Prof. Dr. J. Giesl

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

Hinweise:

- Die **Hausaufgaben** sollen in Gruppen von je 2 Studierenden aus dem gleichen Tutorium bearbeitet werden.
- Die Lösungen der Hausaufgaben müssen bis Mi., 12.05.2010 im Tutorium abgegeben werden. Alternativ ist es bis 17 Uhr möglich, diese in den Kasten im Flur des LuFG I2 einzuwerfen (Ahornstr. 55, E1, 2. Etage).
- Namen und Matrikelnummern der Studierenden sowie **die Nummer der Übungsgruppe** sind auf jedes Blatt der Abgabe zu schreiben. **Heften bzw. tackern Sie die Blätter**!
- Die **Tutoraufgaben** werden in den jeweiligen Tutorien gemeinsam besprochen und bearbeitet.

Tutoraufgabe 1 (Induktionsbeweis):

Sei Σ ein Alphabet. Wir wollen die "Verdoppelungsfunktion" $dup : \Sigma^* \to \Sigma^*$ betrachten, die alle Buchstaben in Wörtern verdoppelt, also z.B. das Wort *ab* auf *aabb* und 123 auf 112233 abbildet.

a) Definieren sie die Funktion *dup* formal auf induktive (rekursive) Weise.

b) Sei M ein DFA. Wie kann man nun **automatisch** (d.h. mit einem allgemeinen Verfahren, das unabhängig vom Automaten M ist) einen DFA M^2 konstruieren, der die Sprache $dup(L(M)) := \{dup(w) \mid w \in L(M)\}$ erkennt?

c) Beweisen Sie, dass $L(M^2) = dup(L(M))$ gilt.

Lösung:

a)

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

b) Die Idee ist hier, jede Transition des Ursprungsautomaten aufzutrennen und einen "Zwischenzustand" einzusetzen, so dass das Zeichen der ursprünglichen Transition zwei mal gelesen wird. Damit der resultierende Automat ein DFA bleibt, müssen für diese neuen Zustände alle Symbole, mit denen die "zerschnittene" Transition nicht beschriftet war, zu einem "Fehlerzustand" führen, von dem aus nie wieder ein Endzustand erreicht werden kann.

Sei $M = (Q, \Sigma, \delta, q_0, F)$. Dann definieren wir $M^2 := (Q^2, \Sigma, \delta^2, q_0, F)$ mit $Q^2 := \{e\} \cup (\bigcup_{q \in Q}(\{q^a \mid a \in \Sigma\} \cup \{q\}))$, wobei wir o.B.d.A. verlangen, dass $q^a \notin Q$ gilt (entsprechendes kann man durch ein Umbenennen der Zustände in Q erreichen). Hier ist e der angesprochene neue "Fehlerzustand".

Zuletzt definieren wir

$$\delta^2(q, a) = \begin{cases} q^a & q \in Q \\ \delta(\tilde{q}, a) & q = \tilde{q}^a \in Q^2, \tilde{q} \in Q \\ e & \text{sonst} \end{cases}$$

c) Wir müssen nun $L(M^2) = dup(L(M))$ zeigen. Wir beweisen dies, indem wir die Aussage (†) $\hat{\delta}(q, w) = q' \Leftrightarrow \hat{\delta}^2(q, dup(w)) = q'$ für alle $q \in Q$ (also nicht die neuen Zustände $Q^2 \setminus Q$) per Induktion über die Wortlänge $|w|$ zeigen.

Zum Induktionsanfang ist $w = \epsilon$ und es gilt $\hat{\delta}(q, \epsilon) = q$ und $\hat{\delta}^2(q, dup(\epsilon)) = \hat{\delta}^2(q, \epsilon) = q$.

Im Induktionsschritt betrachten wir $w = w'a$ für ein $a \in \Sigma$ und setzen voraus, dass die Aussage für w' bereits gilt. Da M ein DFA ist, gibt es genau ein q' mit $\hat{\delta}(q, w' \cdot a) = q'$ und ein $\tilde{q}$ mit $\hat{\delta}(q, w') = \tilde{q}$, so dass $\delta(\tilde{q}, a) = q'$ gilt.

Nach unser Induktionshypothese gilt dies für w' genau dann, wenn auch $\hat{\delta}^2(q, dup(w')) = \tilde{q}$ gilt. Nach Konstruktion gilt aber auch $\delta^2(\tilde{q}, a) = \tilde{q}^a$ und $\delta^2(\tilde{q}^a, a) = q'$. Damit gilt dann auch die Aussage $\hat{\delta}(q, w) = \hat{\delta}(q, w' \cdot a) = q' \Leftrightarrow \hat{\delta}^2(q, dup(w)) = \hat{\delta}^2(q, dup(w' \cdot a)) = \hat{\delta}^2(q, dup(w') \cdot a \cdot a) = q'$.

Nun gilt $w \in L(M) \Leftrightarrow \hat{\delta}(q_0, w) = f \in F$ und nach (†) ist dies äquivalent zu $\hat{\delta}^2(q_0, dup(w)) = f \in F \Leftrightarrow dup(w) \in L(M^2)$. Damit ist die Aussage bewiesen.

Hausaufgabe 2 (Induktionsbeweis): (1 + 3 + 4 = 8 Punkte)

Sei Σ ein Alphabet. Wir wollen die "Spiegelfunktion" $rev : \Sigma^* \to \Sigma^*$ betrachten, die Wörter auf ihr Spiegelbild abbildet, also z.B. das Wort *ab* auf *ba* und 123 auf 321.

a) Definieren sie die Funktion *rev* formal auf induktive (rekursive) Weise.

b) Sei M ein DFA. Wie kann man nun **automatisch** (d.h. mit einem allgemeinen Verfahren, das unabhängig vom Automaten M ist) einen ϵ-NFA M_{rev} konstruieren, der die Sprache $rev(L(M)) := \{rev(w) \mid w \in L(M)\}$ erkennt?

c) Beweisen Sie, dass $L(M_{rev}) = rev(L(M))$ gilt.

Hinweis: Beweisen sie eine **allgemeine** Aussage über den Zusammenhang zwischen der Zustands-Übergangsfunktion $\hat{\delta}$ des Automaten M und der Zustands-Übergangsfunktion $\hat{\delta}_{rev}$ des Automaten M_{rev}. Verwenden Sie dazu wie in der Vorlesung eine Induktion über die Länge der betrachteten Wörter.

Lösung:

a)

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

b) Die Idee ist hier, die Kanten im Automaten "umzudrehen", also in M_{rev} eine a-Transition von q nach q' zu erlauben, wenn es in M eine a-Transition von q' nach q gab. Der Startzustand in M wird dann zum einzigen Endzustand von M_{rev}. Als Startzustände in M_{rev} wollen wir eigentlich die Endzustände von M zulassen. Da aber auch ϵ-NFAs einen eindeutigen Startzustand haben müssen, führen wir einen neuen, eindeutigen Startzustand ein, der per ϵ-Transition mit den Endzuständen von M verbunden wird.

Sei $M = (Q, \Sigma, \delta, q_0, F)$. Mit einem neuen Zustand q_s (d.h. $q_s \notin Q$) ist $M_{rev} := (Q \dot\cup \{q_s\}, \Sigma, \delta_{rev}, q_s, \{q_0\})$. Dabei ist δ_{rev} wie folgt definiert:

$$
\delta_{rev}(q, a) = \begin{cases} \{q' \in Q \mid q = \delta(q', a)\} & q \neq q_s \\ F & q = q_s \wedge a = \epsilon \\ \emptyset & q = q_s \wedge a \neq \epsilon \end{cases}
$$

c) Wir müssen nun beweisen, dass $L(M_{rev}) = rev(L(M))$ gilt. Wir zeigen dafür die allgemeinere Aussage $\hat{\delta}(q, w) = q' \Leftrightarrow q \in \hat{\delta}_{rev}(q', rev(w))$ für alle $q, q' \in Q$ (d.h. für alle Zustände außer dem neu eingeführten Startzustand q_s) per Induktion über die Wortlänge $|w|$.

Im Induktionsanfang ist die Länge 0, also $w = \epsilon$. Dann gilt $\hat{\delta}(q, \epsilon) = q$ und $\hat{\delta}_{rev}(q, \epsilon) = \{q\}$.

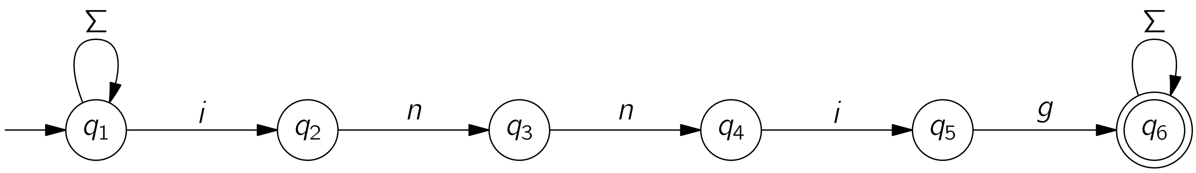
Im Induktionsschluss setzen wir für $w = w' \cdot a$, $a \in \Sigma$ voraus, dass die Aussage für w' gilt. Sei nun $\hat{\delta}(q, w) = q'$ mit $\hat{\delta}(q, w') = \tilde{q}$ und $\delta(\tilde{q}, a) = q'$. Nach Konstruktion von δ_{rev} gilt dann auch $\tilde{q} \in \delta_{rev}(q', a)$.

Nach unserer Induktionshypothese gilt außerdem $\hat{\delta}(q, w') = \tilde{q} \Leftrightarrow q \in \hat{\delta}_{rev}(\tilde{q}, rev(w'))$. Zusammen mit $\tilde{q} \in \delta_{rev}(q', a)$ folgt damit direkt auch $q \in \hat{\delta}_{rev}(q', a \cdot rev(w')) = \hat{\delta}_{rev}(q', rev(w' \cdot a)) = \hat{\delta}_{rev}(q', rev(w)) \Leftrightarrow \hat{\delta}(q, w) = q'$.

Damit gilt $w \in L(M) \Leftrightarrow \hat{\delta}(q_0, w) = f \in F \Leftrightarrow q_0 \in \hat{\delta}_{rev}(f, rev(w))$. Da nach Konstruktion $\delta_{rev}(q_s, \epsilon) = \epsilon{-}Huelle(q_s) = F$, gilt dies genau dann wenn $q_0 \in \hat{\delta}_{rev}(q_s, rev(w))$ und damit gilt $rev(w) \in L(M_{rev}) \Leftrightarrow w \in L(M)$.

Tutoraufgabe 3 (Potenzmengenkonstruktion):

Sei $\Sigma := \{a, b, \ldots, z\}$ ein Alphabet. Der folgende NFA akzeptiert alle Wörter, die mindestens einmal das Wort *innig* enthalten, also die Sprache $\Sigma^* innig \Sigma^*$. Um die Darstellung des Automaten zu vereinfachen, nutzen wir die verkürzende Schreibweise, die in Tutoraufgabe 4 auf dem zweiten Übungsblatt eingeführt wurde, und erlauben es, Transitionen mit Mengen von Symbolen zu beschriften.

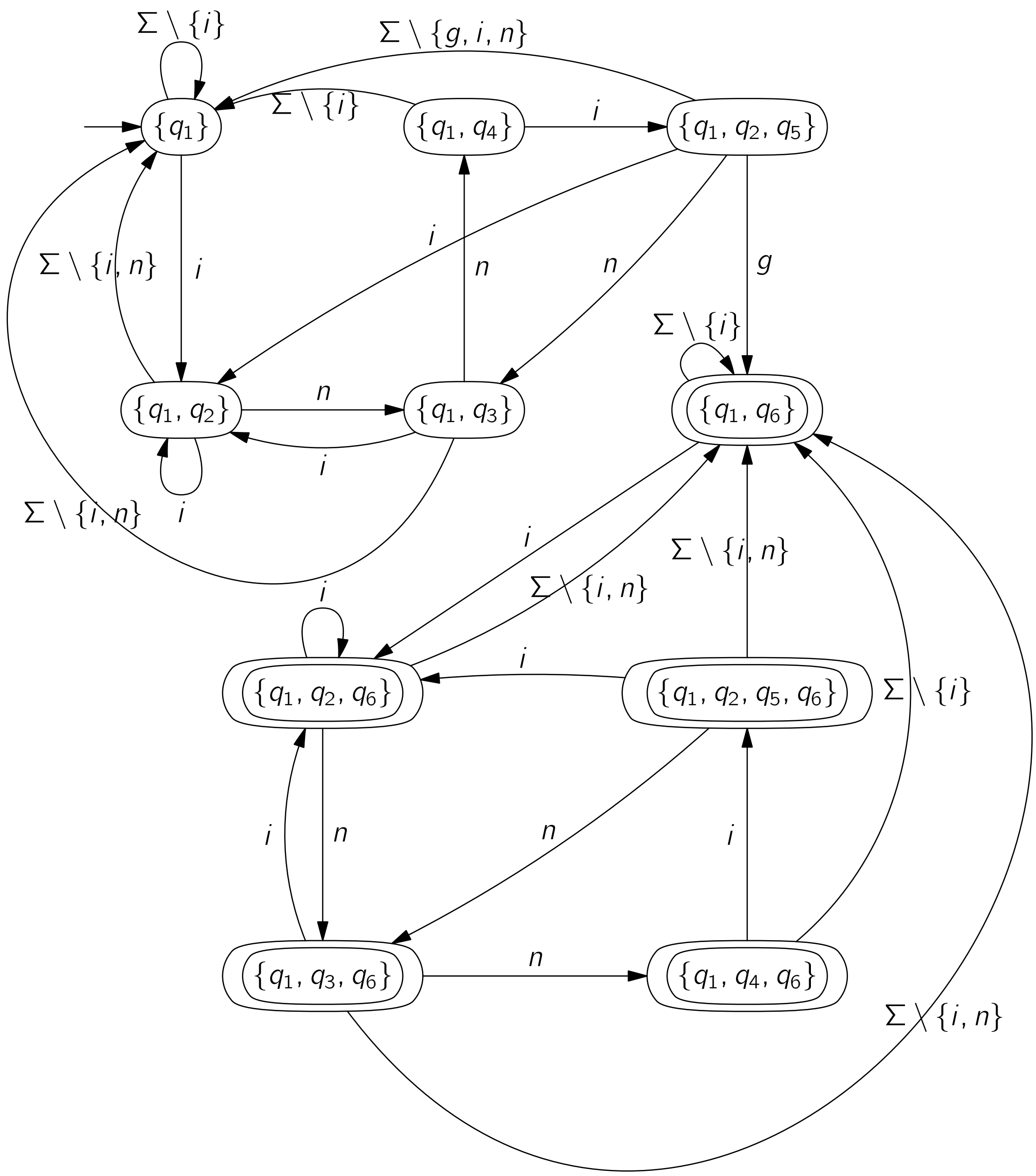


Geben Sie den Potenzautomaten für diesen NFA an, um einen DFA zu erhalten, der dieselbe Sprache akzeptiert. Wie in der Vorlesung dürfen Sie nicht erreichbare Zustände weglassen.

Lösung:

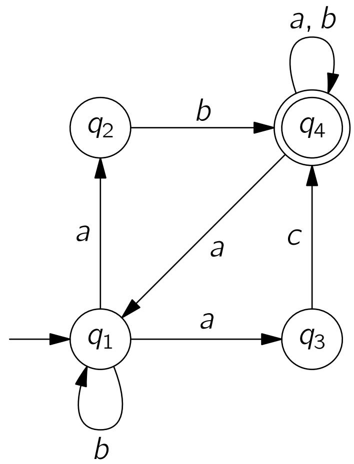
Folgende Tabelle veranschaulicht die Konstruktion des Potenzautomaten.

	g	i	n	$\Sigma \setminus \{g, i, n\}$	$\in F'$?
$\{q_1\}$	$\{q_1\}$	$\{q_1, q_2\}$	$\{q_1\}$	$\{q_1\}$	
$\{q_1, q_2\}$	$\{q_1\}$	$\{q_1, q_2\}$	$\{q_1, q_3\}$	$\{q_1\}$	
$\{q_1, q_3\}$	$\{q_1\}$	$\{q_1, q_2\}$	$\{q_1, q_4\}$	$\{q_1\}$	
$\{q_1, q_4\}$	$\{q_1\}$	$\{q_1, q_2, q_5\}$	$\{q_1\}$	$\{q_1\}$	
$\{q_1, q_2, q_5\}$	$\{q_1, q_6\}$	$\{q_1, q_2\}$	$\{q_1, q_3\}$	$\{q_1\}$	
$\{q_1, q_6\}$	$\{q_1, q_6\}$	$\{q_1, q_2, q_6\}$	$\{q_1, q_6\}$	$\{q_1, q_6\}$	✓
$\{q_1, q_2, q_6\}$	$\{q_1, q_6\}$	$\{q_1, q_2, q_6\}$	$\{q_1, q_3, q_6\}$	$\{q_1, q_6\}$	✓
$\{q_1, q_3, q_6\}$	$\{q_1, q_6\}$	$\{q_1, q_2, q_6\}$	$\{q_1, q_4, q_6\}$	$\{q_1, q_6\}$	✓
$\{q_1, q_4, q_6\}$	$\{q_1, q_6\}$	$\{q_1, q_2, q_5, q_6\}$	$\{q_1, q_6\}$	$\{q_1, q_6\}$	✓
$\{q_1, q_2, q_5, q_6\}$	$\{q_1, q_6\}$	$\{q_1, q_2, q_6\}$	$\{q_1, q_3, q_6\}$	$\{q_1, q_6\}$	✓



Hausaufgabe 4 (Potenzmengenkonstruktion): **(3 Punkte)**

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

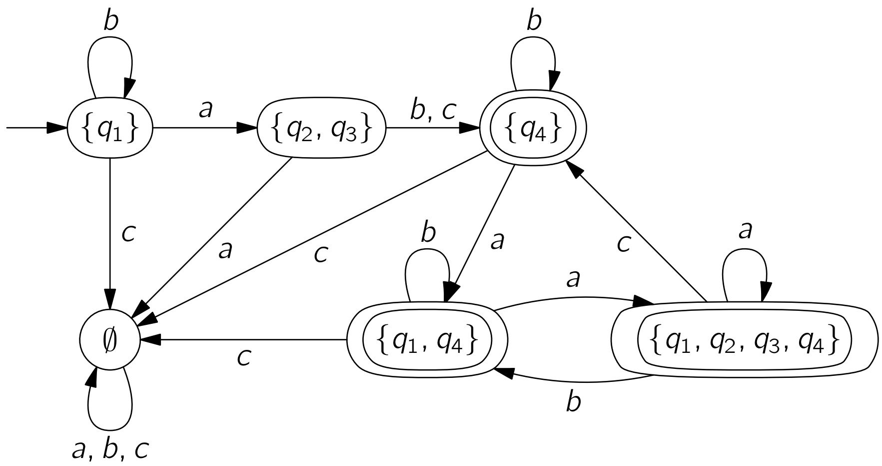


Geben Sie den Potenzautomaten für diesen NFA an, um einen DFA zu erhalten, der dieselbe Sprache akzeptiert. Wie in der Vorlesung dürfen Sie nicht erreichbare Zustände weglassen.

Lösung:

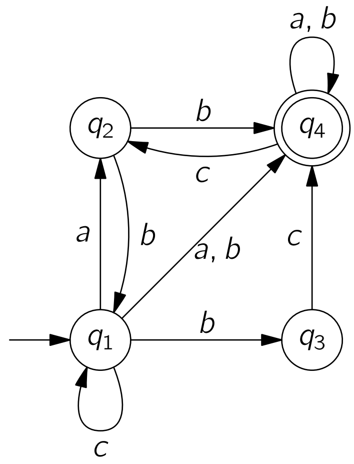
Folgende Tabelle veranschaulicht die Konstruktion des Potenzautomaten.

	a	b	c	$\in F'$?
$\{q_1\}$	$\{q_2, q_3\}$	$\{q_1\}$	$\emptyset$	
$\{q_2, q_3\}$	$\emptyset$	$\{q_4\}$	$\{q_4\}$	
$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	
$\{q_4\}$	$\{q_1, q_4\}$	$\{q_4\}$	$\emptyset$	✓
$\{q_1, q_4\}$	$\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_4\}$	$\emptyset$	✓
$\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_2, q_3, q_4\}$	$\{q_1, q_4\}$	$\{q_4\}$	✓



Tutoraufgabe 5 (Simulation):

Gegeben sei folgender NFA über dem Alphabet $\Sigma = \{a, b, c\}$.



Geben Sie an, wie sich der Algorithmus zur Simulation dieses NFAs bei Eingabe der folgenden Wörter verhält. Es genügt dabei, die Menge S nach jeder Iteration des Simulationsverfahrens aus der Vorlesung und den Rückgabewert anzugeben.

a) *ababc*

b) *bcbca*

Lösung:

a)

Iteration	S
0	$\{q_1\}$
1	$\{q_2, q_4\}$
2	$\{q_1, q_4\}$
3	$\{q_2, q_4\}$
4	$\{q_1, q_4\}$
5	$\{q_1, q_2\}$

Rückgabe: false

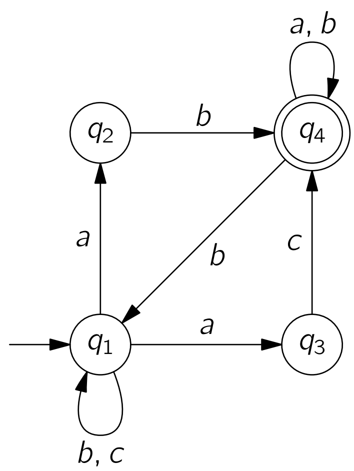
b)

Iteration	S
0	$\{q_1\}$
1	$\{q_3, q_4\}$
2	$\{q_2, q_4\}$
3	$\{q_1, q_4\}$
4	$\{q_1, q_2\}$
5	$\{q_2, q_4\}$

Rückgabe: true

Hausaufgabe 6 (Simulation): (1 + 1 = 2 Punkte)

Gegeben sei folgender NFA über dem Alphabet $\Sigma = \{a, b, c\}$.



Geben Sie an, wie sich der Algorithmus zur Simulation dieses NFAs bei Eingabe der folgenden Wörter verhält. Es genügt dabei, die Menge S nach jeder Iteration des Simulationsverfahrens aus der Vorlesung und den Rückgabewert anzugeben.

a) *bacabbac*

b) *ababca*

Lösung:

a)

Iteration	S
0	$\{q_1\}$
1	$\{q_1\}$
2	$\{q_2, q_3\}$
3	$\{q_4\}$
4	$\{q_4\}$
5	$\{q_1, q_4\}$
6	$\{q_1, q_4\}$
7	$\{q_2, q_3, q_4\}$
8	$\{q_4\}$

Rückgabe: true

b)

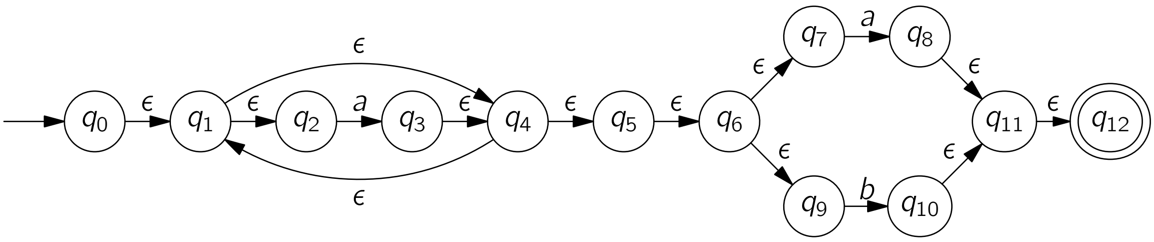
Iteration	S
0	$\{q_1\}$
1	$\{q_2, q_3\}$
2	$\{q_4\}$
3	$\{q_4\}$
4	$\{q_1, q_4\}$
5	$\{q_1\}$
6	$\{q_2, q_3\}$

Rückgabe: false

Tutoraufgabe 7 (Thompson-Konstruktion):

Erzeugen Sie mit der Thompson-Konstruktion einen NFA (mit ϵ-Transitionen) zum regulären Ausdruck $a^*(a+b)$.

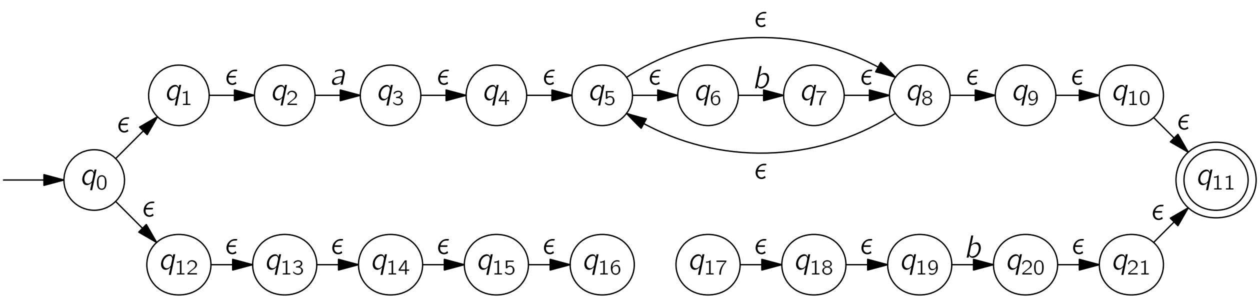
Lösung:



Hausaufgabe 8 (Thompson-Konstruktion): (3 Punkte)

Erzeugen Sie mit der Thompson-Konstruktion einen NFA (mit ϵ-Transitionen) zum regulären Ausdruck $(a \cdot b^*) + ((\epsilon \cdot \emptyset) \cdot b)$.

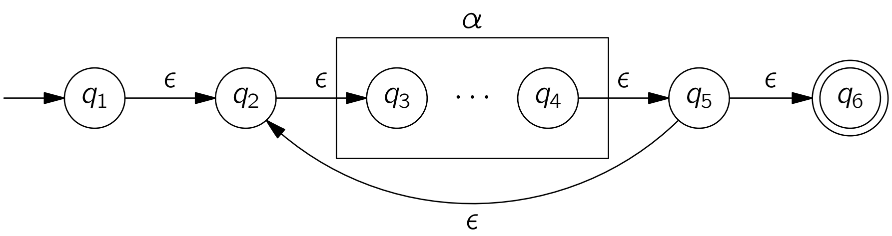
Lösung:



Tutoraufgabe 9 (Thompson-Konstruktion):

Sei α ein beliebiger regulärer Ausdruck. In der Vorlesung wurde die Thompson-Konstruktion für α^*, allerdings nicht für α^+ vorgestellt. Da $\alpha^+ := \alpha \cdot \alpha^*$ definiert wurde, ist das allerdings kein großes Problem. Trotzdem ist es manchmal hilfreich, die Thompson-Konstruktion auch *direkt* für α^+ anwenden zu können.
Erweitern Sie die Thompson-Konstruktion so, dass für jeden Ausdruck α^+ *direkt* ein Automat konstruiert wird, der auf dem Automaten für α basiert. Hierbei sollte der für α^+ erzeugte Automat weniger Transitionen haben als der Automat, der von der Thompson-Konstruktion für α^* erzeugt wird. Der entstehende Automat sollte nur einen Endzustand haben, der Endzustand sollte keine ausgehenden Kanten und der Startzustand sollte keine eingehenden Kanten haben.

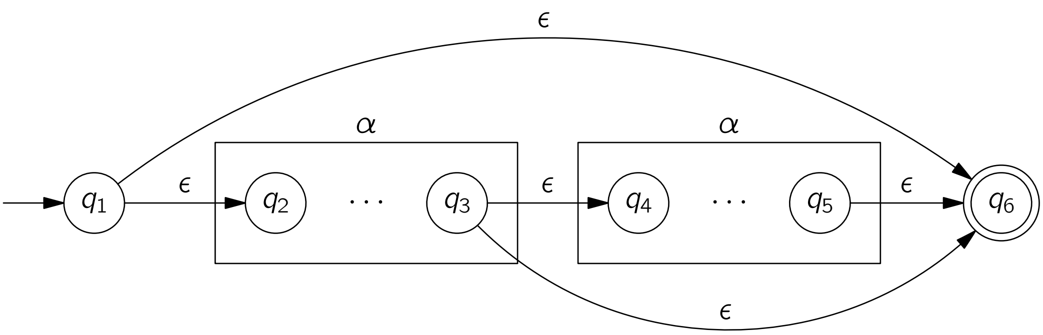
Lösung:



Hausaufgabe 10 (Thompson-Konstruktion): (3 Punkte)

Sei α ein beliebiger regulärer Ausdruck. Um Wiederholungen der von diesem Ausdruck erkannten Wörter auszudrücken, gibt es die regulären Ausdrücke ϵ (α 0-mal), α (α 1-mal), α^+ (α min. 1-mal) und α^* (α beliebig oft).
Unter Umständen ist es aber auch interessant, ein eigenes Konstrukt für die 0-malige bis 2-malige Anwendung von α zu haben. Sei $\alpha^\square := \epsilon + \alpha + \alpha \cdot \alpha$ eine entsprechende Erweiterung der regulären Ausdrücke.
Erweitern Sie die Thompson-Konstruktion so, dass für jeden Ausdruck $\alpha^\square$ **direkt** ein Automat konstruiert wird, der auf dem Automaten für α basiert. Hierbei sollte der für $\alpha^\square$ erzeugte Automat maximal so viele **Zustände** haben wie der Automat, der von der Thompson-Konstruktion für $\alpha \cdot \alpha$ erzeugt wird. Der entstehende Automat sollte nur einen Endzustand haben, der Endzustand sollte keine ausgehenden Kanten und der Startzustand sollte keine eingehenden Kanten haben.
Hinweis: Bis auf den Endzustand darf jeder Zustand beliebig viele ausgehende Transitionen haben. Außerdem darf bis auf den Startzustand jeder Zustand beliebig viele eingehende Transitionen haben.

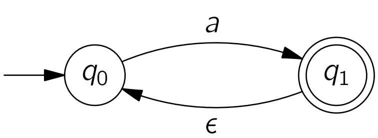
Lösung:



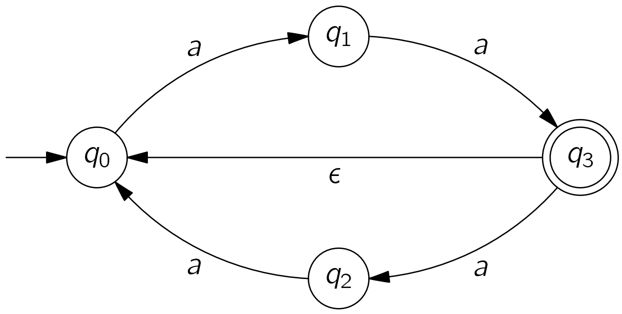
Tutoraufgabe 11 (ϵ-Kanten entfernen):

Sei $\Sigma := \{a, b, c, d\}$ ein Alphabet. Geben sie zu den folgenden ϵ-NFAs jeweils einen NFA ohne ϵ-Transitionen an, der dieselbe Sprache erkennt.

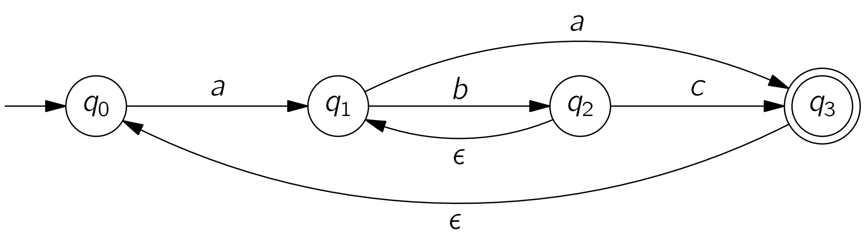
a)



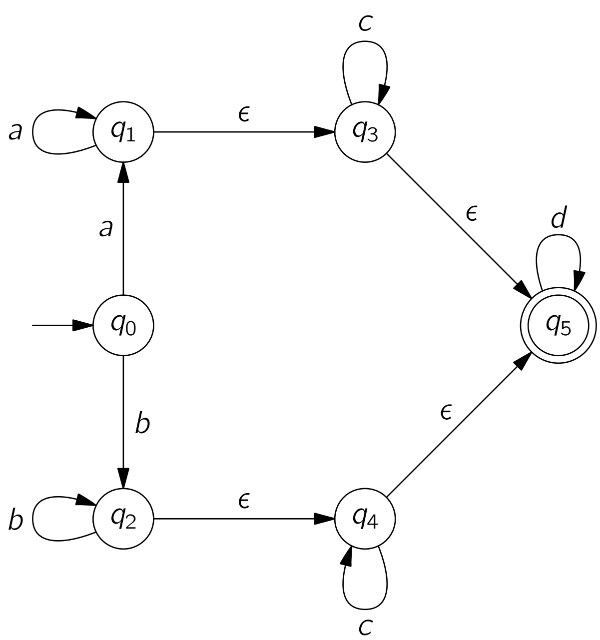
b)


a
q_1
a
q_0
ϵ
q_3
a
q_2
a

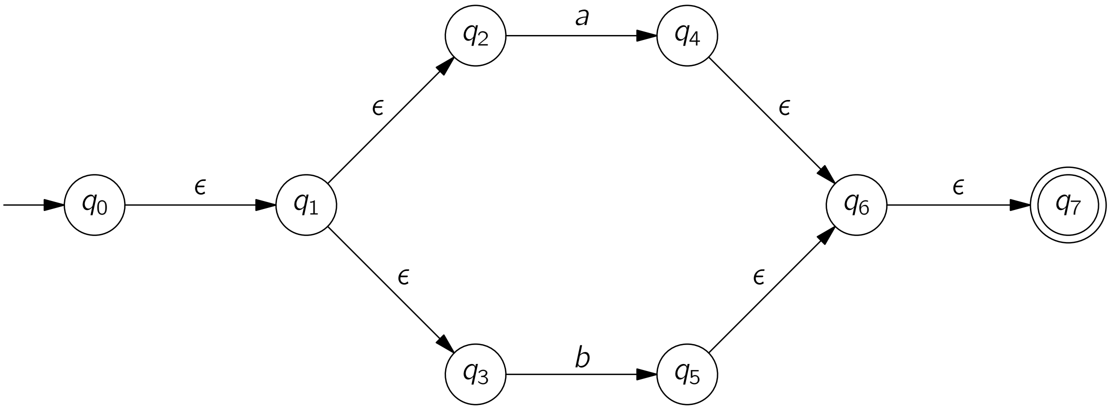
c)


a
q_0
a
q_1
b
q_2
c
q_3
ϵ
ϵ

d)


c
a
q_1
ϵ
q_3
ϵ
d
a
q_0
q_5
b
ϵ
b
q_2
ϵ
q_4
c

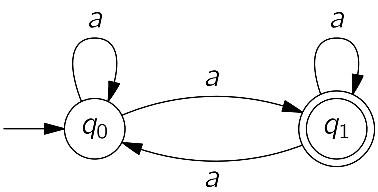
e)



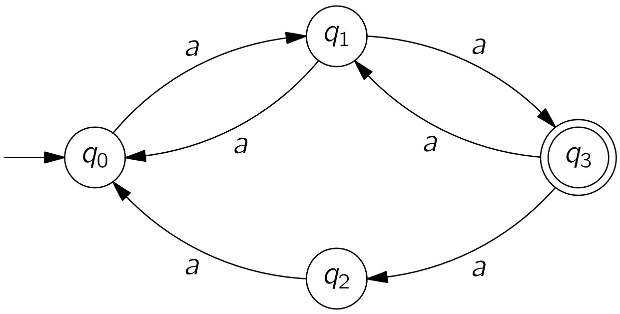
Lösung:

Sei $\Sigma := \{a, b, c, d\}$ ein Alphabet. Geben sie zu den folgenden ϵ-NFAs jeweils einen NFA ohne ϵ-Transitionen an:

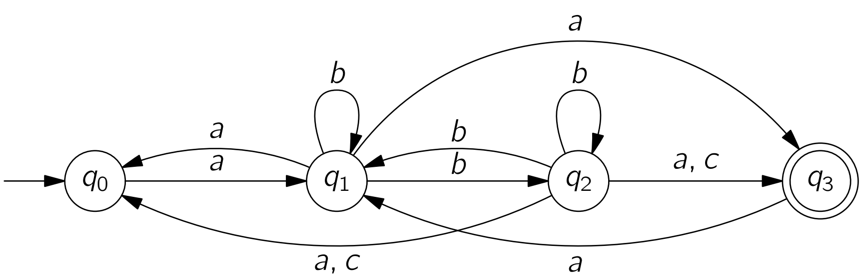
a)



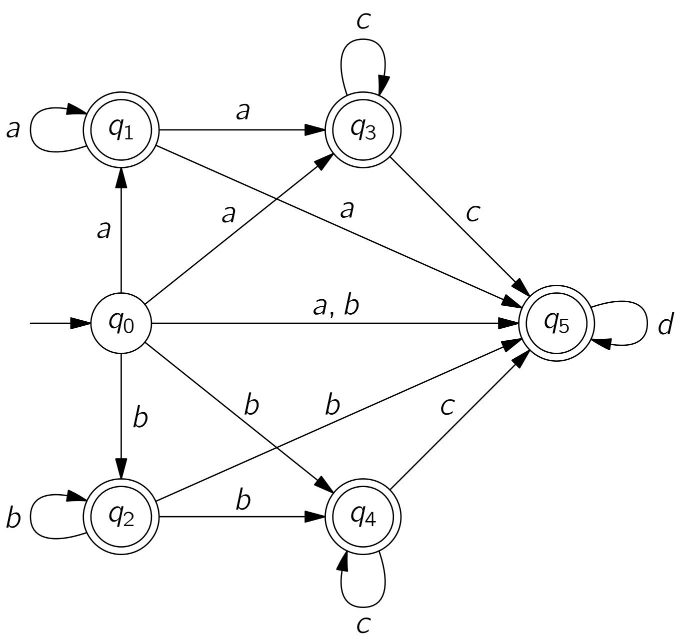
b)



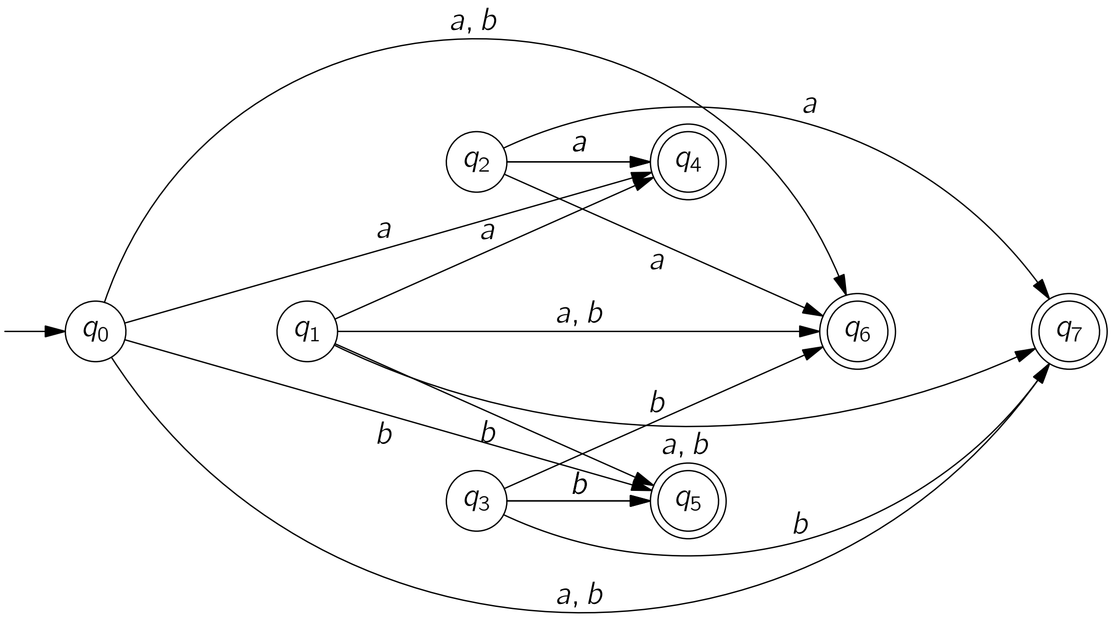
c)



d)



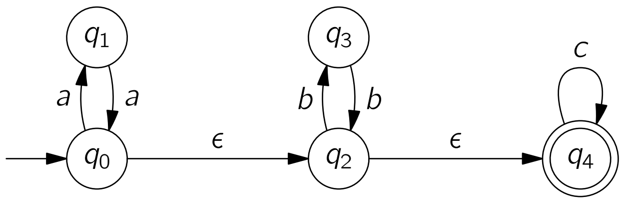
e)



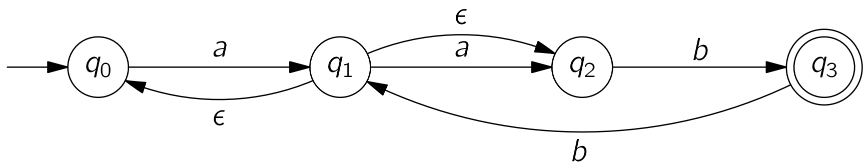
Hausaufgabe 12 (ϵ-Kanten entfernen): (2 + 2 = 4 Punkte)

Sei $\Sigma := \{a, b, c, d\}$ ein Alphabet. Geben sie zu den folgenden ϵ-NFAs jeweils einen NFA ohne ϵ-Transitionen an, der dieselbe Sprache erkennt.

a)

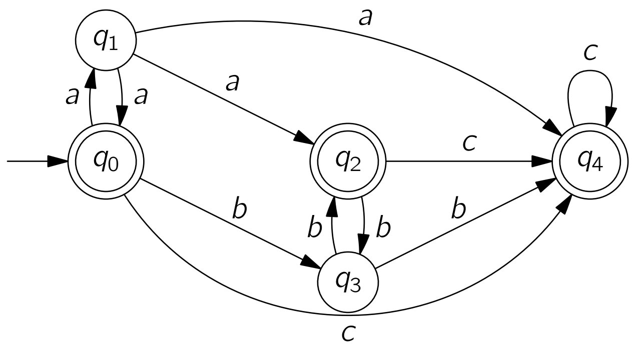

q_1
q_3
c
a
a
b
b
ϵ
ϵ
q_0
q_2
q_4

b)

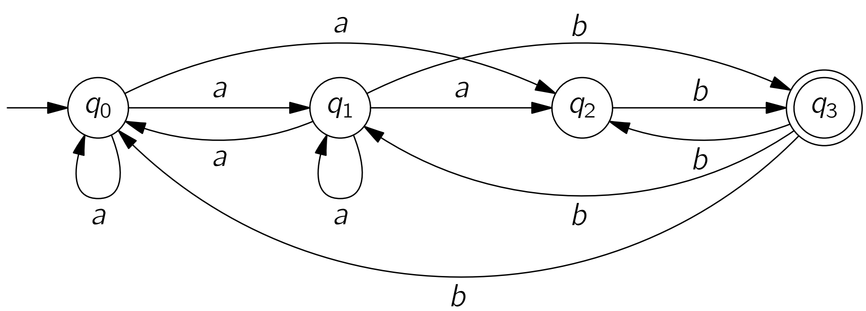

ϵ
a
a
b
q_0
q_1
q_2
q_3
ϵ
b

Lösung:

a)


q_1
a
c
a
a
a
c
q_0
q_2
q_4
b
b
b
b
q_3
c

b)


a
b
a
a
b
q_0
q_1
q_2
q_3
a
b
a
a
b
b

Tutoraufgabe 13 (Abschlusseigenschaften):

Die Sprache $L = \{a^n b^n \mid n \geq 0\}$ über dem Alphabet $\Sigma = \{a, b\}$ ist nicht regulär. Benutzen Sie die Abschlusseigenschaften regulärer Sprachen, um zu zeigen, dass folgende Sprachen ebenfalls nicht regulär sind.
Hinweis: Sie dürfen für jede Teilaufgabe benutzen, dass die Sprachen vorheriger Teilaufgaben nicht regulär sind.

a) $L_1 = \{a^m b^n \mid m, n \geq 0, m + n > 0, m \neq n\} \cup \Sigma^* ba \Sigma^*$

b) $L_2 = \{a^m b^n \mid m, n \geq 0, m \neq n\} \cup \Sigma^* ba \Sigma^*$

c) $L_3 = \{b^n a^n \mid n \geq 0\}$

Lösung:

a) $L = \Sigma^* \setminus L_1$.

b) $L_1 = L_2$.

c) $L = h(L_3)$ mit $h(a) = b$ und $h(b) = a$. Die Funktion h ist ein Homomorphismus, da offensichtlich $h(ww') = h(w)h(w')$. In der Tat ist h sogar ein Isomorphismus.