

Lösung zur Nachholklausur zur Vorlesung
Theoretische Informatik für Medieninformatiker

Die Bearbeitungszeit beträgt **90 Minuten**. Hilfsmittel sind nicht erlaubt, auch das Mitführen 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 1–9**. Sie dürfen die Rückseiten für Nebenrechnungen nutzen. Falls Sie die Rückseiten für Antworten nutzen, so markieren Sie klar, was zu welcher Aufgabe gehört und geben Sie in der entsprechenden Aufgabe an, 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 75 Punkten. Mit 38 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 nicht korrigiert werden

Hinweis: Ist das Kästchen **nicht** angekreuzt, wird die Klausur **gewertet**.

Hiermit erkläre ich die Richtigkeit der obigen Angaben:

(Unterschrift)

Die folgende Tabelle **nicht** ausfüllen:

Aufgabe	1	2	3	4	Σ	Bonus	Note
Punkte	29	18	16	12	75		
Erreicht							

Lösung Aufgabe 1 (Reguläre Sprachen):**(29 Punkte)**

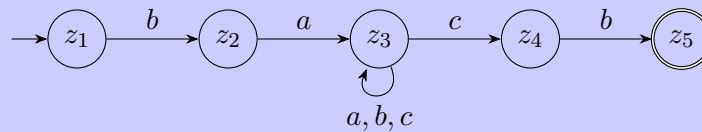
a) Die Sprache $L_1 \subseteq \{a, b, c\}^*$ sei definiert als

$$L_1 := \{baucb \mid u \in \{a, b, c\}^*\}$$

L_1 enthält daher genau alle Wörter über $\{a, b, c\}$, die mit ba anfangen und cb enden.

Geben Sie den Zustandsgraphen eines nichtdeterministischen endlichen Automaten (ohne ε -Übergänge) A_1 an, für den gilt: $L(A_1) = L_1$. (7 Punkte)

LÖSUNG: A_1 :



7 Punkte, wenn alles stimmt

1 Punkt Abzug bei fehlendem Startzustand

1 Punkt Abzug bei fehlendem Endzustand

1 Punkt Abzug bei falschem Übergang

b) Die Sprache L_2 sei definiert als

$$L_2 := \{w \in \{a, b, c, d\}^* \mid \#_b(w) \geq 2\}$$

Die Sprache L_2 enthält daher alle Wörter über $\{a, b, c, d\}$, die mindestens zwei b 's enthalten.

Geben Sie einen regulären Ausdruck α mit $L(\alpha) = L_2$ an. (5 Punkte)

LÖSUNG: Zum Beispiel:

$$\alpha = (a|c|d)^*b(a|c|d)^*b(a|b|c|d)^*$$

oder

$$\alpha = (a|b|c|d)^*b(a|b|c|d)^*b(a|b|c|d)^*$$

2 Punkte Abzug falls ≤ 2 oder $= 2$ statt ≥ 2

1 Punkt Abzug für jeden fehlenden/zusätzlichen $(a|b|c|d)^*$ Block

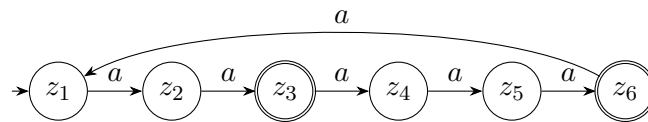
1 Punkt Abzug für jeden fehlenden/zusätzlichen b Block

2 Punkte Abzug für Sprachen-Ausdruck statt Regex (z.B. $\{a, c, d\}^*$ statt $(a|c|d)^*$)

Fortsetzung von Aufgabe 1:

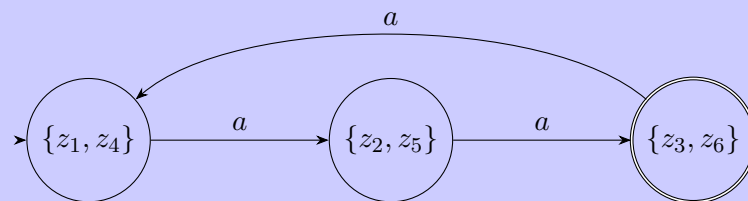
- c) Berechnen Sie für den folgenden deterministischen endlichen Automaten A_2 (der Wörter über dem Alphabet $\Sigma = \{a\}$ verarbeitet) den **Minimalautomaten**, d.h. einen deterministischen endlichen Automaten, der die gleiche Sprache akzeptiert und eine minimale Anzahl an Zuständen benutzt.

Erläutern Sie Ihre Berechnung, indem Sie die Minimierungstabelle angeben, und geben Sie den Minimalautomaten als Zustandsgraphen an.



(11 Punkte)

LÖSUNG: Minimaler DFA



Berechnung:

z_2	X_1				
z_3	X	X			
z_4		X_1	X		
z_5	X_1		X	X_1	
z_6	X	X		X	X
	z_1	z_2	z_3	z_4	z_5

Notation: X : Als verschieden erkannt in der Initialisierung.

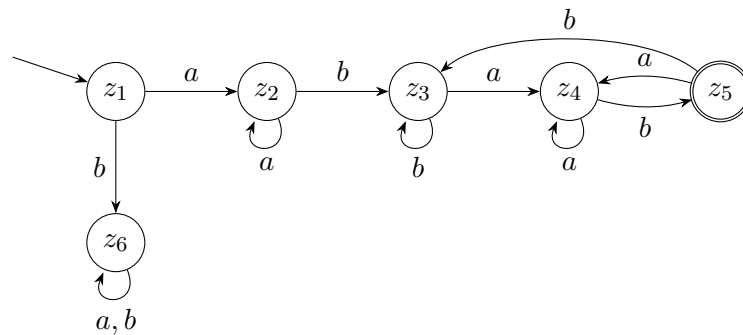
X_1 : Als verschieden erkannt in der ersten Iteration.

- 7 Punkte für die Tabelle:
1 Punkt Abzug per falschem Tabelleneintrag
- 4 Punkte für den Minimalautomaten:
1 Punkt Abzug, wenn Start- und/oder Endzustand nicht gekennzeichnet sind
1 Punkt Abzug für jeden falschen Übergang (bezüglich der Tabelle)

Die Aufgabe wird auf dem nächsten Blatt fortgesetzt.

Fortsetzung von Aufgabe 1:

d) Der folgende Automat A_3



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

Begründen Sie für jedes der Zustandspaare

- (i) (z_4, z_5)
- (ii) (z_1, z_2)
- (iii) (z_3, z_6)

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

(6 Punkte)

LÖSUNG: Zustände z, z' sind äquivalent g.d.w. $\forall w \in \Sigma^* : \widehat{\delta}(z, w) \in E \iff \widehat{\delta}(z', w) \in E$.

- (i) (z_4, z_5) : $\widehat{\delta}(z_4, \varepsilon) = z_4 \notin E$ aber $\widehat{\delta}(z_5, \varepsilon) = z_5 \in E$
- (ii) (z_1, z_2) : $\widehat{\delta}(z_1, bab) = z_6 \notin E$ aber $\widehat{\delta}(z_2, bab) = z_5 \in E$
- (iii) (z_3, z_6) : $\widehat{\delta}(z_3, ab) = z_5 \in E$ aber $\widehat{\delta}(z_6, ab) = z_6 \notin E$ oder
 $\widehat{\delta}(z_3, b) = z_4 \notin E$ aber $\widehat{\delta}(z_6, b) = z_3 \in E$

2 Punkte pro Antwort

Lösung Aufgabe 2 (Kontextfreiheit):**(18 Punkte)**

a) Die Sprache L_3 über $\{a, b, c, d\}$ sei definiert als

$$L_3 := \{a^i b^j c^j d^k \mid i, j, k \in \mathbb{N}_{>0}\}$$

Informell: Alle Wörter aus L_3 sind von der Form $a^i b^j c^j d^k$, wobei i, j, k positive Zahlen sind.

Geben Sie eine **kontextfreie Grammatik** G_1 an, die L_3 erzeugt (d.h. $L(G_1) = L_3$) und erläutern Sie kurz, warum G_1 die Sprache L_3 erzeugt: Beschreiben Sie beispielsweise, welche „Aufgabe“ die einzelnen Nichtterminale bei der Erzeugung übernehmen. (5 Punkte)

Geben Sie zusätzlich eine **Rechtsableitung** für das Wort $abbccdd$ für Ihre Grammatik an. (3 Punkte)

LÖSUNG: $G_1 = (V, \{a, b, c, d\}, P, S)$ mit $V = \{S, X, Y, Z\}$ und $P = \{S \rightarrow XYZ, X \rightarrow aX, X \rightarrow a, Y \rightarrow bYc, Y \rightarrow bc, Z \rightarrow dZ, Z \rightarrow d\}$. Erläuterung:

- S erzeugt zunächst XYZ . Aus X lassen sich beliebig viele a 's erzeugen (jedoch mindestens 1). Aus Y lassen sich beliebig viele b/c -Paare erzeugen (jedoch mindestens 1). Aus Z lassen sich beliebig viele d 's erzeugen (jedoch mindestens 1).

Rechtsableitung: $S \Rightarrow XYZ \Rightarrow XYdZ \Rightarrow XYdd \Rightarrow XbYcdd \Rightarrow Xbbccdd \Rightarrow abbccdd$

5 Punkte für die Grammatik: 3 Punkte für die Grammatik (als Tupel oder Text angegeben), 2 Punkte für die Erläuterung

Maximal 1 Punkt falls Grammatik ganz falsch.

Maximal 1 Punkt für Erläuterung bei falscher Grammatik.

1 Punkt Abzug falls das Startsymbol nicht angegeben ist.

3 Punkte für die Rechtsableitung: 1 Punkt falls keine Rechtsableitung oder falls Syntaxbaum.

Fortsetzung von Aufgabe 2:

b) Sei $G_2 = (\{S, T, U\}, \{a, b\}, P, S)$ mit

$$P = \{S \rightarrow SS \mid TT \mid UU$$

$$T \rightarrow ST \mid a$$

$$U \rightarrow TU \mid b\}$$

eine kontextfreie Grammatik.

Entscheiden Sie, ob $baabaa \in L(G_2)$, indem Sie den Algorithmus von Cocke, Younger und Kasami (CYK-Algorithmus) ausführen. Geben Sie die dabei entstehende Tabelle und die Ausgabe des Algorithmus an und dokumentieren Sie Ihren Rechenweg. (10 Punkte)

LÖSUNG:

Der Algorithmus hat die folgende Tabelle für die $V(i, j)$ -Mengen berechnet:

Wort	b	a	a	b	a	a
$j \setminus i$	1	2	3	4	5	6
1	U	T	T	U	T	T
2		S	U		S	
3		U				
4	S					
5	T					
6	S					

Nachdem $S \in V(1, 6)$ liegt, ist $baabaa \in L(G_2)$.

2 Punkte für richtige Antwort

8 Punkte für Tabelle

1 Punkt Abzug per falsche Zelle

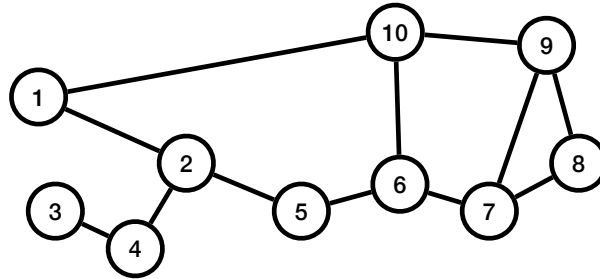
Lösung Aufgabe 3 (Komplexitätstheorie):**(16 Punkte)**

Wir erinnern zunächst an die Definition des INDEPENDENT-SET-Problems:

gegeben: Ein ungerichteter Graph $G = (V, E)$ und eine Zahl $k \in \mathbb{N}$
 gefragt: Besitzt G eine unabhängige Knotenmenge der Größe mindestens k ?

Für einen ungerichteten Graphen $G = (V, E)$ ist $V' \subseteq V$ eine *unabhängige Knotenmenge*, wenn keine zwei Knoten aus V' über eine Kante verbunden sind, d.h. $u, v \in V' \implies \{u, v\} \notin E$.

a) Es sei G_1 der folgende Graph und $k = 4$:



Zeigen Sie, dass $(G_1, k) \in \text{INDEPENDENT-SET}$ gilt.

(5 Punkte)

LÖSUNG: Es gibt viele mögliche Lösungen, z.B. $V' = \{1, 3, 5, 7\}$.

5 Punkte für V'

2 Punkte Abzug für jeden Fehler (zwei benachbarte Knoten in V')

3 Punkte Abzug für weniger als 4 Elemente in V'

Fortsetzung von Aufgabe 3:

- b) Es sei G_1 derselbe Graph wie in Frage a) und $k = 10$. Gilt $(G_1, k) \in \text{INDEPENDENT-SET}$?
Begründen Sie kurz Ihre Antwort. (3 Punkte)

LÖSUNG: Nein. Dann ist $V' = V$, und es gibt z.B. $1, 2 \in V'$ mit $\{1, 2\} \in E$.

2 Punkte für richtige Antwort
1 Punkt für kurze Begründung

Fortsetzung von Aufgabe 3:

- c) Der folgende Beweis ist falsch. Lokalisieren Sie den Fehler und begründen Sie, wieso der Beweis falsch ist. (8 Punkte)

Satz: INDEPENDENT-SET ist in \mathcal{NP} .

Beweis: Wir zeigen $\text{CLIQUE} \leq_p \text{INDEPENDENT-SET}$. Da CLIQUE in \mathcal{NP} liegt, liegt auch INDEPENDENT-SET in \mathcal{NP} .

Sei $f((V, E, m)) = (V, \bar{E}, m)$ wobei $\bar{E} = \{\{u, v\} \mid u, v \in V, \{u, v\} \notin E\}$. Dann gilt: (V, E) hat eine Clique der Größe m g.d.w. (V, \bar{E}) eine unabhängige Knotenmenge der Größe m hat. Da die Funktion f in Polynomialzeit berechnet werden kann, gilt $\text{CLIQUE} \leq_p \text{INDEPENDENT-SET}$.

LÖSUNG: Der Fehler liegt im ersten Teil: „Wir zeigen $\text{CLIQUE} \leq_p \text{INDEPENDENT-SET}$. Da CLIQUE in \mathcal{NP} liegt, liegt auch INDEPENDENT-SET in \mathcal{NP} .“ Um \mathcal{NP} zu beweisen, geht die Reduktion andersherum.

4 Punkte für Ort des Fehlers korrekt identifiziert

4 Punkte für Begründung

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

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

Jede Antwort ist 2 Punkte wert. Eine Antwort, bei der 0 oder mindestens 2 Kästchen angekreuzt sind, zählt als falsch.

Das CLIQUE-Problem ist in \mathcal{NP} . Zudem ist Folgendes bekannt:

- CLIQUE $\in \mathcal{P}$
- CLIQUE $\notin \mathcal{P}$
- CLIQUE ist \mathcal{NP} -schwer
- Wenn $\mathcal{P} \neq \mathcal{NP}$, dann CLIQUE $\in \mathcal{P}$

Sei L_4 eine Sprache mit $\text{SAT} \leq_p L_4$. Dann ist Folgendes bekannt:

- L_4 ist in \mathcal{P}
- $\overline{L_4}$ ist in \mathcal{P}
- L_4 ist in \mathcal{NP}
- L_4 ist \mathcal{NP} -schwer

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

$$P = \{a \rightarrow B, \\ bA \rightarrow A\}$$

ist

- vom Typ 0, aber nicht vom Typ 1
- vom Typ 1, aber nicht vom Typ 2
- vom Typ 2, aber nicht vom Typ 3
- vom Typ 3

Fortsetzung von Aufgabe 4:

Kontextfreie Sprachen

- sind **nicht** bezüglich Vereinigung abgeschlossen
 - sind bezüglich Kleeneschem Abschluss abgeschlossen
 - sind bezüglich Komplement abgeschlossen
 - sind bezüglich Schnitt abgeschlossen
-

Sei L_5 eine kontextsensitive Sprache.

- Dann gibt es einen DFA, der L_5 akzeptiert.
 - Dann gibt es einen NFA, der L_5 akzeptiert, aber keinen DFA.
 - Dann gibt es einen Kellerautomaten, der L_5 akzeptiert, aber keinen NFA oder DFA.
 - Dann gibt es eine Turingmaschine, die L_5 akzeptiert, aber keinen Kellerautomaten, NFA oder DFA.
-

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

$$P = \{A \rightarrow a, \\ A \rightarrow BC, \\ B \rightarrow b, \\ bB \rightarrow CC, \\ D \rightarrow b, \\ aD \rightarrow BC\}$$

ist

- vom Typ 0, aber nicht vom Typ 1
- vom Typ 1, aber nicht vom Typ 2
- vom Typ 2, aber nicht vom Typ 3
- vom Typ 3