2. Übungsblatt zu
Grundzüge der Theoretischen Informatik 1, WS 2013/14

Prof. Dr. Gert Smolka, Jonas Kaiser, M.Sc.
www.ps.uni-saarland.de/courses/ti-ws13/

Lesen Sie im Buch: Lecture 5 – Lecture 6

Aufgabe 2.1 Kennen Sie die grundlegenden Definitionen für Automaten?

(a) Sei $M = (Q, \Sigma, \delta, s, F)$ ein DFA. Definieren Sie die Funktion $\hat{\delta} : Q \times \Sigma^* \to Q$ sowie die von M erkannte Sprache $\mathcal{L}(M)$.
(b) Sei $N = (Q, \Sigma, \Delta, S, F)$ ein NFA (nicht-deterministischer Automat). Definieren Sie die Funktion $\hat{\Delta} : 2^Q \times \Sigma^* \to 2^Q$ sowie die von N erkannte Sprache $\mathcal{L}(N)$.

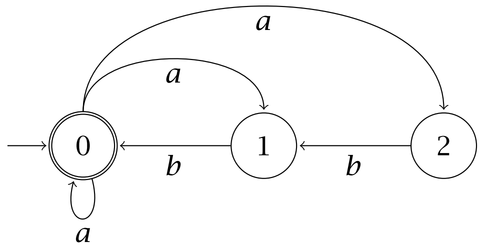
Aufgabe 2.2 Sei $M = (Q, \Sigma, \delta, s, F)$ ein DFA. Wie in der Vorlesung erwähnt, ist es möglich, dass M nicht-erreichbare Zustände enthält. Ziel dieser Aufgabe ist der formale Entwurf einer Konstruktion auf Automaten, welche diese „unnötigen" Zustände entfernt. Verfahren Sie hierfür wie folgt:

- Definieren Sie $Q' \subseteq Q$, die Menge der erreichbaren Zustände in M. Nutzen Sie hierfür die Funktion $\hat{\delta}$.
- Definieren Sie δ', s' und F', so dass der Automat $M' = (Q', \Sigma, \delta', s', F')$ die selbe Sprache erkennt wie der ursprüngliche Automat M, formal $\mathcal{L}(M) = \mathcal{L}(M')$.

Aufgabe 2.3 Ist folgende Aussage wahr oder falsch? Begründen Sie ihre Antwort.

> „NFAs erkennen mehr Sprachen als DFAs, d.h. die Sprachklasse $\mathcal{NFA}$ ist echt größer als die Sprachklasse $\mathcal{DFA}$"

Aufgabe 2.4 Betrachten Sie folgenden NFA N über $\Sigma = \{a, b\}$:



(a) Geben Sie ein Wort $x \in \Sigma^*$ an, welches von N *nicht* erkannt wird.
(b) Geben Sie eine informelle Beschreibung der von N erkannte Sprache $\mathcal{L}(N)$ an.
(c) Konvertieren Sie N mit Hilfe der Teilmengenkonstruktion in einen DFA M, welcher die selbe Sprache wie N erkennt. Geben Sie Ihre Lösung als Übergangstabelle an.
(d) Entfernen Sie aus M alle nicht-erreichbaren Zustände und geben Sie den verbleibenden Automaten M' als Diagramm an.

Aufgabe 2.5 Sei

$$
\begin{aligned}
N = (&\{0,1\},\\
&\{a,b\},\\
&\{((0,a),\{0,1\}),((0,b),\emptyset),((1,a),\emptyset),((1,b),\{0,1\})\},\\
&\{0\},\\
&\{0\})
\end{aligned}
$$

ein NFA. Geben Sie das Diagram von N an. Geben Sie dann mit Hilfe der Teilmengenkonstruktion einen DFA M an, der die selbe Sprache erkennt wir N. Löschen Sie unerreichbare Zustände.

Aufgabe 2.6 Die Funktion **rev** $: \Sigma^* \to \Sigma^*$ reversiert ein Wort über das Alphabet Σ und ist wie folgt definiert:

$$
\begin{aligned}
\mathbf{rev}\ \epsilon &\stackrel{\text{def}}{=} \epsilon\\
\mathbf{rev}\ ax &\stackrel{\text{def}}{=} (\mathbf{rev}\ x)a
\end{aligned}
$$

Analog dazu definiert man **rev** A für eine Sprache $A \subseteq \Sigma^*$ wie folgt:

$$\mathbf{rev}\ A \stackrel{\text{def}}{=} \{\mathbf{rev}\ x \mid x \in A\}$$

Es gilt also zum Beispiel **rev** $\{a, ab, aab, aaab\} = \{a, ba, baa, baaa\}$.

(a) Sei $M = (Q, \Sigma, \delta, s, F)$ ein DFA. Definieren Sie einen NFA $N = (Q', \Sigma, \Delta, S, F')$, so dass $\mathcal{L}(N) = \mathbf{rev}\ \mathcal{L}(M)$. Gesucht ist also eine Konstruktion **Rev** auf Automaten, so dass $\mathcal{L}(\mathbf{Rev}\ M) = \mathbf{rev}\ \mathcal{L}(M)$. **Tip:** Überlegen Sie sich, was Sie ändern müssen, wenn Sie „rückwärts“ durch einen Automaten laufen wollen.

(b) Sei $\Sigma = \{a, b, c\}$ und A wie folgt definiert.

$$A \stackrel{\text{def}}{=} \{x \in \Sigma^* \mid (\text{Das letzte Zeichen in } x \text{ ist } a) \vee (\text{Das vorletzte Zeichen in } x \text{ ist } b)\}$$

Konstruieren Sie einen DFA M, so dass $\mathcal{L}(M) = A$ wie folgt:

- Konstruieren Sie zuerst einen DFA M_0, der (**rev** A) erkennt. Beachten Sie, dass für diese Konstruktion das erste und das zweite Zeichen eines Wortes relevant sind.
- Geben Sie das Diagram des NFA $N_0 = (\mathbf{Rev}\ M_0)$ an.
- Nutzen Sie die Teilmengenkonstruktion um einen DFA M zu erhalten, welcher die selbe Sprache erkennt wie N_0. Geben Sie das Diagram von M an und entfernen Sie unerreichbare Zustände. Unter der Annahme das $\mathbf{rev}\ (\mathbf{rev}\ A) = A$ gilt also $\mathcal{L}(M) = \mathcal{L}(N_0) = \mathcal{L}(\mathbf{Rev}\ M_0) = \mathbf{rev}\ \mathcal{L}(M_0) = \mathbf{rev}\ (\mathbf{rev}\ A) = A$, wie gewünscht.

Aufgabe 2.7 Sei $N = (Q, \Sigma, \Delta, S, F)$ ein NFA und sei $\hat{\Delta} : 2^Q \times \Sigma^* \to 2^Q$ wie folgt definiert:

$$
\begin{aligned}
\hat{\Delta}(A, \epsilon) &\stackrel{\text{def}}{=} A\\
\hat{\Delta}(A, ax) &\stackrel{\text{def}}{=} \hat{\Delta}\Big(\bigcup_{q \in A} \Delta(q, a), x\Big)
\end{aligned}
$$

Beweisen Sie mittels struktureller Induktion über x, dass

$$\forall x, y \in \Sigma^*, \forall A \in 2^Q, \hat{\Delta}(A, xy) = \hat{\Delta}(\hat{\Delta}(A, x), y).$$

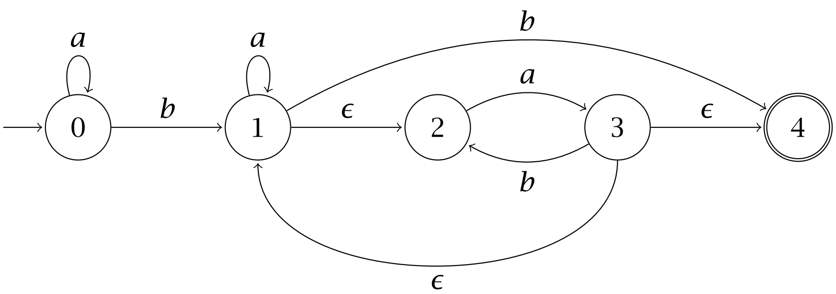
Aufgabe 2.8 Seien Sprachen $A, B \subseteq \{a, b, c\}^*$ wie folgt definiert:

$$A = \{ab, ac\}$$
$$B = \{b^n \mid 0 \leq n\}$$

Geben Sie für jede der folgenden Sprachen einen NFA als Diagram an, welcher die Sprache erkennt. Sie dürfen ϵ-Übergänge verwenden.

(a) A
(b) B
(c) AB
(d) A^3
(e) A^*

Aufgabe 2.9 Betrachten Sie folgenden NFA N mit ϵ-Übergängen. Geben Sie einen NFA N' ohne ϵ-Übergängen an, mit $\mathcal{L}(N') = \mathcal{L}(N)$. Hierzu müssen Sie die ϵ-Übergänge entfernen und durch geeignete, neue Übergänge ersetzen.



Tip: Die neuen Übergänge verbinden nicht notwendigerweise die selben Zustände welche zuvor durch den ϵ-Übergang verbunden waren.