

Zeigen Sie, dass lazy NTMs **gleichmächtig** zu üblichen NTMs sind.

9. 3-Row-PCP

3 + 7 Punkte

Betrachten Sie das folgende Problem.

3-Row-PCP

Gegeben: Eine endliche Sequenz von 3-Tupeln aus Wörtern $(x_1, y_1, z_1), \ldots, (x_k, y_k, z_k)$.

Entscheide: Gibt es eine endliche, nicht-leere Sequenz von Indizes $i_1, \ldots, i_n$ mit $x_{i_1} x_{i_2} \ldots x_{i_n} = y_{i_1} y_{i_2} \ldots y_{i_n} = z_{i_1} z_{i_2} \ldots z_{i_n}$

a) Finden Sie eine Reduktion von PCP auf 3-Row-PCP.

b) Finden Sie eine Reduktion von 3-Row-PCP auf PCP.

Hinweis zu b): Suchen Sie keine direkte Reduktion. Reduzieren Sie zuerst zu einem geeigneten Zwischenproblem.

10. Entscheidbarkeit 2

$5 + 5 = 10$ Punkte

Betrachten Sie das folgende Problem.

<u>DUPLICATION-INTOLERANT-TM</u>

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

Entscheide: Immer wenn M_w ein Wort x akzeptiert, dann akzeptiert es keine Potenz x^k ($k \in \mathbb{N}, k > 1$)?

Die k-te Potenz x^k ist hierbei die k-fache Wiederholung von x.

a) Zeigen Sie, dass DUPLICATION-INTOLERANT-TM co-semi-entscheidbar ist.

b) Zeigen Sie, dass DUPLICATION-INTOLERANT-TM unentscheidbar ist. Benutzen Sie **nicht** den Satz von Rice.