Institut für Theoretische Informatik
Lehrstuhl Prof. Dr. D. Wagner

Übungsblatt 2

Vorlesung Theoretische Grundlagen der Informatik im WS 19/20

Ausgabe: 5. November 2019
Abgabe: 19. November 2019, 11:00 Uhr (im Kasten im UG von Gebäude 50.34)

Aufgabe 1

(2 + 2 = 4 Punkte)

(a) Geben Sie einen Algorithmus an, der für einen gegebenen vollständigen DEA $\mathcal{A} = (Q, \Sigma, \delta, s, F)$ in Zeit $\mathcal{O}(|Q||\Sigma|)$ entscheidet, ob $L(\mathcal{A}) = \emptyset$ gilt. Begründen Sie die Laufzeit!

(b) Geben Sie einen Algorithmus an, der für zwei gegebene vollständige DEAs $\mathcal{A}_1 = (Q_1, \Sigma, \delta_1, s_1, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \delta_2, s_2, F_2)$ in Zeit $\mathcal{O}(|Q_1||Q_2||\Sigma|)$ entscheidet, ob $L(\mathcal{A}_1) = L(\mathcal{A}_2)$ gilt. Begründen Sie die Laufzeit!

Lösung:

(a) Führe im Zustandsübergangsgraph von $\mathcal{A}$ eine Breiten- oder Tiefensuche von s aus. Es gilt genau dann $L(\mathcal{A}) = \emptyset$, wenn dabei kein Endzustand erreicht wird. Die Laufzeit einer Breiten-/Tiefensuche ist $\mathcal{O}(|V| + |E|)$, wobei $|V| = |Q|$ die Anzahl der Knoten ist und $|E| \leq |Q||\Sigma|$ die Anzahl der Kanten.

(b) Es gilt:

$$\begin{aligned} L(\mathcal{A}_1) = L(\mathcal{A}_2) \quad &\Leftrightarrow \quad L(\mathcal{A}_1) \subseteq L(\mathcal{A}_2) \quad \wedge \quad L(\mathcal{A}_2) \subseteq L(\mathcal{A}_1) \\ &\Leftrightarrow \quad L(\mathcal{A}_1) \cap \overline{L(\mathcal{A}_2)} = \emptyset \quad \wedge \quad L(\mathcal{A}_2) \cap \overline{L(\mathcal{A}_1)} = \emptyset \end{aligned}$$

In der Übung wurde gezeigt, wie Automaten für die Komplementsprache bzw. den Schnitt zweier Sprachen konstruiert werden können. Konstruiere also die Automaten für $L(\mathcal{A}_1) \cap \overline{L(\mathcal{A}_2)}$ und $L(\mathcal{A}_2) \cap \overline{L(\mathcal{A}_1)}$ und wende darauf jeweils den Algorithmus aus Aufgabenteil (a) an.

Beim Komplementautomaten bleibt die Anzahl der Zustände gleich. Für zwei Automaten mit $|Q_1|$ und $|Q_2|$ vielen Zuständen hat der Schnittautomat $|Q_1 \times Q_2| = |Q_1||Q_2|$ viele Zustände. Damit ergibt sich die gewünschte Laufzeit.

Aufgabe 2

(4 Punkte)

Gegeben sei ein endliches Alphabet Σ. Das *Shuffle-Produkt* von zwei Wörtern $u, v \in \Sigma^*$ ist wie folgt definiert:

$$u \between v := \{u_1 v_1 u_2 v_2 \dots u_n v_n \mid n \in \mathbb{N}_0, u_1, \dots, u_n, v_1, \dots, v_n \in \Sigma^*, u_1 \dots u_n = u, v_1 \dots v_n = v\}$$

Die beiden Wörter werden also auf (fast) beliebige Weise durchmischt, wobei nur gefordert wird, dass die interne Reihenfolge von u bzw. v erhalten bleibt.

Beispiel: $\texttt{ab} \between \texttt{cde} = \{\texttt{abcde}, \texttt{acbde}, \texttt{acdbe}, \texttt{acdeb}, \texttt{cabde}, \texttt{cadbe}, \texttt{cadeb}, \texttt{cdabe}, \texttt{cdaeb}, \texttt{cdeab}\}$

Dabei setzt sich z.B. `cadeb` wie folgt zusammen: $u_1 = \varepsilon$, $v_1 = \texttt{c}$, $u_2 = \texttt{a}$, $v_2 = \texttt{de}$, $u_3 = \texttt{b}$, $v_3 = \varepsilon$.

Das Shuffle-Produkt zweier Sprachen L_1, L_2 ist dann definiert als:

$$L_1 \between L_2 := \bigcup_{u \in L_1, v \in L_2} u \between v$$

Zeigen Sie: Wenn L_1 und L_2 regulär sind, dann ist auch $L_1 \between L_2$ regulär.

Hinweis: Erinnern Sie sich an den Kreuzproduktautomaten, der $L_1 \cap L_2$ erkennt. Passen Sie diesen geeignet an.

Lösung:
Für $i \in \{1,2\}$ sei $\mathcal{A}_i = (Q_i, \Sigma, \delta_i, s_i, F_i)$ ein DEA, der L_i erkennt. Dann wird $L_1 \between L_2$ durch den NEA $\mathcal{A}' = (Q_1 \times Q_2, \Sigma, \delta', (s_1, s_2), F_1 \times F_2)$ mit

$$\delta'((q_1, q_2), a) := \{(\delta_1(q_1, a), q_2), (q_1, \delta_2(q_2, a))\}$$

erkannt. Der NEA darf sich also für jedes Zeichen „aussuchen", ob es zu u oder v gehört. In dem entsprechenden Automatenteil wird dann ein Übergang genommen, während im anderen der Zustand beibehalten wird.

Wir beweisen, dass $\mathcal{A}'$ genau $L_1 \between L_2$ erkennt, indem wir induktiv zeigen, dass für jedes Wort $w \in \Sigma^*$ gilt:

$$\bar{\delta}'((s_1, s_2), w) = \{(\delta_1(q_1, u), \delta_2(q_2, v)) \mid u, v \in \Sigma^*, w \in u \between v\}$$

- **Induktionsanfang:** $\bar{\delta}'((s_1, s_2), \varepsilon) = \{(s_1, s_2)\} = \{(\delta_1(q_1, \varepsilon), \delta_2(q_2, \varepsilon))\}$. Die einzige Wahl von u und v, für die $\varepsilon \in u \between v$ gilt, ist $u = v = \varepsilon$.
- **Induktionsvoraussetzung:** Es gelte die zu beweisende Aussage für alle $w \in \Sigma^*$ mit $|w| \leq n$.
- **Induktionsschritt:** Beweise die Aussage für jedes wx mit $w \in \Sigma^*$, $x \in \Sigma$ und $|w| = n$.

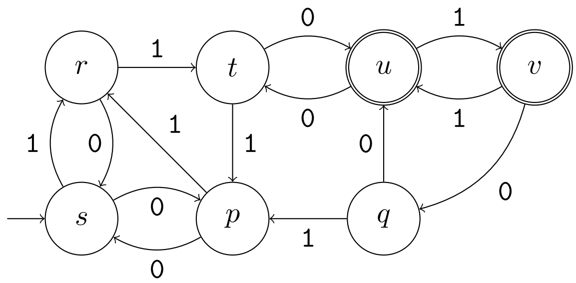
$$\begin{aligned}
\bar{\delta}'((s_1, s_2), wx) &= \bigcup_{(q_1', q_2') \in \bar{\delta}'((s_1, s_2), w)} \delta'((q_1', q_2'), x) \\
&= \{\delta'((\delta_1(q_1, u), \delta_2(q_2, v)), x) \mid u, v \in \Sigma^*, w \in u \between v\} \\
&= \{(\delta_1(\delta_1(q_1, u), x), \delta_2(q_2, v)), (\delta_1(q_1, u), \delta_2(\delta_2(q_2, v), x)) \mid u, v \in \Sigma^*, w \in u \between v\} \\
&= \{(\delta_1(q_1, ux), \delta_2(q_2, v)), (\delta_1(q_1, u), \delta_2(q_2, vx)) \mid u, v \in \Sigma^*, w \in u \between v\} \\
&= \{(\delta_1(q_1, u'), \delta_2(q_2, v')) \mid u', v' \in \Sigma^*, wx \in u' \between v'\}
\end{aligned}$$

Daraus folgt:

$$\begin{aligned}
\bar{\delta}'((s_1, s_2), w) \in F_1 \times F_2 &\Leftrightarrow \exists u, v \in \Sigma^* \colon w \in u \between v \wedge \delta_1(q_1, u) \in F_1 \wedge \delta_2(q_2, v) \in F_2 \\
&\Leftrightarrow \exists u, v \in \Sigma^* \colon w \in u \between v \wedge u \in L_1 \wedge v \in L_2 \\
&\Leftrightarrow w \in L_1 \between L_2
\end{aligned}$$

Aufgabe 3 (3 Punkte)

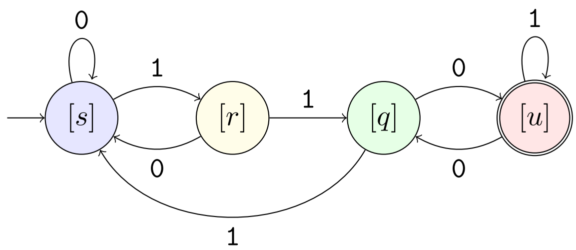
Konstruieren Sie zu folgendem Automaten den zugehörigen Äquivalenzklassenautomaten und zeichnen Sie den Übergangsgraphen.



Lösung:

Zeuge	Äquivalenzklassen
ε	$\{s,p,q,r,t\}$, $\{u,v\}$
`0`	$\{s,p,r\}$, $\{q,t\}$, $\{u,v\}$
`1`	$\{s,p,r\}$, $\{q,t\}$, $\{u,v\}$
`00`	$\{s,p,r\}$, $\{q,t\}$, $\{u,v\}$
`01`	$\{s,p,r\}$, $\{q,t\}$, $\{u,v\}$
`10`	$\{s,p\}$, $\{r\}$, $\{q,t\}$, $\{u,v\}$
`11`	$\{s,p\}$, $\{r\}$, $\{q,t\}$, $\{u,v\}$

Alle Wörter der Länge 3 trennen keine weiteren Äquivalenzklassen.



Aufgabe 4 (2 + 3 + 2 + 1 = 8 Punkte)

Ein Kommilitone von Ihnen behauptet, dass er ein alternatives Verfahren zur Konstruktion von Äquivalenzklassenautomaten gefunden hat. Statt nach möglicherweise langen Zeugen suchen zu müssen, betrachtet er immer nur einzelne Zeichen. Zu einem deterministischen endlichen Automaten $\mathcal{A} = (Q, \Sigma, \delta, s, F)$ geht er folgendermaßen vor. Im ersten Schritt partitioniert er die Zustandsmenge in zwei Mengen $Q \setminus F$ und F. In jedem weiteren Schritt wählt er zunächst ein Zeichen $a \in \Sigma$. Jede Menge[1] $[q]$ trennt er genau dann weiter auf, wenn für zwei Zustände $q_1, q_2 \in [q]$ nach dem vorherigen Schritt galt, dass $[\delta(q_1, a)] \neq [\delta(q_2, a)]$. Solch einen Schritt wiederholt er, bis sich bei keinem Zeichen $a \in \Sigma$ weitere Trennungen ergeben. Mit den entstandenen Mengen und dem Verfahren aus der Vorlesung konstruiert er dann den Äquivalenzklassenautomaten.

[1] Disjunkte Mengen $\{\ldots, q, \ldots\} = [q]$ können durch einen Repräsentanten q identifiziert werden.

(a) Führen Sie das Verfahren für den Automaten aus Aufgabe 3 durch, indem Sie die unten stehende Tabelle ausfüllen. Finden Sie dieselben Mengen wie die Äquivalenzklassen aus Aufgabe 3?

Lösung:

Schritt	Zeichen a	Partition nach Trennung durch a
1		$\{s,p,q,r,t\}$, $\{u,v\}$
2	`0`	$\{s,p,r\}$, $\{q,t\}$, $\{u,v\}$
3	`1`	$\{s,p\}$, $\{r\}$, $\{q,t\}$, $\{u,v\}$
4	`0`	$\{s,p\}$, $\{r\}$, $\{q,t\}$, $\{u,v\}$
5	`1`	$\{s,p\}$, $\{r\}$, $\{q,t\}$, $\{u,v\}$

(b) Folgern Sie aus $[q_1] \neq [q_2]$ induktiv die Existenz eines Zeugen w, der q_1 und q_2 trennt.

Lösung:
Induktionsanfang: Nach dem Anfangsschritt gibt es zwei Mengen $Q \setminus F$ und F. Diese werden durch den Zeugen ε getrennt.

Induktionsannahme: Nach dem i-ten Schritt folgt aus $[q_1] \neq [q_2]$ die Existenz eines Zeugen w, der q_1 und q_2 trennt.

Induktionsschluss: Betrachte jetzt den Schritt $i+1$ und zwei Zustände $q_1, q_2 \in Q$ mit $[q_1] \neq [q_2]$. Galt schon im Schritt i der Zusammenhang $[q_1] \neq [q_2]$, existiert nach Induktionsannahme ein Zeuge, der q_1 und q_2 trennt. Ansonsten galt im Schritt i dass $[q_1] = [q_2]$. Das bedeutet, dass a die Zustände q_1 und q_2 im Schritt $i+1$ getrennt hat. Das ist der Fall genau dann, wenn im Schritt i galt, dass $[\delta(q_1, a)] \neq [\delta(q_2, a)]$. Nach Induktionsannahme existiert dann ein Zeuge w, der $\delta(q_1, a)$ von $\delta(q_2, a)$ trennt. Dann ist aw ein Zeuge, der im Schritt $i+1$ die Zustände q_1 und q_2 trennt.

(c) Zeigen Sie, dass am Ende des Verfahrens $[q_1] = [q_2]$ impliziert, dass kein Zeuge existiert, der q_1 und q_2 trennt.

Lösung:
Sei am Ende des Verfahrens $q_1, q_2 \in Q$ mit $[q_1] = [q_2]$. Wähle ein beliebiges Wort $w = a_1 a_2 \cdots a_k$ mit $a_i \in \Sigma$ für $1 \leq i \leq k$. Es gilt $[\delta(q_1, a_1)] = [\delta(q_2, a_1)]$, denn sonst würden die Mengen $[q_1]$ und $[q_2]$ durch a_1 weiter aufgetrennt werden. Genauso gilt $[\delta(\delta(q_1), a_1), a_2)] = [\delta(\delta(q_2, a_1), a_2)]$, denn sonst würden die Mengen $[\delta(q_1, a_1)]$ und $[\delta(q_2, a_1)]$ durch a_2 weiter aufgetrennt werden. Es folgt $[\delta(\ldots, a_k)] = [\delta(\ldots, a_k)]$. Da schon im ersten Schritt akzeptierende von nicht akzeptierenden Zuständen getrennt wurden, kann w also q_1 und q_2 nicht trennen. Da w beliebig gewählt ist, kann also kein Zeuge existieren, der q_1 und q_2 trennt.

(d) Ist das Verfahren Ihres Kommilitonen korrekt?

Lösung:
Ja, denn aus (b) folgt, dass nicht-äquivalente Zustände in unterschiedlichen Mengen liegen, und aus (c) folgt, dass Zustände, die in derselben Menge liegen, äquivalent sind. Das Verfahren berechnet also genau die (eindeutigen) Äquivalenzklassen und ist ansonsten identisch zu dem Verfahren aus der Vorlesung.

Aufgabe 5

(2 + 2 + 2 + 2 = 8 Punkte)

Ist die Aussage des Pumping-Lemmas für folgende Sprachen erfüllt? Welche Sprachen sind regulär? Begründen Sie!

(a) $\Sigma = \{\mathtt{a}\}$, $L_a = \{\mathtt{a}^{\lfloor 10/j \rfloor !} \mid j \in \mathbb{N}\}$

(b) $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$, $L_b = \{\mathtt{a}^j \mathtt{b}^k \mathtt{c}^\ell \mid j, k, \ell \in \mathbb{N}_0, j < k < \ell\}$

(c) $\Sigma = \{\mathtt{0}, \mathtt{1}\}$, $L_c = \{y\mathtt{0}z \mid y, z \in \Sigma^*, |y| = |z|\}$

(d) $\Sigma = \{\mathtt{0}, \mathtt{1}\}$, $L_d = \{aza \mid a \in \Sigma, z \in \Sigma^*\}$

Lösung:

(a) Es gilt $L_a = \{a^{j!} \mid j \in \{1, 2, 3, 5, 10\}\}$. Also ist L_a ist endlich und somit regulär. Die Aussage des Pumping-Lemmas ist für alle Wörter der Länge $n > 10!$ erfüllt, da die zugrunde liegende Menge leer ist.

(b) Wir zeigen, dass die Aussage des Pumping-Lemmas nicht erfüllt ist und somit L_b nicht regulär sein kann.

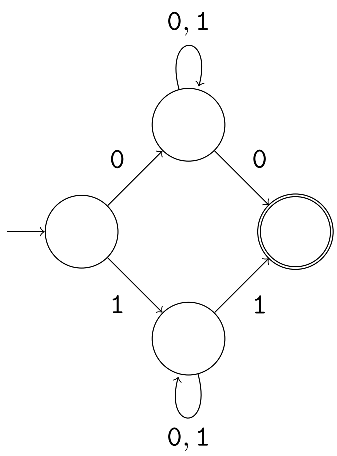
Beweis: Sei $n \in \mathbb{N}$ beliebig, aber fest. Wir wählen $w = \mathtt{a}^n \mathtt{b}^{n+1} \mathtt{c}^{n+2}$. Sei $w = uvx$ eine beliebige Zerlegung mit $|uv| \leq n$, $v \neq \varepsilon$. Dann gilt $v = \mathtt{a}^i$ für $1 \leq i \leq n$. Das Wort $uv^2x = \mathtt{a}^{n+i} \mathtt{b}^{n+1} \mathtt{c}^{n+2}$ liegt nicht in L_b, da $n + i \geq n + 1$. Somit gibt es für jedes $n \in \mathbb{N}$ ein Wort w in L_b, für das keine Zerlegung entsprechend dem Pumping-Lemma existiert.

(c) Wir zeigen, dass die Aussage des Pumping-Lemmas nicht erfüllt ist und somit L_c nicht regulär sein kann.

Beweis: Sei $n \in \mathbb{N}$ beliebig, aber fest. Wir wählen $w = \mathtt{1}^n \mathtt{0} \mathtt{1}^n$. Sei $w = uvx$ eine beliebige Zerlegung mit $|uv| \leq n$, $v \neq \varepsilon$. Dann gilt $v = \mathtt{1}^i$ für $1 \leq i \leq n$. Das Wort $uv^2x = \mathtt{1}^{n+i}\mathtt{0}\mathtt{1}^n$ liegt nicht in L_c, denn die einzig mögliche Wahl von y und z ist $y = \mathtt{1}^{n+1}$ und $z = \mathtt{1}^n$. Dann gilt aber $|y| \neq |z|$. Somit gibt es für jedes $n \in \mathbb{N}$ ein Wort w in L_c, für das keine Zerlegung entsprechend dem Pumping-Lemma existiert.

(d) Wir zeigen, dass die Aussage des Pumping-Lemmas für $n = 2$ erfüllt ist. Alle Wörter $w \in L_d$ mit $|w| > 2$ haben die Form $w = abza$ mit $a, b \in \Sigma$, $z \in \Sigma^*$. Wähle als Zerlegung $u = a$, $v = b$ und $x = za$. Offensichtlich gilt $|uv| \leq 2$ und $v \neq \varepsilon$. Für alle $i \in \mathbb{N}_0$ gilt: $uv^ix = ab^iza \in L_d$, da $b^iz \in \Sigma^*$.

Die Sprache L_d ist regulär, da sie von folgendem NEA erkannt wird:



Aufgabe 6 (3 + 2 = 5 Punkte)

Gegeben seien die folgenden Sprachen über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$:

- $L_1 = \{w \in \Sigma^* \mid |w|_\mathtt{a} = |w|_\mathtt{b}\}$
- $L_2 = \{w \in \Sigma^* \mid |w|_\mathtt{a} \mod 3 = |w|_\mathtt{b} \mod 3\}$

Dabei bezeichnet $|w|_x$ die Anzahl der Vorkommen des Zeichens x in w. In dieser Aufgabe sollen Sie die gegebenen Sprachen mithilfe der Nerode-Relation untersuchen.

(a) Geben Sie für L_1 und L_2 jeweils die Äquivalenzklassen der Nerode-Relation (in Mengenschreibweise) an. Geben Sie außerdem für jede Äquivalenzklasse K die Menge S_K der gültigen Suffixe an, also die Menge der $z \in \Sigma^*$, sodass für jedes $u \in K$ gilt: $uz \in L_1$ bzw. $uz \in L_2$.

(b) Ist L_1 bzw. L_2 regulär? Wenn ja, geben Sie einen DEA mit der minimalen Anzahl von Zuständen an.

Lösung:

(a) L_1 hat unendlich viele Äquivalenzklassen: Für jedes $i \in \mathbb{N}_0$ bildet $[\mathtt{a}^i] = \{w \in \Sigma^* \mid |w|_\mathtt{a} - |w|_\mathtt{b} = i\}$ sowie $[\mathtt{b}^i] = \{w \in \Sigma^* \mid |w|_\mathtt{b} - |w|_\mathtt{a} = i\}$ eine eigene Äquivalenzklasse. Dabei gilt $S_{[\mathtt{a}^i]} = [\mathtt{b}^i]$ und $S_{[\mathtt{b}^i]} = [\mathtt{a}^i]$.

Bei L_2 fallen für alle $i \in \mathbb{N}_0$ die Klassen $[\mathtt{a}^i]$ und $[\mathtt{b}^{3-i}]$ zusammen. Außerdem fallen für alle $j \in \mathbb{N}_0$ mit $(i - j) \mod 3 = 0$ die Klassen $[\mathtt{a}^i]$ und $[\mathtt{a}^j]$ zusammen. Es bleiben drei Äquivalenzklassen übrig:

- $[\varepsilon] = L_2 = \{w \in \Sigma^* \mid (|w|_\mathtt{a} - |w|_\mathtt{b}) \mod 3 = 0\}$
- $[\mathtt{a}] = \{w \in \Sigma^* \mid (|w|_\mathtt{a} - |w|_\mathtt{b}) \mod 3 = 1\}$
- $[\mathtt{aa}] = \{w \in \Sigma^* \mid (|w|_\mathtt{a} - |w|_\mathtt{b}) \mod 3 = 2\}$

Dabei gilt $S_{[\varepsilon]} = [\varepsilon]$, $S_{[\mathtt{a}]} = [\mathtt{aa}]$ und $S_{[\mathtt{aa}]} = [\mathtt{a}]$.

(b) L_1 ist nicht regulär, da es unendliche viele Äquivalenzklassen gibt. L_2 hat 3 Äquivalenzklassen und ist somit regulär. Der Minimalautomat sieht wie folgt aus: