

Klausur zur Vorlesung Formale Sprachen und Komplexität

Die Bearbeitungszeit beträgt **90 Minuten**. Hilfsmittel sind nicht erlaubt, auch der Besitz ausgeschalteter elektronischer Geräte wird als Betrug gewertet. Schreiben Sie Ihren vollständigen Namen und Ihre Matrikelnummer deutlich lesbar auf dieses Deckblatt, sowie Ihren Namen in die Kopfzeile auf **jedem Blatt** der Klausurangabe! Geben Sie alle Blätter ab, lassen Sie diese zusammengeheftet! Verwenden Sie nur **dokumentenechte Stifte** und **weder** die Farben **rot noch grün**.

Kontrollieren Sie, ob Sie alle Aufgabenblätter erhalten haben. Aufgabenstellungen befinden sich auf den **Seiten 1–12**. Sie dürfen die Rückseiten für Nebenrechnungen nutzen. Falls Sie die Rückseiten für Antworten nutzen, so markieren Sie klar, was zu welcher Aufgabe gehört und schreiben Sie in der entsprechenden Aufgabe, wo alle Teile Ihrer Antwort zu finden sind. Streichen Sie alles durch, was nicht korrigiert werden soll!

Lesen Sie die Aufgabenstellungen vollständig durch, bevor Sie mit der Bearbeitung der Aufgaben beginnen. Es gibt 4 unterschiedlich gewichtete Aufgaben zu insgesamt 90 Punkten. Mit 45 Punkten haben Sie sicherlich bestanden. Die Teilaufgaben können unabhängig voneinander bearbeitet werden. Mit Ihrer Unterschrift bestätigen Sie, dass Sie zu Beginn der Klausur in ausreichend guter gesundheitlicher Verfassung sind, und diese Klausurprüfung verbindlich annehmen.

Nachname:

Vorname:

Matrikelnummer:

Studiengang:

Die Klausur soll gewertet werden: **JA** **NEIN**

Hinweis: Ist keines der beiden Felder angekreuzt, so wird **JA** angenommen.

Hiermit erkläre ich die Richtigkeit der obigen Angaben:

(Unterschrift)

Die folgende Tabelle **NICHT** ausfüllen:

Aufgabe	1	2	3	4	Σ	Bonus	Note
Punkte	34	28	16	12	90		
erreicht							

Aufgabe 1 (Reguläre Sprachen):**(34 Punkte)**

a) Die Sprache $L_1 \subseteq \{a, b\}^*$ sei definiert als:

$$L_1 := \{uabbav \mid u, v \in \{a, b\}^*\}$$

L_1 enthält daher genau alle Wörter, die $abba$ als Teilwort haben.

Geben Sie den Zustandsgraphen eines nichtdeterministischen endlichen Automaten (NFA) A_1 an, für den gilt: $L(A_1) = L_1$.

Fortsetzung von Aufgabe 1:

b) Die Sprache $L_2 \subseteq \{a, b\}^*$ sei definiert als:

$$\begin{aligned}L_2 &:= L_3 \cup L_4 \\L_3 &:= \{w \mid \#_a(w) \geq 2\} \\L_4 &:= \{w \mid \#_b(w) \leq 3\}\end{aligned}$$

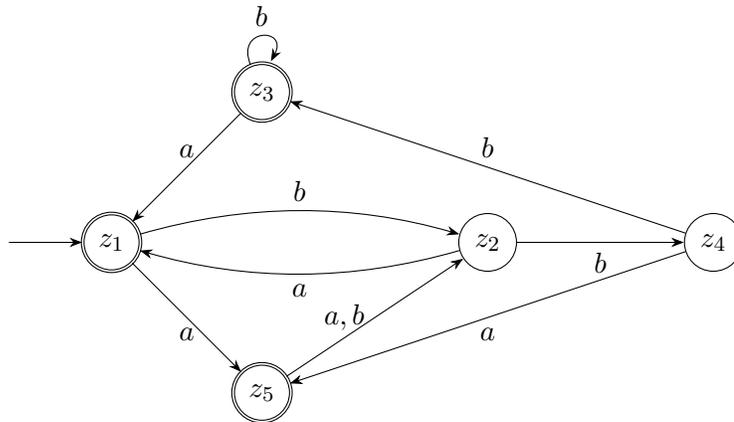
Die Sprache L_2 enthält daher alle Wörter, die mindestens zwei a 's oder höchstens drei b 's enthalten.

Geben Sie einen regulären Ausdruck α mit $L(\alpha) = L_2$ an.

c) Bestimmen Sie den Nerode-Index der Sprache $L_4 := \{w \mid \#_b(w) \leq 3\}$ und begründen Sie Ihre Antwort.

Fortsetzung von Aufgabe 1:

d) Der folgende Automat A_2



ist bereits ein Minimalautomat mit Eingabealphabet $\Sigma = \{a, b\}$.

Begründen Sie für jedes der Zustandspaare

- (i) (z_1, z_2)
- (ii) (z_1, z_3)
- (iii) (z_2, z_4)

warum die beiden Zustände des Paares **nicht äquivalent** sind.

Fortsetzung von Aufgabe 1:

e) Sei $N = (Z_1, \Sigma, \delta_1, S_1, E_1)$ ein NFA und $D = (Z_2, \Sigma, \delta_2, z_{0,2}, E_2)$ ein DFA.

Erläutern Sie, wie man einen endlichen Automaten M (dieser kann ein DFA oder ein NFA sein) aus den Automaten N und D konstruieren kann, so dass gilt:

$$L(M) = L(N) \cap L(D)$$

Begründen Sie die Korrektheit Ihrer Konstruktion.

Der Automat M soll dabei $O(|Z_1| \cdot |Z_2|)$ Zustände besitzen. Wenn Ihre Konstruktion korrekt ist, aber mehr Zustände benutzt, erhalten Sie noch Teilpunkte.

Fortsetzung von Aufgabe 1:

Aufgabe 2 (Kontextfreiheit):**(28 Punkte)**

a) Die Sprache L_5 sei definiert als

$$L_5 := \{w \mid w = a^m w', w' \in \{a, b\}^*, m = \#_a(w'), m \geq 0\}$$

Informell: Alle Worte aus L_5 lassen sich in zwei Teilworte $w_1 w_2$ aufteilen, so dass w_1 nur aus a 's besteht und w_2 genauso viele a 's wie w_1 enthält.

- (i) Geben Sie eine **kontextfreie Grammatik** G an, die L_5 erzeugt (d.h. $L(G) = L_5$) und erläutern Sie, warum G die Sprache L_5 erzeugt: Beschreiben Sie beispielsweise, welche „Aufgabe“ die einzelnen Nichtterminale bei der Erzeugung übernehmen.

Geben Sie zusätzlich eine **Linksableitung** für das Wort $aabbbbaa$ für Ihre Grammatik an.

Fortsetzung von Aufgabe 2:

(ii) Zeigen Sie mit dem Pumpinglemma für reguläre Sprachen, dass

$$L_5 = \{w \mid w = a^m w', w' \in \{a, b\}^*, m = \#_a(w'), m \geq 0\}$$

nicht regulär ist.

Zur Erinnerung:

Pumpinglemma für reguläre Sprachen.

Sei L eine reguläre Sprache. Dann gibt es eine Zahl $n \in \mathbb{N}_{>0}$, sodass jedes Wort $z \in L$, das Mindestlänge n hat (d.h. $|z| \geq n$), als $z = uvw$ geschrieben werden kann, so dass gilt:

i) $|uv| \leq n$,

ii) $|v| \geq 1$ und

iii) für alle $i \geq 0$: $uv^i w \in L$.

Fortsetzung von Aufgabe 2:

- b) Für eine kontextfreie Grammatik G mit Startsymbol S und Variablen $\{S, T, A, B, C, D, E, F\}$ wurde der CYK-Algorithmus für das Wort $abaaca$ ausgeführt. Der Algorithmus hat dabei die folgende Tabelle für die $V(i, j)$ -Mengen berechnet:

Wort	a	b	a	a	c	a
$j \setminus i$	1	2	3	4	5	6
1	T, A	B, D, F	T, A	T, A	C	T, A
2	$S, B, C,$ D, E, F			S, B, C		
3	T		S, B, C	T		
4			T			
5	S, T					
6	T					

- (i) Geben Sie anhand obiger Tabelle fünf verschiedene Wörter an, die **nicht** in $L(G)$ liegen und begründen Sie Ihre Antwort.
- (ii) Geben Sie anhand obiger Tabelle drei verschiedene Wörter an, die in $L(G)$ liegen und begründen Sie Ihre Antwort.

Aufgabe 3 (Berechenbarkeit und PCP):**(16 Punkte)**

Wir erinnern zunächst an die Definition des 01-Postschen Korrespondenzproblems:

Sei $\Sigma = \{0, 1\}$ und sei K eine Folge von Wortpaaren $K = ((x_1, y_1), \dots, (x_k, y_k))$ mit $x_i, y_i \in \Sigma^+$. Das 01-Postsche Korrespondenzproblem (01-PCP) ist die Frage, ob es für die gegebene Folge K eine Folge von Indizes i_1, \dots, i_m mit $i_j \in \{1, \dots, k\}$ gibt, sodass $x_{i_1} \cdots x_{i_m} = y_{i_1} \cdots y_{i_m}$ gilt.

In der Vorlesung wurde die Unentscheidbarkeit des 01-PCP gezeigt.

Für ein Wort $a_1 \cdots a_n$ mit $a_i \in \{0, 1\}$ und $n \geq 0$ ist das Komplement $C(a_1 \cdots a_n)$ definiert als $C(a_1) \cdots C(a_n)$ wobei

$$C(x) := \begin{cases} 1, & \text{falls } x = 0 \\ 0, & \text{falls } x = 1 \end{cases}$$

Das Komplement-Postsche Korrespondenzproblem (CPCP) ist definiert als

Sei $\Sigma = \{0, 1\}$ und sei K eine Folge von Wortpaaren $K = ((x_1, y_1), \dots, (x_k, y_k))$ mit $x_i, y_i \in \Sigma^+$. Das Komplement-Postsche Korrespondenzproblem (CPCP) ist die Frage, ob es für die gegebene Folge K eine Folge von Indizes i_1, \dots, i_m mit $i_j \in \{1, \dots, k\}$ gibt, sodass $C(x_{i_1} \cdots x_{i_m}) = y_{i_1} \cdots y_{i_m}$ gilt.

a) Es sei $K = ((x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4))$ mit

$$\begin{aligned} (x_1, y_1) &= (00, 111) \\ (x_2, y_2) &= (00, 0) \\ (x_3, y_3) &= (11, 0) \\ (x_4, y_4) &= (1, 100) \end{aligned}$$

Alternativ in der Spielstein-Schreibweise

$$K = \left(\begin{bmatrix} x_1 \\ y_1 \end{bmatrix}, \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}, \begin{bmatrix} x_3 \\ y_3 \end{bmatrix}, \begin{bmatrix} x_4 \\ y_4 \end{bmatrix} \right) = \left(\begin{bmatrix} 00 \\ 111 \end{bmatrix}, \begin{bmatrix} 00 \\ 0 \end{bmatrix}, \begin{bmatrix} 11 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 100 \end{bmatrix} \right)$$

Zeigen Sie, dass $K \in \text{CPCP}$ gilt.

(Hinweis: Es gibt Lösungen, die nicht mehr als vier Indizes verwenden).

Fortsetzung von Aufgabe 3:

b) Zeigen Sie die Unentscheidbarkeit von CPCP, indem Sie 01-PCP auf CPCP reduzieren.

Aufgabe 4 (Gemischte Fragen):**(12 Punkte)**

Pro Frage gibt es **genau eine richtige** Antwort.

Punkte in dieser Aufgabe werden nach folgender Tabelle vergeben. Eine Antwort, bei der 0 oder mindestens 2 Kästchen angekreuzt sind, zählt als falsch.

Richtige Antworten	0	1	2	3	4	5	6
Erzielte Punkte	0	1	3	6	8	10	12

Seien L_6 und L_7 formale Sprachen. $\text{Index}(\sim_L)$ bezeichne den Index der Nerode-Relation von L .

- Wenn $\text{Index}(\sim_{L_6})$ und $\text{Index}(\sim_{L_7})$ beide endlich sind, dann ist $\text{Index}(\sim_{\overline{L_6 \cap L_7}})$ endlich.
- Mit $m_1 = \text{Index}(\sim_{L_6})$ und $m_2 = \text{Index}(\sim_{L_7})$ gilt $\text{Index}(\sim_{L_6 \cup L_7}) = \max(m_1, m_2)$, wobei $\max(\infty, m) = \infty = \max(m, \infty)$ für alle $m \in \mathbb{N} \cup \{\infty\}$.
- Wenn $L_6 \subseteq L_7$, dann gilt $\text{Index}(\sim_{L_6}) \leq \text{Index}(\sim_{L_7})$, wobei $m \leq \infty$ für alle $m \in \mathbb{N} \cup \{\infty\}$.
- Wenn $\text{Index}(\sim_{L_7}) = 1$ und $\text{Index}(\sim_{L_6 \cup L_7}) = 1$, dann gilt $L_6 = L_7$.

μ -rekursive Funktionen sind manchmal

- nicht WHILE-berechenbar
- nicht LOOP-berechenbar
- nicht GOTO-berechenbar
- nicht Turing-berechenbar

Das Präfixfreie Postsche Korrespondenzproblem (PPCP) ist eine Variante des PCP-Problems, welche nur solche Eingaben $((x_1, y_1), \dots, (x_k, y_k))$ verwendet, sodass

für alle $i \in \{1, \dots, k\}$ gilt: x_i ist kein Präfix von y_i und y_i ist kein Präfix von x_i .

Eingaben, die diese Bedingung verletzen, liegen nicht in PPCP. Andere Eingaben liegen in PPCP genau dann, wenn sie in PCP liegen. Es gilt:

- PPCP ist unentscheidbar, aber semi-entscheidbar
- PPCP ist entscheidbar, aber nicht semi-entscheidbar
- PPCP ist unentscheidbar und nicht semi-entscheidbar
- $\text{PPCP} \in \mathcal{P}$

Fortsetzung von Aufgabe 4:

Wenn $L_8 \in \mathcal{NP}$, dann

- ist L_8 LOOP-berechenbar
 - gilt $L_8 \notin \mathcal{P}$
 - führt jede nichtdeterministische Turingmaschine M mit $L(M) = L_8$ mehr als polynomiell viele Schritte aus, d.h. für jedes Polynom $p(n)$ gilt $\text{ntime}_M(|w|) > p(|w|)$ für alle $w \in \Sigma^*$
 - führt jede deterministische Turingmaschine M mit $L(M) = L_8$ höchstens polynomiell viele Schritte aus, d.h. es gibt ein Polynom $p(n)$, sodass $\text{ntime}_M(|w|) \leq p(|w|)$ für alle $w \in \Sigma^*$ gilt
-

Die Grammatik $G = (\{A, B, C, D\}, \{a, b\}, P, A)$ mit

$$P = \{A \rightarrow a, \\ A \rightarrow BC, \\ B \rightarrow b, \\ B \rightarrow C, \\ BC \rightarrow CD, \\ D \rightarrow b, \\ D \rightarrow B\}$$

ist

- in Chomsky-Normalform
 - in Kuroda-Normalform
 - vom Typ 2, aber nicht vom Typ 3
 - vom Typ 0, aber nicht vom Typ 1
-

Das LOOP-Programm

```
 $x_0 := x_2 + 5;$   
loop  $x_1$  do  
     $x_0 := x_0 + 2$   
end
```

berechnet die Funktion f mit

- $f(x) = 2 \cdot x$
 - $f(x, y) = 2 \cdot x + 5$
 - $f(x, y, z) = x \cdot y + 5 \cdot z$
 - $f(x, y) = 2 \cdot x + y + 5$
-

