Prof. Dr. J. Giesl

M. Brockschmidt, F. Emmes, C. Fuhs, C. Otto, T. Ströder

Vorname: ______________________________

Nachname: ______________________________

Matrikelnummer: ______________________________

Studiengang (bitte ankreuzen):

○ Informatik Bachelor
○ Informatik Master (Auflage)
○ Mathematik Bachelor
○ Technik-Kommunikation M.A.
○ Informatik Lehramt
○ Informatik Promotion (Auflage)
○ Technik-Kommunikation Bachelor
○ Sonstige: ______________________

	Anzahl Punkte	**Erreichte Punkte**
Aufgabe 1	**4**	
Aufgabe 2	**5**	
Aufgabe 3	**4**	
Aufgabe 4	**4**	
Aufgabe 5	**4**	
Aufgabe 6	**10**	
Aufgabe 7	**8**	
Aufgabe 8	**6**	
Summe	**45**	

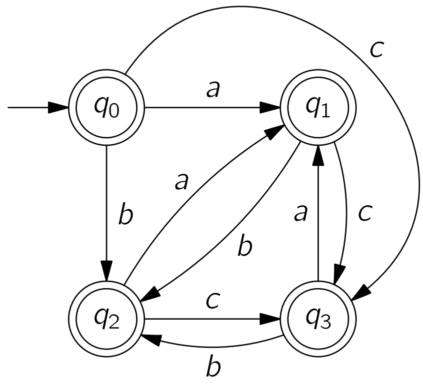
Hinweise:

- Geben Sie Ihre Antworten in lesbarer und verständlicher Form an.
- Schreiben Sie mit dokumentenechten Stiften, nicht mit roten Stiften oder mit Bleistiften.
- Bitte beantworten Sie die Aufgaben auf den Aufgabenblättern (benutzen Sie auch die Rückseiten).
- **Auf alle Blätter** (inklusive zusätzliche Blätter) müssen Sie **Ihren Namen und Ihre Matrikelnummer** schreiben.
- Was nicht bewertet werden soll, streichen Sie bitte durch.
- Werden **Täuschungsversuche** beobachtet, so wird die Präsenzübung mit **0 Punkten** bewertet.
- Geben Sie am Ende der Übung **alle Blätter zusammen mit den Aufgabenblättern ab**.

Matrikelnummer: **Name:**

Aufgabe 1 (Universalität): **(3 + 1 = 4 Punkte)**

Sei $\Sigma = \{a, b, c\}$ ein Alphabet. Betrachten Sie den folgenden NFA M.



a) Überführen Sie den NFA M in einen DFA M' mit $L(M) = L(M')$.

b) Gilt $L(M) = \Sigma^*$? Geben Sie eine kurze Begründung für Ihre Aussage.

Matrikelnummer: **Name:**

Aufgabe 2 (Induktionsbeweis): (5 Punkte)

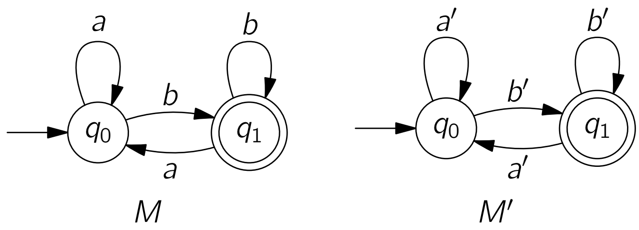
Seien $\Sigma = \{0, 1\}$ und $\Sigma' = \{0', 1'\}$ Alphabete. Wir definieren die folgende Funktion *prime* : $\Sigma^* \to \Sigma'^*$ mit

$$\begin{aligned} \mathit{prime}(\epsilon) &= \epsilon \\ \mathit{prime}(a \cdot w) &= a' \cdot \mathit{prime}(w) \text{ für } w \in \Sigma^*, a \in \Sigma \end{aligned}$$

Es gilt also zum Beispiel $\mathit{prime}(011) = 0'1'1'$.
Betrachten wir nun einen DFA $M = (Q, \Sigma, \delta, q_0, F)$.
Wir definieren dazu den Prime-Automaten $M' = (Q, \Sigma', \delta', q_0, F)$ mit

$$\delta'(q, a') = \delta(q, a) \text{ für alle } q \in Q, a' \in \Sigma'.$$

Diese Konstruktion ist untenstehend für einen DFA beispielhaft illustriert.



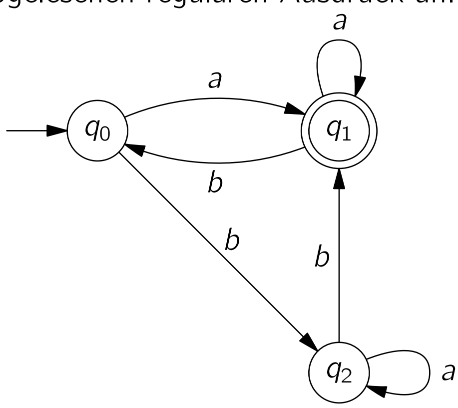
Beweisen Sie nun folgende Aussage für alle $w \in \Sigma^*$ per Induktion über die Wortlänge $|w|$:

$$w \in L(M) \text{ genau dann, wenn } \mathit{prime}(w) \in L(M')$$

Matrikelnummer: **Name:**

Aufgabe 3 (DFAs und reguläre Ausdrücke): (4 Punkte)

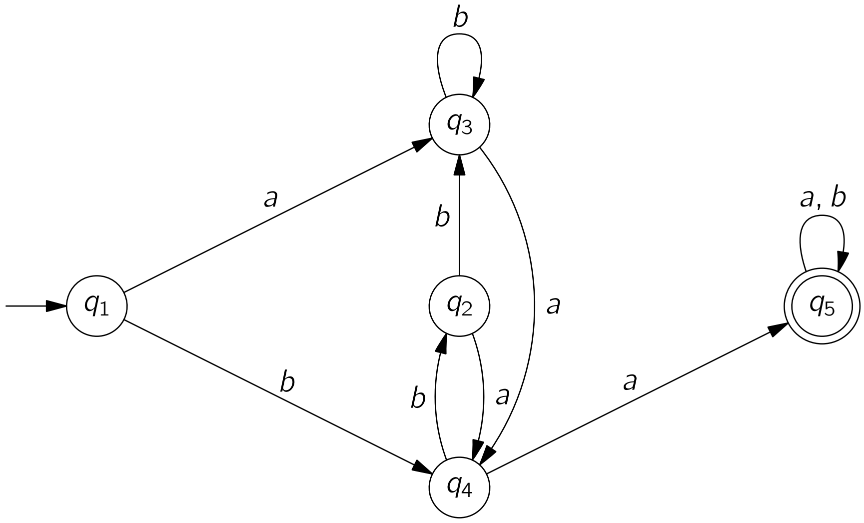
Sei $\Sigma = \{a, b\}$ ein Alphabet. Wandeln Sie folgenden DFA über Σ in einen regulären Ausdruck um, indem Sie Zustände schrittweise entfernen und die betroffenen Kanten durch reguläre Ausdrücke ersetzen. Geben Sie den resultierenden Automaten nach Entfernung von q_2 an und geben Sie den zum Schluss abgelesenen regulären Ausdruck an.



Matrikelnummer: **Name:**

Aufgabe 4 (DFA minimieren): (4 Punkte)

Verwenden Sie den **schnellen** Markierungsalgorithmus aus der Vorlesung, um folgenden DFA zu minimieren. Geben Sie die beim Anwenden des Algorithmus entstehende Tabelle und den minimalen Automaten an.



Aufgabe 5 (Thompson-Konstruktion): (4 Punkte)

Geben Sie einen ϵ-NFA an, der die Sprache $(aba)^* + b$ akzeptiert.

Matrikelnummer: **Name:**

Aufgabe 6 (Pumping Lemma): **(2 + 8 = 10 Punkte)**

a) Geben Sie das Pumping Lemma für reguläre Sprachen an.

b) Führen Sie für alle $i \in \{1, 2, 3\}$ die folgende Aufgabe für die Sprache L_i über dem Alphabet Σ_i durch:

- Falls L_i regulär ist, dann geben Sie einen regulären Ausdruck r_i mit $L(r_i) = L_i$ an.
- Falls L_i nicht regulär ist, dann beweisen Sie dies mit Hilfe des Pumping Lemmas.

Wir erinnern an die Funktion $rev : \Sigma^* \to \Sigma^*$ aus den Übungen:

$$\begin{aligned} rev(\epsilon) &:= \epsilon \\ rev(w \cdot a) &:= a \cdot rev(w) \qquad\qquad a \in \Sigma, w \in \Sigma^* \end{aligned}$$

- $L_1 := \{w \cdot rev(w) \mid w \in \Sigma_1^*\}$ über $\Sigma_1 := \{a\}$.
- $L_2 := \{0^k(10)^k \mid k \in \mathbb{N}_0\}$ über $\Sigma_2 := \{0, 1\}$.
- $L_3 := \{(0^k1^{k+2})^\ell \mid k, \ell \in \mathbb{N}_0, k < 2\}$ über $\Sigma_3 := \{0, 1\}$.

Aufgabe 7 (Myhill-Nerode-Äquivalenzklassen): **(5 + 3 = 8 Punkte)**

Führen Sie für alle $i \in \{1, 2\}$ die folgende Aufgabe für die Sprache L_i über dem Alphabet $\Sigma := \{0, 1\}$ durch:

- Falls L_i regulär ist, geben Sie den minimalen Myhill-Nerode-Automaten an.
- Falls L_i nicht regulär ist, dann begründen Sie, warum der Index von $\equiv_{L_i}$ unendlich ist. Zeigen Sie hierzu, wie man unendlich viele Wörter $w_0, w_1, \ldots$ erhalten kann, so dass $w_k \not\equiv_{L_i} w_\ell$ für alle $k \neq \ell$ ist.

Wir führen zunächst die Funktionen $\sharp_0, \sharp_1 : \Sigma^* \to \mathbb{N}_0$ ein, die die Vorkommen von 0 bzw. 1 in einem Wort w zählen. Formal ist $\sharp_0$ wie folgt definiert:

$$\begin{aligned} \sharp_0(\epsilon) &:= 0 & \\ \sharp_0(w \cdot 0) &:= \sharp_0(w) + 1 & w \in \Sigma^* \\ \sharp_0(w \cdot 1) &:= \sharp_0(w) & w \in \Sigma^* \end{aligned}$$

$\sharp_1$ ist analog definiert.

a) $L_1 := \{w \in \Sigma^* \mid \sharp_0(w) = \sharp_1(w)\}$

b) $L_2 := \{w \in \Sigma^* \mid \sharp_0(w) \text{ ist durch 3 teilbar}\}$

Matrikelnummer: **Name:**

Aufgabe 8 (Reguläre und Nicht-Reguläre Sprachen): (1,5 + 1,5 + 1,5 + 1,5 = 6 Punkte)

Sei $\Sigma = \{a, b\}$ ein Alphabet. Beantworten Sie folgende Fragen mit einer kurzen Begründung.

a) Gibt es eine reguläre Sprache, die Teilmenge einer nicht-regulären Sprache ist?

b) Gibt es eine nicht-reguläre Sprache, die Teilmenge einer regulären Sprache ist?

c) Gibt es eine reguläre Sprache L_1 und eine nicht-reguläre Sprache L_2 über Σ mit $L_1 \cap L_2 = \emptyset$?

d) Gibt es eine reguläre Sprache L_1 und eine nicht-reguläre Sprache L_2 über Σ mit $L_1 \cup L_2 = \Sigma^*$?