Abschlussklausur Theoretische Informatik I 2016/2017

16. Februar 2017

---

1. **Bitte ausfüllen:**

   Vorname:

   Nachname:

   Matrikelnummer:

   Unterschrift:

2. Achten Sie darauf, dass Ihre Klausur vollständig ist und getackert bleibt (14 Blätter)!
3. Benutzen Sie **nur das an dieses Blatt angeheftete Papier**. Bei Bedarf können wir weitere Leerblätter austeilen. Wenn der Platz auf der Vorderseite des jeweiligen Aufgabenblatts nicht ausreicht, machen Sie kenntlich, wo Sie die Bearbeitung der Aufgabe fortsetzen.
4. Als Hilfsmittel sind **ausschließlich** Sprachwörterbücher und ein zweiseitig **handgeschriebenes DIN A4-Blatt** erlaubt. Elektronische Geräte müssen während der Klausur ausgeschaltet bleiben. Täuschungsversuche werden mit 0 Punkten gewertet und dem Prüfungsamt gemeldet.
5. Schreiben Sie leserlich und bearbeiten Sie Ihre Klausur **nicht mit Bleistift** und auch **nicht in roter oder grüner Farbe!**
6. Wir werden das Deckblatt während der Klausur auf korrekte Daten überprüfen. Legen Sie dazu Ihren **Studentenausweis** und einen **amtlichen Lichtbildausweis** bereit.

Die Bearbeitungszeit beträgt 120 Minuten.

Mit 40 Punkten ist die Klausur sicher bestanden.

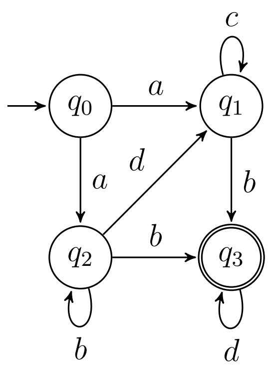
Punkteverteilung: (wird von den Korrektoren ausgefüllt)

Aufgabe	1	2	3	4	5	6	7	8	9	10	Σ
Max.	10	10	10	10	10	10	10	10	10	10	**100**
Punkte											

1. Ardens Lemma

10 Punkte

Gegeben sei der folgende NFA A über dem Alphabet $\Sigma = \{a, b, c, d\}$:



Geben Sie das zu A gehörige Gleichungssystem an und lösen Sie dieses unter Verwendung von Ardens Lemma.

2. CYK-Algorithmus

$\boxed{\text{10 Punkte}}$

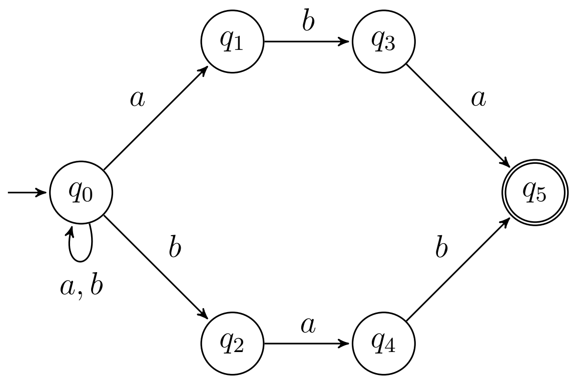
Entscheiden Sie mit Hilfe des Cocke–Younger–Kasami-Algorithmus, ob das Wort *abbaa* von folgender Grammatik erzeugt wird:

$$\begin{aligned} S &\to AB \mid SC, \\ A &\to CB \mid a, \\ B &\to BC \mid b, \\ C &\to AA \mid a. \end{aligned}$$

3. Determinisierung

10 Punkte

Determinisieren Sie folgenden NFA über $\Sigma = \{a, b\}$ unter Verwendung der Potenzmengen-Konstruktion:



4. Äquivalenzklassen

$\boxed{7 + 3 = 10 \text{ Punkte}}$

Gegeben sei die Sprache:

$$L = \{w \in \{a, b\}^* \mid w \text{ enthält } a.a \text{ oder } b.b\}.$$

a) Geben Sie alle Äquivalenzklassen an.

b) Konstruieren Sie den Äquivalenzklassen-Automaten A_L.

5. Pumping-Lemma

10 Punkte

Zeigen Sie, dass die Sprache

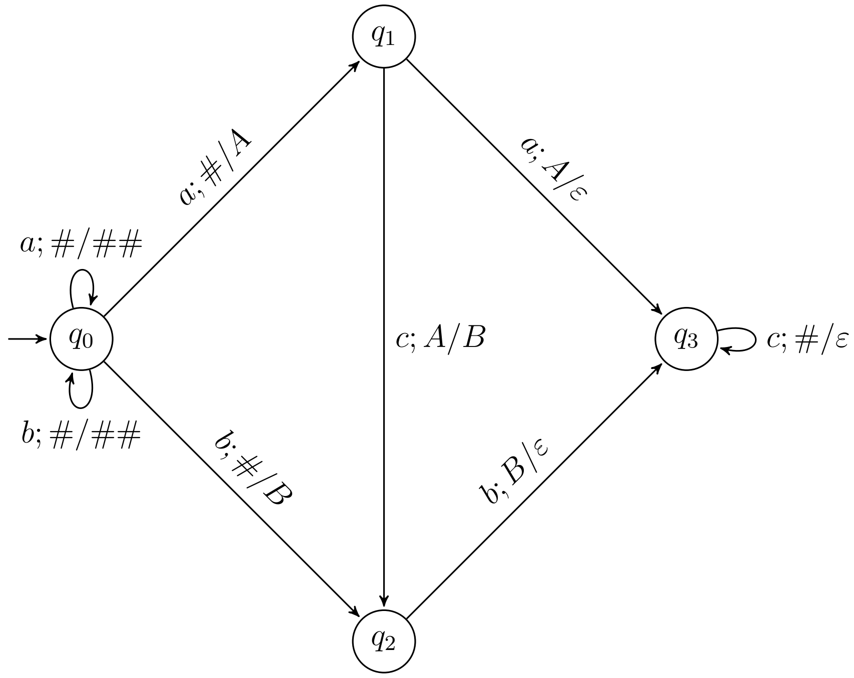
$$L = \left\{ a^i b^j c^k d^l \;\middle|\; i, j, k, l \in \mathbb{N} \text{ und } i = k, j = l \right\}$$

nicht kontextfrei ist.

6. Tripelkonstruktion

10 Punkte

Gegeben sei der folgende PDA P über dem Alphabet $\Sigma = \{a, b, c\}$ mit initialem Stack-Symbol $\#$. Ferner akzeptiert P mit leeren Stack.



Benutzen Sie das Verfahren aus der Vorlesung um eine kontextfreie Grammatik G mit $L(G) = L(P)$ zu konstruieren.

7. Regularität

10 Punkte

Es sei $K \in \mathbb{N}$ mit $K \geq 2$ eine Konstante und R eine reguläre Sprache über dem Alphabet $\Sigma = \{a, b, \#\}$. Betrachten Sie nun die Sprache

$$L = \left\{ a^n.\#.w \;\middle|\; n \in \mathbb{N} \text{ und } (w \in R, \text{ falls } K \text{ teilt } n) \text{ oder } (w \in \bar{R}, \text{ falls } K \text{ teilt } n \text{ nicht}) \right\}.$$

Zeigen Sie, dass L regulär ist, indem Sie einen Automaten (schematisch) angeben.

8. Fragen zu Sprachen

$\boxed{2 + 2 + 3 + 3 = 10 \text{ Punkte}}$

Beantworten Sie die folgenden Fragen. Begründen Sie ihre Antwort mit einem kurzen Beweis oder einem Gegenbeispiel.

a) Es seien L_1, L_2 reguläre Sprachen und L eine Sprache mit: $L_1 \subseteq L \subseteq L_2$. Ist L regulär?

b) Es seien L_1, L_2 zwei nicht reguläre Sprachen. Ist dann auch $L_1 \cup L_2$ nicht regulär?

c) Ist jede endliche Sprache regulär?

d) Ist die unendliche Vereinigung von kontextfreien Sprachen kontextfrei?

9. Pushdown-Automaten

$\boxed{3 + 7 = 10 \text{ Punkte}}$

Es sei P ein PDA über dem Alphabet Σ mit Stack-Alphabet Γ und R eine reguläre Sprache über Γ. Wir führen eine neue Akzeptanzbedingung für P ein: P akzeptiert, falls der aktuelle Stack-Inhalt w, gelesen als Wort von oben nach unten, in der Sprache R liegt. Falls also gilt: $w \in R$. Wir nennen diese Akzeptanzbedingung **reguläre Akzeptanz**.

a) Zeigen Sie, dass es zu jedem PDA mit leerer-Stack-Akzeptanz einen PDA mit regulärer Akzeptanz gibt, der die gleiche Sprache akzeptiert.

b) Zeigen Sie, dass es zu jedem PDA mit regulärer Akzeptanz einen PDA mit leerer-Stack-Akzeptanz gibt, der die gleiche Sprache akzeptiert.

10. Upward Closure

3+3+4 = 10 Punkte

Es sei Σ ein endliches Alphabet. Wir definieren auf Σ^* eine Ordnungsrelation $(\leq)$. Für v und w aus Σ^* gilt:

$v \leq w$ genau dann, wenn v aus w hervorgeht, indem man Buchstaben aus w streicht.

Alternativ kann man sich vorstellen, dass in v Buchstaben hinzugefügt werden.
Zum Beispiel ist $aba \leq \underline{ac}\,a\,\underline{abcb}\,b\,\underline{acc}\,a\,\underline{abca}$. Hier sind die Buchstaben, die entfernt werden, unterstrichen.

Für eine Sprache $L \subseteq \Sigma^*$ definieren wir den **upward closure** als die Sprache

$$L{\uparrow} = \{w \in \Sigma^* \mid \text{ Es gibt ein } v \in L, \text{ sodass } v \leq w\}.$$

a) Die Sprache L sei für diesen Aufgabenteil kontextfrei. Zeigen Sie, dass $L{\uparrow}$ auch kontextfrei ist.

b) Sei L nun wieder eine beliebige Sprache. Wir nennen ein Wort $w \in L$ **minimal**, falls für alle $v \in L$ gilt, aus $v \leq w$ folgt schon $v = w$. Es gibt also kein $v \in L$, dass echt kleiner ist als w. Die Menge der minimalen Elemente bezeichnen wir mit $\mathrm{Min}(L)$. Zeigen Sie, dass

$$L{\uparrow} = \bigcup_{w \in \mathrm{Min}(L)} \{w\}{\uparrow}\,.$$

c) Nehmen Sie an, dass $N \in \mathbb{N}$ eine Schranke für die Länge der minimalen Elemente ist. Beweisen Sie, dass $L{\uparrow}$ regulär ist.