Formale Grundlagen der Informatik I
4. Übungsblatt

Fachbereich Mathematik
Prof. Dr. Martin Ziegler
Alexander Kreuzer
Carsten Rösnick

SS 2011
04.05.11

Minitest Lösung

Bestimmen Sie die korrekten Implikationen. Sei $L \subseteq \Sigma^*$ eine beliebige Sprache. Dann gilt:

L ist regulär

a) $\square \Rightarrow$ $\boxtimes \Leftarrow$ L ist endlich

Begründung: Besteht L nur aus den endlich vielen Elementen $w_1, \dots, w_n$, dann kann L durch den regulären Ausdruck $w_1 + w_2 + \cdots + w_n$ beschrieben werden. Umgekehrt ist Σ^* regulär, aber nicht endlich.

b) $\boxtimes \Rightarrow$ $\boxtimes \Leftarrow$ L wird von einem DFA akzeptiert

Begründung: Satz von Kleene (Satz 2.3.1 im Skript).

c) $\boxtimes \Rightarrow$ $\boxtimes \Leftarrow$ L wird von einem NFA akzeptiert

Begründung: Satz von Kleene (Satz 2.3.1 im Skript).

d) $\boxtimes \Rightarrow$ $\square \Leftarrow$ L enthält eine reguläre Sprache,
d.h. es gibt eine reguläre Sprache $L_1 \subseteq \Sigma^*$ mit $L_1 \subseteq L$

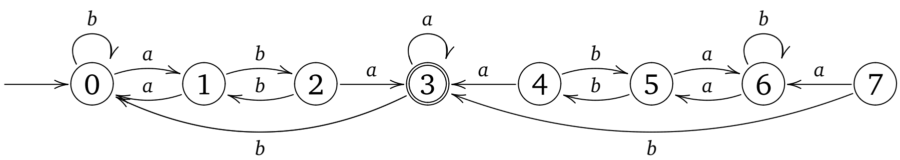
Begründung: Hinrichtung ist klar mit $L = L_1$. Rückrichtung: Jede Sprache L enthält die reguläre Sprache $\emptyset$, aber nicht jede Sprache ist regulär.

e) $\boxtimes \Rightarrow$ $\square \Leftarrow$ L ist Teilmenge einer regulären Sprache,
d.h. es gibt eine reguläre Sprache $L_2 \subseteq \Sigma^*$ mit $L \subseteq L_2$

Begründung: Hinrichtung ist klar mit $L = L_2$. Rückrichtung: Wie d) aber mit Σ^*.

Gruppenübung

Aufgabe G1 (DFA Minimierung)
Betrachten Sie den folgenden DFA:



Gegeben ist die folgende unvollständige Tabelle für die Relation $\not\sim$. (Ein $\times$ an der Stelle p,q in der Tabelle bedeutet, dass $p \not\sim q$.) Vervollständigen Sie die Tabelle und geben Sie ggf. ein Wort an, für das diese Unterscheidung notwendig ist, d.h. ein Wort w, das zu L_q gehört, aber nicht zu $L_{q'}$ (oder umgekehrt), wobei $L_q := \{w \in \Sigma^* \mid \hat{\delta}(q,w) \in A\}$.

$\not\sim$	0	1	2	3	4	5	6	7
0			×	×	×			×
1			×	×	×			×
2	×	×		×		×	×	×
3	×	×	×		×	×	×	×
4	×	×		×		×	×	×
5			×	×	×		×	×
6			×	×	×	×		×
7	×	×	×	×	×	×	×	

Lösungsskizze:

$\not\sim$	0	1	2	3	4	5	6	7
0		×	×	×	×	×		×
1	×		×	×	×		×	×
2	×	×		×		×	×	×
3	×	×	×		×	×	×	×
4	×	×		×		×	×	×
5	×		×	×	×		×	×
6		×	×	×	×	×		×
7	×	×	×	×	×	×	×	

(p,q)	w
$(0,1)$	ba
$(0,5)$	aba
$(1,6)$	ba

Die Diagonale in der Tabelle bleibt frei, da $\sim$ (siehe Skript S. 39) reflexiv ist.

Aufgabe G2 (Umkehrung regulärer Sprachen)
Zeigen Sie, dass für jede reguläre Sprache L auch die Umkehrung $\text{rev}(L)$ regulär ist, indem Sie zeigen, wie man aus einem regulären Ausdruck für die Sprache L einen regulären Ausdruck für $\text{rev}(L)$ gewinnen kann. Zur Erinnerung: Die Sprache $\text{rev}(L)$ ist definiert als

$$\text{rev}(L) := \{w^{-1} \in \Sigma^* \mid w \in L\}.$$

Lösungsskizze: Man kann induktiv über den Aufbau regulärer Ausdrücke $\alpha \in \text{REG}(\Sigma)$ einen neuen regulären Ausdruck $\text{rv}(\alpha)$ konstruieren, wie folgt:

(i) $\text{rv}(\emptyset) := \emptyset$,

(ii) $\text{rv}(a) := a$ für jedes $a \in \Sigma$,

(iii) $\text{rv}(\alpha + \beta) := \text{rv}(\alpha) + \text{rv}(\beta)$,

(iv) $\text{rv}(\alpha\beta) := \text{rv}(\beta)\text{rv}(\alpha)$,

(v) $\text{rv}(\alpha^*) := (\text{rv}(\alpha))^*$.

Wir zeigen per Induktion dann, dass für alle $\alpha \in REG(\Sigma)$ gilt

$$L(\text{rv}(\alpha)) = \text{rev}(L(\alpha)). \tag{1}$$

Für den Induktionsanfang muss gezeigt werden, dass (1) für $\alpha = \emptyset$ und $\alpha = a$ für $a \in \Sigma$ gilt. Dies folgt sofort aus der Definition von rv.
Für die Induktionsschritt muss gezeigt werden, dass falls (1) für $\beta, \gamma \in \text{REG}(\Sigma)$ eingesetzt für α gilt, dann auch (1) für $\alpha = \beta\gamma$, $\alpha = \beta + \gamma$ und $\alpha = \beta^*$ gilt. Wir zeigen hier nur den Fall $\alpha = \beta\gamma$: Gelte also

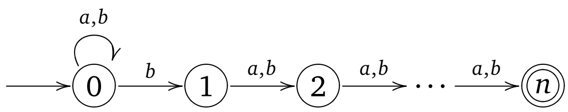
$$L(\text{rv}(\beta)) = \text{rev}(L(\beta)) \quad \text{und} \quad L(\text{rv}(\gamma)) = \text{rev}(L(\gamma)) \tag{2}$$

dann gilt

$$\begin{aligned} L(\mathrm{rv}(\beta\gamma)) &= L(\mathrm{rv}(\gamma)\mathrm{rv}(\beta)) && \text{nach Definition von rv} \\ &= L(\mathrm{rv}(\gamma)) \cdot L(\mathrm{rv}(\beta)) \\ &= \mathrm{rev}(L(\gamma)) \cdot \mathrm{rev}(L(\beta)) && \text{nach (2)} \\ &= \mathrm{rev}(L(\beta) \cdot L(\gamma)) && \text{nach Definition von rev} \\ &= \mathrm{rev}(L(\beta\gamma)). \end{aligned}$$

Also gilt (1) für $\alpha = \beta\gamma$.

Aufgabe G3 (NFA, DFA Vergleich)
Betrachten Sie den folgenden NFA $\mathscr{A}_n$:



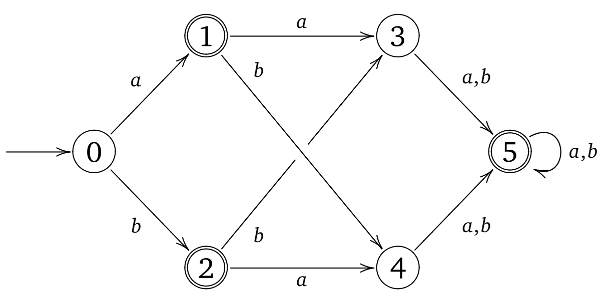
(a) Bestimmen Sie $L(\mathscr{A}_n)$.
(b) Zeigen Sie, dass es keinen äquivalenten DFA gibt mit weniger als 2^n Zuständen.

Lösungsskizze:

(a) $L(\mathscr{A}_n)$ is die Menge von a/b-Folgen, an deren n-ter Position vor dem Ende ein b steht.
(b) Nehmen wir an, $\mathscr{Q}$ ist ein äquivalenter DFA mit weniger als 2^n Zuständen. Dann gäbe es einen Zustand q und zwei verschiedene Zeichenreihen $x_1x_2 \dots x_n$ und $y_1y_2 \dots y_n$, sodass sich $\mathscr{Q}$ nach dem Einlesen sowohl von $x_1x_2 \dots x_n$ als auch von $y_1y_2 \dots y_n$ im Zustand q befände („Schubfachprinzip"). Da die Zeichenreihen verschieden sind, müssen sie sich an einer bestimmten Position unterscheiden; sei $x_i \neq y_i$. Angenommen (auf Grund der Symmetrie ohne Beschränkung der Allgemeinheit), $x_i = b$ und $y_i = a$. Wenn $i = 1$, dann muss q sowohl ein akzeptierender Zustand als auch ein nicht akzeptierender Zustand sein, da $x_1x_2 \dots x_n$ akzeptiert wird (das n-te Zeichen vor dem Ende ist b) und $y_1y_2 \dots y_n$ nicht. Ist $i > 1$, dann betrachten wir den Zustand p, in den $\mathscr{Q}$ nach dem Einlesen sowohl von $x_1x_2 \dots x_n a^{i-1}$ als auch von $y_1y_2 \dots y_n a^{i-1}$ käme. Da $x_i = b$ und $y_i = a$, müsste $x_1x_2 \dots x_n a^{i-1}$ akezptiert werden (das n-te Zeichen vor dem Ende ist $x_i = b$) und $y_1y_2 \dots y_n a^{i-1}$ nicht, d.h. p müsste ein akzeptierender und zugleich ein nicht akzeptierender Zustand sein.

Hausübung

Aufgabe H1 (Minimalautomaten und Minimierung) (4 Punkte)
Finden Sie einen äquivalenten DFA minimaler Größe für den folgenden DFA. Geben Sie im Zuge der Lösung auch die Relationen $\not\sim_i$ (für alle notwendigen i) explizit an.



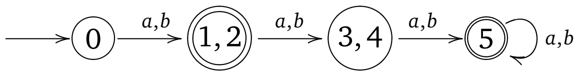
Lösungsskizze: Wir bestimmen die Relationen $\not\sim_i$.

$\not\sim_0$	0	1	2	3	4	5
0		×	×			×
1	×			×	×	
2	×			×	×	
3		×	×			×
4		×	×			×
5	×			×	×	

$\not\sim_1$	0	1	2	3	4	5
0		×	×			×
1	×			×	×	×
2	×			×	×	×
3		×	×			×
4		×	×			×
5	×	×	×	×	×	

$\not\sim_2$	0	1	2	3	4	5
0		×	×	×	×	×
1	×			×	×	×
2	×			×	×	×
3	×	×	×			×
4	×	×	×			×
5	×	×	×	×	×	

Da $\not\sim_2 = \not\sim_3$ ist die Relation $\not\sim$ durch die letzte Tabelle gegeben. Das heißt, dass wir die Zustände 1 und 2, bzw. 3 und 4 identifizieren können. Deshalb sieht der DFA minimaler Größe wie folgt aus:



Aufgabe H2 (Abgeschlossenheit der regulären Sprachen) (4 Punkte)
Beweisen oder widerlegen Sie: Die Menge der regulären Sprachen ist abgeschlossen unter den folgenden Operationen:

(a) In jedem Wort werden alle Buchstaben a durch b ersetzt und alle b durch a.

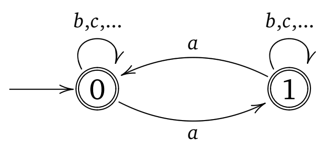
(b) Jedes zweite Vorkommen des Buchstaben a wird durch das Wort aba ersetzt.

(Extra) Die Buchstaben in jedem Wort dürfen beliebig umsortiert werden, d.h. ist etwa das Wort $aaba$ in der Sprache, so fügen wir auch die Wörter $aaab$, $abaa$ und $baaa$ hinzu.

Lösungsskizze:

(a) Die Menge der regulären Sprachen ist unter dieser Operation abgeschlossen. Hat man einen regulären Ausdruck für eine Sprache, so kann man daraus einen Ausdruck für die neue Sprache konstruieren, indem man jedes a durch b und jedes b durch a ersetzt.

(b) Die Menge der regulären Sprachen ist unter dieser Operation abgeschlossen. Aus einem DFA für die alte Sprache können wir wie folgt einen Automaten für die neue Sprache konstruieren. Zuerst bilden wir das Produkt mit dem Automaten



um die Anzahl der a zu zählen. Im resultierenden Automaten können wir jede a-Transition

$$(q,1) \longrightarrow (p,0),$$

welche bei einem Zustand mit zweiter Komponente 1 beginnt, ersetzen durch eine Folge van Transitionen

$$(q,1) \xrightarrow{a} \bullet \xrightarrow{b} \bullet \xrightarrow{a} (p,0).$$

(Hierzu müssen wir für jede solche Transition zwei neue Zwischenzustände einführen.)

(Extra) Die Menge der regulären Sprache ist unter dieser Operation nicht abgeschlossen. Wenn wir diese Operation auf die sprache $L_0 = L((ab)^*)$ anwenden, erhalten wir die Sprache $L_1 = \{w \in \{a,b\}^* : |w|_a = |w|_b\}$. Nach dem Schnitt mit $L(a^*b^*)$ erhalten wir

$$L_2 = L_1 \cap L(a^*b^*) = \{a^n b^n : n \in \mathbb{N}\}.$$

Wäre die Menge der regulären Sprachen unter der Operation abgeschlossen, dann wären auch L_1 und L_2 regulär. Aber wir wissen, dass L_2 nicht regulär ist.