2. Klausur zur Vorlesung
Theoretische Grundlagen der Informatik
Wintersemester 2014/2015

Hier Aufkleber mit Name und Matrikelnummer anbringen

Vorname: ______________________

Nachname: ______________________

Matrikelnummer: ______________________

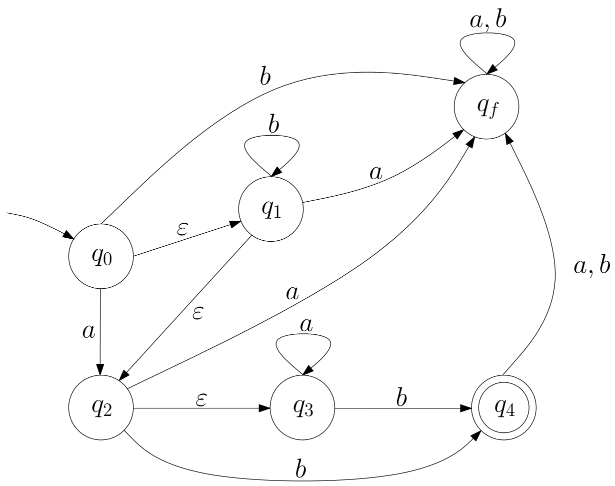
Beachten Sie:

- Bringen Sie den Aufkleber mit Ihrem Namen und Matrikelnummer auf diesem Deckblatt an und beschriften Sie jedes Aufgabenblatt mit Ihrem Namen und Matrikelnummer.
- Schreiben Sie die Lösungen auf die Aufgabenblätter und Rückseiten. Zusätzliches Papier erhalten Sie bei Bedarf von der Aufsicht.
- Zum Bestehen der Klausur sind **20** der möglichen **60** Punkte hinreichend.
- Es sind keine Hilfsmittel zugelassen.

Aufgabe	Mögliche Punkte					Erreichte Punkte				
	a	b	c	d	Σ	a	b	c	d	Σ
1	3	4	–	–	7			–	–	
2	2	1	2	2	7					
3	1	6	–	–	7			–	–	
4	1	3	1	–	5				–	
5	1	2	1	4	8					
6	1,5	2,5	3	1	8					
7	1	5	–	–	6			–	–	
8	3	2	–	–	5			–	–	
9	7×1				7					
Σ					60					

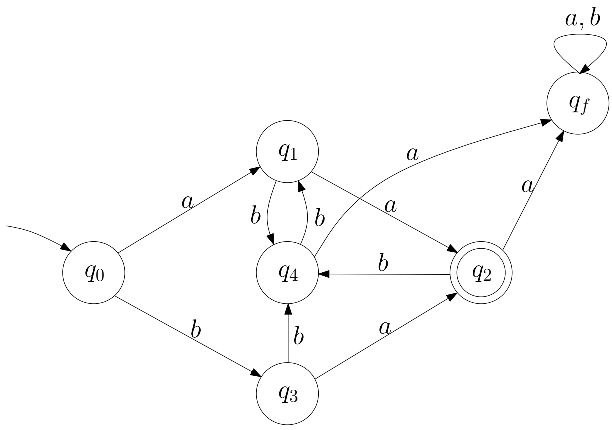
Problem 1: Endliche Automaten 3+4=7 Punkte

Gegeben sei der endliche Automat $\mathcal{A} = (Q = \{q_0, q_1, q_2, q_3, q_4, q_f\}, \Sigma = \{a, b\}, \delta, q_0, \{q_4\})$, wobei δ durch folgendes Zustandsdiagramm definiert ist.



(a) Geben Sie für jeden Zustand dessen ε-Abschluss an und geben Sie für $\mathcal{A}$ einen äquivalenten endlichen nicht-deterministischen Automaten $\mathcal{A}'$ an, der keinen ε-Übergang enthält und vollständig ist. Der Automat $\mathcal{A}'$ soll die gleiche Anzahl an Zuständen wie $\mathcal{A}$ besitzen.

(b) Gegeben sei der endliche Automat $\mathcal{A}_d = (Q_d = \{q_0, q_1, q_2, q_3, q_4, q_f\}, \Sigma = \{a, b\}, \delta_d, q_0, \{q_2\})$, wobei δ_d durch folgendes Zustandsdiagramm definiert ist.



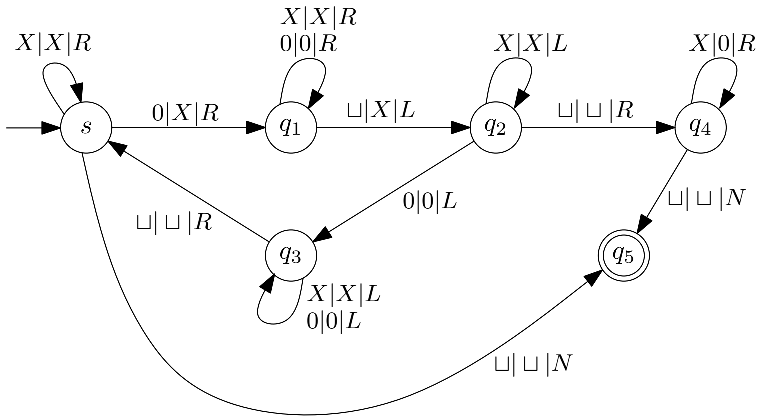
Beweisen Sie, dass $\mathcal{A}_d$ nicht minimal ist. Konstruieren Sie hierfür, mithilfe eines Verfahrens aus der Vorlesung, einen zu $\mathcal{A}_d$ äquivalenten minimalen Automaten. Geben Sie dabei alle Schritte an.

Problem 2: Turingmaschine 2+1+2+2=7 Punkte

Gegeben sei die folgende Turingmaschine

$$\mathcal{M} = (Q = \{s, q_1, q_2, q_3, q_4, q_5\}, \Sigma = \{0\}, \Gamma = \Sigma \cup \{\sqcup, X\}, \delta, s, F = \{q_5\}).$$

Die Übergangsfunktion δ von $\mathcal{M}$ ist durch untenstehendes Zustandsübergangsdiagramm dargestellt.



Für $\mathcal{M}$ gelten folgende Konventionen: $\mathcal{M}$ hält und akzeptiert sobald ein Endzustand erreicht wird. Alle Übergänge, die aus Endzuständen herausführen, werden demnach nicht verwendet. $\mathcal{M}$ hält und **akzeptiert nicht** für nicht dargestellte Zustandsübergänge.

(a) Dokumentieren Sie die Berechnung des Wortes 00 bis zu dem Punkt, an dem die Turingmaschine in den Zustand q_4 gelangt. Geben Sie dazu für jeden Schritt die aktuelle Konfiguration an.

(b) Für welche Eingaben wird der eingezeichnete Zustandsübergang von s nach q_5 benötigt?

(c) Geben Sie die Sprache $L(\mathcal{M}) \subseteq \Sigma^\star$ an, die $\mathcal{M}$ akzeptiert. Geben Sie außerdem die Funktion $f_\mathcal{M} \colon \Sigma^\star \to \Gamma^\star$ an, die $\mathcal{M}$ realisiert. Begründen Sie Ihre Antwort, indem Sie die Funktionsweise der Turingmaschine informell beschreiben.

(d) Zeigen Sie, dass für zwei entscheidbare Sprachen L_1 und L_2 folgende Aussage gilt. Die Sprache $L_1 \setminus L_2$ ist entscheidbar.

Problem 3: Reguläre und kontextfreie Sprachen 1+6=7 Punkte

(a) Gegeben ist die Sprache $L_1 = \{a^{h(x)} \in \{a\}^\star \mid x \in \mathbb{N} \setminus \{0\}\}$, wobei

$$h(x) = \begin{cases} 0, & \text{falls } x = 0, \\ 1, & \text{falls } x = 1, \\ h(x-2) + h(x-1), & \text{falls } x > 1. \end{cases}$$

Ist die Sprache L_1 regulär?

(b) Gegeben ist die Sprache $L_2 = \{a^n b^{m+n} c^n \in \{a, b, c\}^\star \mid m, n \in \mathbb{N}\}$.

(i) Geben Sie die formale Definition für das Pumping-Lemma für kontextfreie Sprachen an.

(ii) Beweisen oder widerlegen Sie, dass L_2 nicht kontextfrei ist.

Problem 4: Berechenbarkeit 1+3+1=5 Punkte

(a) In der Vorlesung wurde die Situation, in der sich die Turingmaschine während der Abarbeitung eines Wortes befindet, als *Konfiguration* bezeichnet. Geben Sie die formale Definition einer *Konfiguration* wieder und erläutern Sie diese.

(b) Eine *wiederholungssensitive* Turingmaschine $\mathcal{M} = (Q, \Sigma, \Gamma, \delta, s, F)$, kurz WSTM, hält auf einer Eingabe $w \in \Sigma^\star$ genau dann *nicht*, wenn eine Konfiguration von $\mathcal{M}$ während der Abarbeitung von w mehr als einmal auftritt.

Gegeben sei eine WSTM $\mathcal{M}$. Zeigen Sie, dass die Sprache $L_{\mathcal{M}} = \{w \in \Sigma^\star \mid \mathcal{M} \text{ akzeptiert } w\}$ entscheidbar ist.

(c) Sei L eine entscheidbare Sprache. Zeigen oder widerlegen Sie folgende Aussage: Jede Teilmenge von L ist ebenfalls entscheidbar.

Problem 5: NP-Vollständigkeit 1+2+1+4=8 Punkte

Das Entscheidungsproblem FeedbackArcSet ist wie folgt definiert:

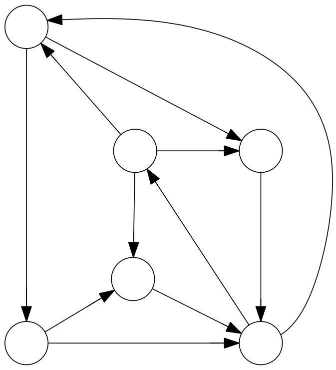
Problem FeedbackArcSet

Gegeben: Gerichteter Graph $G = (V, E)$ und Zahl $k \in \mathbb{N}$.

Frage: Existiert eine Kantenmenge $E' \subseteq E$ mit $|E'| \leq k$, sodass G ohne die Kanten aus E' azyklisch ist?

Hinweis: Ein Graph heißt *azyklisch*, wenn er keinen gerichteten Kreis enthält.

(a) Markieren Sie in dem unten dargestellten Graphen $G = (V, E)$ eine Kantenmenge E', die das Problem FeedbackArcSet für $k = 2$ auf G löst.

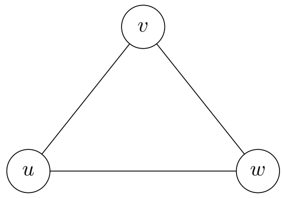


(b) Gegeben sei ein Problem Π. Erläutern Sie, wie man mithilfe eines NP-vollständigen Problems Π' zeigen kann, dass Π NP-schwer ist.

(c) Ein ungerichteter Graph $G = (V, E)$ kann wie folgt in einen gerichteten Graphen $G' = (V', E')$ transformiert werden:

> Für jeden Knoten $u \in V$ besitzt die Menge V' die Knoten u_{in} und u_{out}. Die Menge E' enthält für jeden Knoten $u \in V$ die Kante $(u_{\text{in}}, u_{\text{out}})$. Zudem enthält sie für jede Kante $\{u, v\} \in E$ die Kanten $(u_{\text{out}}, v_{\text{in}})$ und $(v_{\text{out}}, u_{\text{in}})$.

Konstruieren Sie gemäß der beschriebenen Transformation aus dem unten abgebildeten ungerichteten Graphen $G = (V, E)$ einen gerichteten Graphen $G' = (V', E')$.



(d) Gegeben sei das NP-vollständige Problem Knotenüberdeckung wie folgt:

Problem Knotenüberdeckung

Gegeben: Ungerichteter Graph $G = (V, E)$ und Zahl $k \in \mathbb{N}$.

Frage: Gibt es eine Knotenüberdeckung $V' \subseteq V$ mit $|V'| \leq k$?

Hinweis: $V' \subseteq V$ heißt *Knotenüberdeckung* von G, falls für jede Kante $\{u, v\} \in E$ gilt $u \in V'$ oder $v \in V'$.

Zeigen Sie, dass das Problem FeedbackArcSet NP-schwer ist.

Hinweis: Verwenden Sie die in Teilaufgabe (c) beschriebene Transformation.

Problem 6: Approximation – Maximum Matching 1,5 + 2,5 + 3 + 1 = 8 Punkte

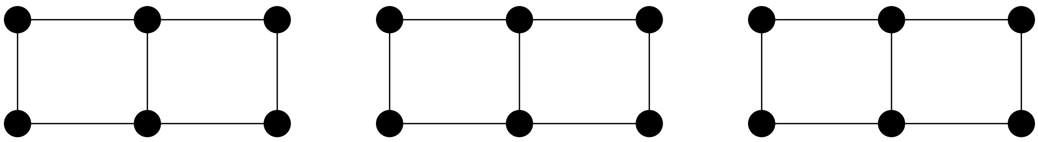
In einem einfachen, ungerichteten Graphen $G = (V, E)$ ist ein *Matching* eine Kantenmenge $M \subseteq E$, sodass keine zwei Kanten aus M zu einem gemeinsamen Knoten inzident sind. Ein Matching M heißt

- **inklusions***maximal*, wenn es keine echte Teilmenge eines anderen Matchings M' ist, d.h. es existiert kein Matching M', sodass $M \subsetneq M'$.
- **kardinalitäts***maximal*, wenn es kein Matching M' mit echt größerer Kantenanzahl gibt, d.h. es existiert kein Matching M', so dass $|M| < |M'|$.

(a) Zeichnen Sie in die untenstehenden Graphen Folgendes ein:

- in (i) ein **kardinalitäts**maximales Matching.
- in (ii) ein **inklusions**maximales Matching, das nicht **kardinalitäts**maximal ist.
- in (iii) ein nicht-leeres, **nicht inklusions**maximales Matching.

Zeichnen Sie hierzu die Kanten der Matchings fett ein.



(i) **kardinalitäts**maximal. (ii) **inklusions**maximal, nicht kardinalitätsmaximal. (iii) nicht leer, **nicht** inklusionsmaximal.

Eine Knotenmenge $V' \subseteq V$ heißt *Überdeckung* der Kantenmenge E, wenn für alle $e \in E$ gilt: $\exists u \in V'$ mit u inzident zu e.

(b) Sei M ein beliebiges Matching in $G = (V, E)$ und sei V' eine beliebige Überdeckung von E.

Zeigen Sie: $|M| \leq |V'|$.

(c) Sei $M_{\max}$ ein **inklusions**maximales Matching in $G = (V, E)$ und sei $V'_{\min}$ eine Überdeckung von E mit minimaler Kardinalität.

Zeigen Sie: $|V'_{\min}| \leq 2\,|M_{\max}|$.

(d) Das Problem MAXIMUM-MATCHING ist wie folgt definiert:

Problem MAXIMUM MATCHING
Gegeben: Einfacher, ungerichteter Graph $G = (V, E)$.
Gesucht: **Kardinalitäts**maximales Matching M_M.

Sei I eine Instanz des Matchingproblems, sei M_M ein **kardinalitäts**maximales Matching bzgl. I und sei $M_{\max}$ ein **inklusions**maximales Matching bzgl. I.

Zeigen Sie: $\mathrm{OPT}(I) \leq 2 \cdot \mathcal{A}(I)$, mit $\mathrm{OPT}(I) := |M_M|$, $\mathcal{A}(I) := |M_{\max}|$.

Hinweis: Man kann die Teilaufgaben (b) und (c) nutzen.

Problem 7: Grammatiken und Chomsky-Hierarchien 1+5=6 Punkte

Gegeben sei die Grammatik $G = \{\Sigma, V, S, R\}$ mit Terminalen $\Sigma = \{a, b, c\}$, Nichtterminalen $V = \{S, A, B, C, D\}$ und Produktionen

$$R = \left\{ \begin{array}{lll} S & \to & aAb, \\ A & \to & B \mid c, \\ B & \to & BCB \mid \varepsilon, \\ C & \to & aA \mid cc, \\ D & \to & SB \end{array} \right\}$$

(a) Enthält diese Grammatik G nutzlose Variablen? Falls dies zutrifft, entfernen Sie diese. Begründen Sie in beiden Fällen Ihre Entscheidung.

(b) Formen Sie die obige Grammatik G nach Entfernen nutzloser Variablen so um, dass diese in Chomsky-Normalform ist. Geben Sie die einzelnen Schritte aus der Vorlesung an und beschreiben Sie Ihr Vorgehen. Sie können dabei auf die topologische Sortierung im letzten Schritt verzichten.

Problem 8: Kellerautomaten 3+2=5 Punkte

Gegeben sei der Kellerautomat $P = (\{q_0, q_1, q_2, q_3\}, \{a, b, c\}, \{X, Y, Z_0\}, q_0, Z_0, \delta)$, der mit leerem Stack akzeptiert, wobei die Übergangsfunktion δ wie folgt definiert ist:

$$
\begin{array}{lllll}
1. & \delta(q_0, a, \varepsilon) & = & (q_0, X), \\
2. & \delta(q_0, b, \varepsilon) & = & (q_1, Y), \\
3. & \delta(q_0, c, \varepsilon) & = & (q_2, \varepsilon), \\
4. & \delta(q_1, b, \varepsilon) & = & (q_1, Y), \\
5. & \delta(q_1, c, \varepsilon) & = & (q_2, \varepsilon), \\
6. & \delta(q_2, b, Y) & = & (q_2, \varepsilon), \\
7. & \delta(q_2, a, X) & = & (q_3, \varepsilon), \\
8. & \delta(q_3, a, X) & = & (q_3, \varepsilon), \\
9. & \delta(q_3, \varepsilon, Z_0) & = & (q_3, \varepsilon).
\end{array}
$$

(a) Geben Sie die von dem Kellerautomaten P akzeptierte Sprache L an. Zudem wandeln Sie P in eine Grammatik G um, sodass $L(G) = L$ gilt.

(b) Geben Sie an, ob der Kellerautomat P deterministisch ist. Begründen Sie Ihre Antwort.

Problem 9: Verschiedenes 7 Punkte

Jeder der folgenden Aussagenblöcke umfasst mehrere Einzelaussagen. Die sich unterscheidenden Aussagenbausteine sind durch Kästchen gekennzeichnet. Kreuzen Sie genau jene Bausteine an, die in einer wahren Einzelaussage enthalten sind. Jeder Aussagenblock enthält mindestens eine wahre Einzelaussage. Unvollständig oder falsch angekreuzte Aussagenblöcke werden mit null Punkten bewertet. Sie erhalten einen Punkt für jeden Aussagenblock, für den Sie genau die richtige Menge an Aussagenbausteinen angekreuzt haben.

Sei $G = (\Sigma, V, S, R)$ eine Grammatik.

- ☐ Falls G kontextfrei ist, dann haben alle Ableitungsregeln in R die Form $A \to v$ mit $A \in V$ und $v \in (V \cup \Sigma)^\star$.
- ☐ Falls G kontextsensitiv ist, darf jedes Nichtterminal auf das leere Wort abgeleitet werden.
- ☐ Falls G eine rekursiv-aufzählbare Grammatik ist, dann gilt für jede Ableitungsregel, dass die linke Seite weniger Nichterminale als die rechte Seite besitzt.
- ☐ Falls G eine rechtslineare Grammatik ist, dann besteht die rechte Seite jeder Ableitungsregel aus einem Terminal gefolgt von einem Nichtterminal.

Der Cocke-Younger-Kasami Algorithmus

- ☐ überprüft, ob für eine gegebene Chomsky-2-Grammatik G in Chomsky-Normalform ein Wort w in $L(G)$ liegt.
- ☐ überprüft, ob für eine gegebene Chomsky-1-Grammatik G ein Wort w in $L(G)$ liegt.
- ☐ besitzt eine polynomielle Laufzeit.

Sei L eine Sprache über dem Alphabet Σ und sei $L^c = \Sigma^\star \setminus L$ das Komplement zu L, sodass L und L^c semientscheibar sind.

- ☐ Weder L noch L^c ist entscheidbar.
- ☐ Entweder L oder L^c ist entscheidbar.
- ☐ L und L^c sind entscheidbar.
- ☐ Die Sprache $L \cup L^c$ ist entscheidbar.

Kreuzen Sie an, welche der folgenden Aussagen wahr ist.

- ☐ Die Klasse der von Chomsky-1-Grammatiken erzeugten Sprachen stimmt mit der Klasse NTAPE(n) überein, wobei n die Eingabelänge ist.
- ☐ Für jede Sprache, die durch einen nichtdeterministischen Kellerautomaten akzeptiert wird, gibt es einen deterministischen Kellerautomaten, der dieselbe Sprache akzeptiert.
- ☐ Zu jedem nichtdeterministischen endlichen Automaten gibt es einen äquivalenten deterministischen endlichen Automaten.
- ☐ Jede Turingmaschine mit mehreren Bändern kann von einer Turingmaschine mit einem Band simuliert werden.

Die maximale Entropie für 42 Buchstaben ist auf zwei Nachkommastellen gerundet (unter Verwendung des Logarithmus zur Basis 2)

- ☐ $4,43$
- ☐ $7,81$
- ☐ $5,39$
- ☐ $3,14$

Kreuzen Sie an, welche der folgenden Aussagen wahr ist.

- ☐ Für einen Präfix-Code gilt, dass kein Codewort Anfang eines anderen Codeworts ist.
- ☐ Die Huffman-Codierung setzt nicht voraus, dass die Wahrscheinlichkeiten der Zeichen im Voraus bekannt sind.
- ☐ Die Datenkompression erhöht die Redundanz und reduziert damit die Fehleranfälligkeit.
- ☐ Der Huffman-Algorithmus berechnet einen Codierungsbaum mit minimaler mittlerer Codewortlänge.

Kreuzen Sie an, welche der folgenden Aussagen wahr ist.

- ☐ Ein Block-Code betrachtet Codewörter fester Länge, wobei aufeinanderfolgende Blöcke unabhängig voneinander kodiert werden.
- ☐ Die mittlere Codewortlänge der Shannon-Fano Codierung ist optimal.
- ☐ In einem Faltungs-Code können Codewörter beliebige Länge besitzen, wobei die Zeichen vom Vorgeschehen unabhängig sind.
- ☐ Jeder Paritätscode erkennt Einzelfehler.