Lehrstuhl für Grundlagen der Software-Zuverlässigkeit und der theoretischen Informatik
School of Computation, Information and Technology
Technische Universität München

Eexam
Sticker mit SRID hier einkleben

Hinweise zur Personalisierung:

- Ihre Prüfung wird bei der Anwesenheitskontrolle durch Aufkleben eines Codes personalisiert.
- Dieser enthält lediglich eine fortlaufende Nummer, welche auch auf der Anwesenheitsliste neben dem Unterschriftenfeld vermerkt ist.
- Diese wird als Pseudonym verwendet, um eine eindeutige Zuordnung Ihrer Prüfung zu ermöglichen.

Einführung in die Theoretische Informatik

Klausur: IN0011 / Retake **Datum:** Mittwoch, 4. Oktober 2023
Prüfer: Prof. Javier Esparza **Uhrzeit:** 11:00 – 14:00

Bearbeitungshinweise

- Diese Klausur umfasst **20 Seiten** mit insgesamt **8 Aufgaben**.
  Bitte kontrollieren Sie jetzt, dass Sie eine vollständige Angabe erhalten haben.
- Die Gesamtpunktzahl in dieser Klausur beträgt 100 Punkte.
- Zum Bestehen brauchen Sie 45 Punkte.
- Das Heraustrennen von Seiten aus der Prüfung ist untersagt.
- **Wenn Sie eine Aufgabe nicht im vorgesehenen Platz lösen können, verwenden Sie die zusätzlichen Blätter am Ende der Klausur und kennzeichnen Sie dies deutlich bei der entsprechenden Aufgabe.**
- Als Hilfsmittel sind zugelassen:
  - ein **doppelseitig beschriebenes DIN A4 Merkblatt**
  - ein **analoges Wörterbuch** Deutsch $\leftrightarrow$ Muttersprache **ohne Anmerkungen**
- Mit * gekennzeichnete Teilaufgaben sind ohne Kenntnis der Ergebnisse vorheriger Teilaufgaben lösbar.
- **Es werden nur solche Ergebnisse gewertet, bei denen der Lösungsweg erkennbar ist.** Auch Textaufgaben sind **grundsätzlich zu begründen**, sofern es in der jeweiligen Teilaufgabe nicht ausdrücklich anders vermerkt ist.
- Schreiben Sie weder mit roter / grüner Farbe noch mit Bleistift.
- Schalten Sie alle mitgeführten elektronischen Geräte vollständig aus, verstauen Sie diese in Ihrer Tasche und verschließen Sie diese.
- $\mathbb{N} := \{0, 1, 2, \ldots\}$ und $\mathbb{N}_+ := \mathbb{N} \setminus \{0\}$

Hörsaal verlassen von ________ bis ________ / Vorzeitige Abgabe um ________

Aufgabe 1 Quiz: Reguläre und Kontextfreie Sprachen (10 Punkte)

Für die folgenden Fragen ist keine Begründung gefordert. Sie erhalten die Punkte auf eine Teilaufgabe genau dann wenn Sie alle Antwortmöglichkeiten korrekt angekreuzt haben. Es ist immer mindestens eine Antwortmöglichkeit richtig. Jede Teilaufgabe bringt 2 Punkte.

Kreuzen Sie richtige Antworten an ☒
Kreuze können durch vollständiges Ausfüllen gestrichen werden ■
Gestrichene Antworten können durch nebenstehende Markierung erneut angekreuzt werden ×■

Wählen Sie alle wahren Aussagen.

a) Wir betrachten die Sprache $L := \{a^n b^m c^m \mid n, m \in \mathbb{N}\}$ über dem Alphabet $\{a, b, c\}$.

- ☐ Es gibt einen DPDA, der mit Endzustand akzeptiert und L erkennt.
- ☐ Es gibt einen NFA, der L erkennt.
- ☐ Es gibt einen DPDA, der mit leerem Keller akzeptiert und L erkennt.

b) Wir betrachten Sprachen über dem Alphabet {a, b}.

- ☐ Die Menge der regulären Sprachen ist abzählbar.
- ☐ Die Menge der Sprachen, die nicht kontextfrei sind, ist abzählbar.
- ☐ Die Menge der Sprachen, die kontextfrei aber nicht regulär sind, ist abzählbar.

c) Seien A, B Sprachen über demselben Alphabet.

- ☐ Wenn sowohl A als auch B nicht regulär ist, dann ist auch $A \cup B$ nicht regulär.
- ☐ Wenn A regulär ist und B kontextfrei, dann ist auch $A \cap B$ kontextfrei.
- ☐ Wenn $A \cup B = B$ und B regulär ist, dann ist auch A regulär.

d) Wir definieren $L_1 := \{uv \mid u, v \in \Sigma^* \wedge u \neq v\}$ und $L_2 := \{uv \mid u, v \in \Sigma^* \wedge |u| = |v|\}$ für ein Alphabet Σ.

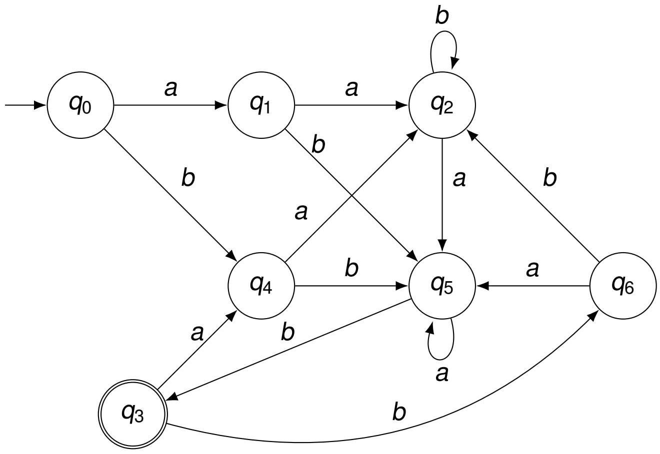
- ☐ L_1 ist regulär.
- ☐ $L_2^* = L_2$.
- ☐ $L_1 \cup L_2$ ist regulär.

e) Wir betrachten die Sprache $L := \{v_1 b w_1 \# v_2 a w_2 \mid v_1, v_2, w_1, w_2 \in \{a, b\}^* \wedge v_1 = v_2 \wedge w_1 = w_2\}$ über dem Alphabet $\Sigma := \{a, b, \#\}$.
Erinnerung: L^w ist die Residualsprache von L bezüglich $w \in \Sigma^*$.

- ☐ $L^{ab\#bb} = L^{a\#b}$.
- ☐ $L^{aba\#ba} = L^{ab\#ba}$.
- ☐ $L^{ab\#} = L^{ba\#}$.

Aufgabe 2 **Minimierung von DFAs (10 Punkte)**

Sei $\Sigma := \{a, b\}$. Wir betrachten den folgenden DFA M:



a)* Minimieren Sie den gegebenen DFA M. Füllen Sie dazu untenstehende Minimierungstabelle nach Vorlesungsalgorithmus aus und zeichnen Sie den resultierenden DFA vollständig.

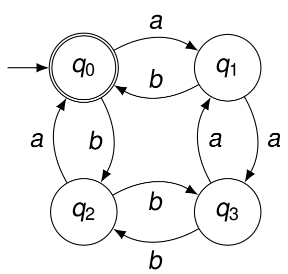
0 1 2 3 4 5 6

	q_0	q_1	q_2	q_3	q_4	q_5	q_6
q_0	–	–	–	–	–	–	–
q_1		–	–	–	–	–	–
q_2			–	–	–	–	–
q_3				–	–	–	–
q_4					–	–	–
q_5						–	–
q_6							–

Resultierender DFA:

0 b)* Wir betrachten den folgenden minimalen DFA:

1



Nach Vorlesung korrespondiert jeder Zustand des minimalen Automaten zu einer Residualsprache. Geben Sie für jeden Zustand das kürzeste Wort an, das in der zum Zustand korrespondierenden Residualsprache liegt. Falls es mehrere passende kürzeste Wörter gibt, geben Sie das alphabetisch kleinste davon an.

Wort für q_0:

Wort für q_1:

Wort für q_2:

Wort für q_3:

c)* Sei $D = (Q, \Sigma, \delta, q_0, F)$ ein DFA. Sei M der minimale DFA für D. Zeigen Sie oder widerlegen Sie folgende Ausssage:

Wenn M eine Schleife enthält, dann enthält D auch eine Schleife.

Ihre Wahl: ☐ Aussage ist wahr ☐ Aussage ist falsch

Beweis:

0 1 2 3

Aufgabe 3 Grammatiken und Sprachen (10 Punkte)

Sei $\Sigma := \{a, b\}$ und $N := \{a^n b^n \mid n \in \mathbb{N}\}$. Wir betrachten die Sprache L der nichtleeren Wörtern, die aus N hervorgehen, indem man **höchstens** ein Zeichen austauscht. Formal definieren wir:

$$L := \{vxw \mid v, w \in \Sigma^* \wedge x \in \Sigma \wedge (\exists x' \in \Sigma.\ vx'w \in N)\}.$$

Beispielsweise gilt $aaab \in L$ und $abab \notin L$. Außerdem betrachen wir die Grammatik G über Σ mit Startsymbol S und folgenden Produktionen:

$$S \rightarrow aSb \mid aTa \mid bTb \qquad T \rightarrow aTb \mid \epsilon$$

0
1
2
3
4

a)* Lösen Sie genau eine der folgenden beiden Aufgaben:

(i) Zeigen Sie $L \not\subseteq L(G)$ per Angabe eines Wortes w mit $w \in L$ und $w \notin L(G)$. Die Korrektheit des Beispiels müssen Sie nicht begründen.

Falls möglich, erweitern Sie außerdem G zu G' mit $L = L(G')$ durch Hinzufügen genau einer Produktion (keine sonstigen Änderungen sind erlaubt). Falls dies nicht möglich ist, begründen Sie dies informell.

(ii) Zeigen Sie $L \subseteq L(G)$ per Induktion über die Wortlänge von Wörtern $w \in L$. Sie dürfen in Ihrem Beweis folgende Eigenschaft verwenden: (1) $N \subseteq L_G(T)$. Kennzeichnen Sie Anwendungen dieser Eigenschaft und der Induktionshypothesen im Beweis deutlich.

Ihre Wahl: ☐ Aufgabe (i) ☐ Aufgabe (ii)

Beweis:

b)* Lösen Sie genau eine der folgenden beiden Aufgaben:

(i) Zeigen Sie $L(G) \not\subseteq L$ per Angabe eines Wortes w mit $w \in L(G)$ und $w \notin L$. Beweisen Sie, dass Ihr Beispiel korrekt ist.

Falls möglich, reduzieren Sie außerdem G zu G' mit $L = L(G')$ durch Entfernen von Produktionen (keine sonstigen Änderungen sind erlaubt). Falls dies nicht möglich ist, begründen Sie dies informell.

(ii) Zeigen Sie $L(G) \subseteq L$ per Induktion über die Erzeugung von Wörtern $w \in L(G)$ (strukturelle Induktion). Den Fall $S \to bTb$ müssen Sie nicht beweisen. Sie dürfen in Ihrem Beweis folgende Eigenschaft verwenden: (1) $L_G(T) \subseteq N$. Kennzeichnen Sie Anwendungen dieser Eigenschaft und der Induktionshypothesen im Beweis deutlich.

0
1
2
3
4
5
6

Ihre Wahl: ☐ Aufgabe (i) ☐ Aufgabe (ii)

Beweis:

Aufgabe 4 CYK und Co (16 Punkte)

a)* Sei G eine Grammatik mit Startsymbol S und folgenden Produktionen:

$$S \to AB \mid BC, \qquad A \to a \mid BA, \qquad B \to b \mid CC, \qquad C \to a \mid AB.$$

Verwenden Sie den aus der Vorlesung bekannten CYK-Algorithmus, um untenstehende CYK-Tabelle für das Wort *bbab* **vollständig** auszufüllen:

$1,4$			
$1,3$	$2,4$		
$1,2$	$2,3$	$3,4$	
$1,1$	$2,2$	$3,3$	$4,4$
b	b	a	b

b) Bestimmen Sie das kürzeste Suffix von *bbab*, das in $L(G)$ enthalten ist. Begründen Sie Ihre Antwort kurz, indem Sie auf die Tabelle aus (a) verweisen.
Erinnerung: ein Wort $u \in \Sigma^*$ ist ein Suffix eines Wortes $w \in \Sigma^*$ falls ein Wort $v \in \Sigma^*$ existiert mit $w = vu$.

Wort:
Begründung:

c) Lesen Sie ein Wort der Länge 3, welches nicht in $L(G)$ enthalten ist, aus der Tabelle aus (a) ab. Begründen Sie Ihre Antwort kurz.

Wort:
Begründung:

In Anlehnung an die Chomsky-Normalform aus der Vorlesung ist eine Grammatik in k-Chomsky-Normalform, falls jede Produktion die Form $X \to a$ oder $X \to X_1 \dots X_k$ hat.

d)* Welche Sprachen können von Grammatiken in 1-Chomsky-Normalform erzeugt werden? Sie müssen Ihre Antwort nicht begründen.

e)* Kann jede Grammatik in 2-Chomsky-Normalform in eine äquivalente Grammatik in 3-Chomsky-Normalform überführt werden? Falls ja, skizzieren Sie kurz einen Algorithmus hierfür. Falls nein, geben Sie ein Gegenbeispiel an und begründen Sie dessen Korrektheit.

0 1 2 3

Ihre Wahl: ☐ Ja, es ist möglich. ☐ Nein, es ist nicht möglich.

Algorithmus bzw. Gegenbeispiel:

f)* Sei H eine Grammatik mit Startsymbol S und den Produktionen

0 1 2 3

$$S \to AbA \mid BC, \qquad A \to a \mid AB, \qquad B \to SB \mid AB \mid CB, \qquad C \to AC \mid c, \qquad D \to AB \mid SB.$$

Bestimmen Sie die erreichbaren, erzeugenden und nützlichen Nichtterminalsymbole von H.

Erreichbar:

Erzeugend:

Nützlich:

Aufgabe 5 **Kontextfreie Sprachen (13 Punkte)**

Sei $L := \{a^n b^n \mid n \in \mathbb{N} \text{ und } n \text{ ist ungerade}\}$.

0 1 2

a)* Geben Sie eine kontextfreie Grammatik für L mit höchstens 4 Produktionen an.

0 1 2 3 4

b) Verwenden Sie den aus der Vorlesung bekannten Algorithmus (CFG→PDA), um Ihre Grammatik aus Aufgabe a) in einen PDA zu überführen, der L mit leerem Keller akzeptiert.
Beachten Sie: Sie können nur Punkte auf diese Teilaufgabe erhalten, wenn Ihre Abgabe im Teil a) vollständig richtig ist.

c)* Sei $N \subseteq \Sigma^*$ eine Sprache und $K = (Q, \Sigma, \Gamma, q_0, Z_0, \delta, F)$ ein Kellerautomat, der die Sprache N mit Endzustand akzeptiert. Wir möchten einen Kellerautomaten K' konstruieren, der die Sprache N^* mit Endzustand akzeptiert.
Untenstehend wird eine entsprechende Konstruktion begonnen. Verwenden Sie die restlichen 7 Zeilen der Tabelle, um die Übergangsrelation geeignet zu ergänzen. Sie müssen nicht alle 7 Zeilen verwenden und Sie dürfen nicht mehr als die 7 restlichen Zeilen verwenden.
Beschreiben Sie die Idee Ihrer Konstruktion außerdem informell!

0 1 2 3 4 5 6 7

Setze $K' := \left(Q \cup \{q_{neu}\}, \Sigma, \Gamma \cup \{Z_{neu}\}, q_{neu}, Z_{neu}, \delta', \{q_{neu}\}\right)$. Definiere δ' wie folgt:

Gleichung für δ'	**Quantifikation der Variablen**
$\delta'(q, a, Z) := \delta(q, a, Z)$	für alle $q \in Q, a \in \Sigma, Z \in \Gamma$
$\delta'($ $):=$	
$\delta'($ $):=$	
$\delta'($ $):=$	
$\delta'($ $):=$	
$\delta'($ $):=$	
$\delta'($ $):=$	
$\delta'($ $):=$	

Konstruktionsidee:
Problem ist wie wir hier festlegen ab wann die entsprechenden Transitionen korrekt sind, da sehr viel Quatsch geschrieben wird in dem man aber noch eine korrekte Idee erraten kann (Schwierig unter den Tutoren konsistent zu halten).
Fehlende "für alle"würde ich keine Punkte abziehen. Wenn Variablen quantifiziert aber nicht verwendet werden auch nicht.
Evtl. kann man Punkte zusammenlegen (zB die ersten 3 in "ϵ wird richtig behandelt")

Aufgabe 6 **Rechts-links-lineare Sprachen (15 Punkte)**

Eine Grammatik $G = (V, \Sigma, P, S)$ ist *rechts-links-linear* (RLL) wenn jede Produktion der Gestalt $X \to aY$, $X \to Ya$, $X \to Y$, $X \to a$, oder $X \to \epsilon$ ist, wobei $X, Y \in V$ und $a \in \Sigma$. Eine Sprache ist *rechts-links-linear* (RLL) wenn sie von einer rechts-links-linearen Grammatik erzeugt wird. Beweisen oder widerlegen Sie jede der folgenden Aussagen:

0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐

a)* Aussage: RLL-Sprachen sind regulär.

Ihre Wahl: ☐ Wahr ☐ Falsch

Beweis:

0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐

b)* Aussage: RLL-Sprachen sind abgeschlossen unter Vereinigung.

Ihre Wahl: ☐ Wahr. ☐ Falsch.

Beweis:

c)* Aussage: RLL-Sprachen sind abgeschlossen unter Schnitt.

Ihre Wahl: ☐ Wahr. ☐ Falsch.

0
1
2
3
4

Beweis:

d)* Aussage: RLL-Sprachen sind abgeschlossen unter Komplement.

Ihre Wahl: ☐ Wahr. ☐ Falsch.

0
1
2
3

Beweis:

Aufgabe 7 Ask Me Anything (10 Punkte)

Für die folgenden Fragen ist keine Begründung gefordert. Sie erhalten die Punkte auf eine Teilaufgabe genau dann wenn Sie alle Antwortmöglichkeiten korrekt angekreuzt haben. Es ist immer mindestens eine Antwortmöglichkeit richtig. Jede Teilaufgabe bringt 2 Punkte.

Kreuzen Sie richtige Antworten an ☒
Kreuze können durch vollständiges Ausfüllen gestrichen werden ■
Gestrichene Antworten können durch nebenstehende Markierung erneut angekreuzt werden ×■

a)* Welche der folgenden Probleme sind entscheidbar?

☐ Gegeben: Eine deterministische Turingmaschine M mit Eingabealphabet Σ und $n \in \mathbb{N}$.
Problem: Hält M in n Schritten für jede beliebige Eingabe $w \in \Sigma^*$?

☐ Gegeben: Eine deterministische Turingmaschine M.
Problem: Besucht M bei leerer Eingabe den Startzustand mindestens zweimal?

☐ Gegeben: Eine deterministische Turingmaschine M und $n \in \mathbb{N}$.
Problem: Gibt es einen Zustand, sodass M bei leerer Eingabe diesen Zustand mindestens n-mal besucht?

b)* Sei $\Sigma := \{0, 1\}$. Welche der folgenden Sprachen sind unentscheidbar?

☐ $\{w \in \Sigma^* \mid w \in L(M_w)\}$.

☐ $\{w \in \Sigma^* \mid |L(M_w)| \leq |w|\}$.

☐ $\{w \in \Sigma^* \mid L(M_w) \text{ ist nicht regulär}\}$.

c)* Seien A, B, C Sprachen über demselben Alphabet Σ. Welche der folgenden Aussagen sind wahr?
Erinnerung: $A \leq B$ bedeutet, dass die Sprache A auf die Sprache B reduzierbar ist.

☐ Wenn $AC \leq BC$, dann $A \leq B$.

☐ $(A \cap B) \leq A$.

☐ Angenommen A ist NP-vollständig und $\overline{A} \in \text{NP}$. Dann gilt für alle $B \in \text{NP}$, dass $\overline{B} \in \text{NP}$.

d)* Aus welchen der folgenden Aussagen würde P = NP folgen?

☐ Für mindestens ein Problem in NP existiert ein polynomiell zeitbeschränkter Algorithmus.

☐ 3-SAT $\leq_p \{0^n 1^n 0^n \mid n \in \mathbb{N}\}$

☐ $\{0^n 1^n 0^n \mid n \in \mathbb{N}\} \leq_p$ 3-SAT

e)* Sei Σ ein Alphabet. Sei $f : \mathcal{P}(\Sigma^*) \times \mathcal{P}(\Sigma^*) \to \mathcal{P}(\Sigma^*)$ eine Funktion von Paaren von Sprachen auf Sprachen.
Wählen Sie alle Aussagen, die zutreffen.
Erinnerung: $\mathcal{P}(A)$ ist die Potenzmenge einer Menge A.

☐ Angenommen $f(A, \overline{B})$ ist semi-entscheidbar für alle semi-entscheidbaren Sprachen A, B. Dann ist auch $f(\overline{A}, B)$ semi-entscheidbar für alle semi-entscheidbaren Sprachen A, B.

☐ Angenommen $f(A, B)$ ist entscheidbar für alle semi-entscheidbare Sprachen A, B. Seien A, B, C semi-entscheidbar. Dann ist $f(A, f(B, C))$ entscheidbar.

☐ Angenommen $f(A, \overline{B})$ ist semi-entscheidbar für alle entscheidbaren Sprachen A, B. Dann ist auch $f(\overline{A}, B)$ semi-entscheidbar für alle entscheidbaren Sprachen A, B.

Aufgabe 8 **Reduktion (16 Punkte)**

Zwei Formeln über denselben Variablen sind *äquivalent*, wenn sie dieselben erfüllenden Belegungen haben, d.h. jede Belegung erfüllt beide Formeln oder keine.
Nehmen Sie P $\neq$ NP an. Entscheiden Sie unter dieser Annahme, welche der folgenden Probleme in P und welche NP-vollständig sind.
Wenn das Problem in P liegt, beschreiben Sie einen polynomiell zeitbeschränkten Algorithmus und begründen Sie die Korrektheit und polynomielle Beschränktheit des Algorithmus kurz.
Wenn das Problem NP-vollständig ist, geben Sie eine Reduktion von 3KNF-SAT an. Die Reduktionsfunktion muss präzise definiert sein. Die Korrektheit der Reduktion müssen Sie aber nur skizzieren und nicht formal beweisen.

a)* **Gegeben:** Eine KNF-Formel F.
Problem: Ist $\neg F$ erfüllbar?

0 1 2 3 4

Ihre Wahl: ☐ P ☐ NP-vollständig

Algorithmus bzw. Reduktion:

0 1 2 3 4

b)* **Gegeben:** Zwei KNF-Formeln F und G über denselben Variablen.
Problem: Sind F und G nicht äquivalent?

Ihre Wahl: ☐ P ☐ NP-vollständig

Algorithmus bzw. Reduktion:

0 1 2 3 4

c)* **Gegeben:** Eine KNF-Formel F.
Problem: Gibt es eine erfüllbare Belegung, die höchstens 3 oder mindestens $v - 3$ Variablen wahr macht, wobei v die Anzahl der Variablen von F ist?

Ihre Wahl: ☐ P ☐ NP-vollständig

Algorithmus bzw. Reduktion:

d)* **Gegeben:** Eine KNF-Formeln *F* mit einer geraden Anzahl von Variablen.
Problem: Gibt es eine erfüllbare Belegung, die genau die Hälfte der Variablen von *F* wahr macht?

Ihre Wahl: ☐ P ☐ NP-vollständig

0 1 2 3 4

Algorithmus bzw. Reduktion:

Zusätzlicher Platz für Lösungen. Markieren Sie deutlich die Zuordnung zur jeweiligen Teilaufgabe. Vergessen Sie nicht, ungültige Lösungen zu streichen.