Prof. Javier Esparza
Philipp Czerner, Martin Helfrich

Technische Universität München
Lehrstuhl für Theoretische Informatik

Einführung in die Theoretische Informatik

Sommersemester 2021 – Übungsblatt 3

- Das Übungsblatt ist in zwei Teile gegliedert: den Vorbereitungsteil, den Sie vor der Übung selbstständig bearbeiten sollen, und den Übungs-/Nachbereitungsteil, der Aufgaben enthält, die in der Übung besprochen werden und von Ihnen anschließend zur Nachbereitung verwendet werden können.
- Das ist nicht das Hausaufgabenblatt! Die Hausaufgaben finden Sie auf einem separaten Blatt.

Vorbereitung (→ vor der Übung selbständig zu bearbeiten)

Individualaufgabe Ü3.1. (*Wichtige Begriffe*)

Überprüfen Sie, dass Sie die folgenden Begriffe oder Notationen korrekt definieren können.

- Wortproblem
- Leerheitsproblem
- Endlichkeitsproblem
- Äquivalenzproblem
- Ardens Lemma
- Pumping Lemma

Individualaufgabe Ü3.2. (*Kahoot*)

Falls Sie das Kahoot aus der Vorlesung verpasst haben: Spielen Sie es jetzt! Link. [1]

Individualaufgabe Ü3.3. (*Automata Tutor: Pumping Lemma Game*)

Lösen Sie die Aufgaben Ü3.3 (a–b) auf Automata Tutor. [2]

Individualaufgabe Ü3.4. (*Strukturelle Induktion*)

Geben Sie eine rekursive Prozedur $empty(r)$ an, die für einen gegebenen regulären Ausdruck r entscheidet, ob $L(r) = \emptyset$. Für Ihre Definition sollten Sie das folgende Gerüst verwenden:

- $empty(\boldsymbol{\emptyset}) =$
- $empty(a) =$
- $empty(\boldsymbol{\epsilon}) =$
- $empty(\alpha\beta) =$
- $empty(\alpha \mid \beta) =$
- $empty(\alpha^*) =$

Beweisen Sie mittels struktureller Induktion, dass Ihre Definition korrekt ist.

Zu dieser Aufgabe gibt es eine Video-Lösung: rekursive Prozedur, strukturelle Induktion.

Lösungsskizze. Wir teilen den Beweis in zwei Teile auf.

Konstruktion.

- $empty(\boldsymbol{\emptyset}) = true$

[1] Falls der Link nicht mehr funktioniert, teilen Sie es bitte der Übungsleitung mit. Die Teilnehmeranzahl ist leider auf 2000 begrenzt.

[2] Wenn Sie Automata Tutor noch nicht verwendet haben, folgen Sie erst den Schritten in Ü1.2, um sich richtig zu registrieren.

- $\mathit{empty}(a) = \mathit{false}$
- $\mathit{empty}(\boldsymbol{\epsilon}) = \mathit{false}$
- $\mathit{empty}(\alpha\beta) = \mathit{empty}(\alpha) \lor \mathit{empty}(\beta)$
- $\mathit{empty}(\alpha \mid \beta) = \mathit{empty}(\alpha) \land \mathit{empty}(\beta)$
- $\mathit{empty}(\alpha^*) = \mathit{false}$

Korrektheit. Wir zeigen $L(r) = \emptyset \iff \mathit{empty}(r)$ mittels struktureller Induktion.

Fall $r = \boldsymbol{\emptyset}$, $r = a$, $r = \boldsymbol{\epsilon}$. Trivial.

Fall $r = \alpha^*$. Wir haben $\boldsymbol{\epsilon} \in L(\alpha^*) \neq \emptyset \iff \neg\mathit{empty}(\alpha^*)$. Die Aussage gilt per Definition von *empty*.

Fall $r = \alpha\beta$. Als Induktionshypothesen erhalten wir $L(\alpha) = \emptyset \iff \mathit{empty}(\alpha)$ und $L(\beta) = \emptyset \iff \mathit{empty}(\beta)$. Es gilt $L(\alpha\beta) = \emptyset \iff L(\alpha)L(\beta) = \emptyset \iff L(\alpha) = \emptyset \lor L(\beta) = \emptyset \overset{\text{I.H.}}{\iff} \mathit{empty}(\alpha) \lor \mathit{empty}(\beta) = \mathit{empty}(\alpha\beta)$.

Fall $r = \alpha \mid \beta$. Es gelten dieselben Induktionshypothesen wie im vorherigen Fall. Wir haben $L(\alpha \mid \beta) = \emptyset \iff L(\alpha) \cup L(\beta) = \emptyset \iff L(\alpha) = \emptyset \land L(\beta) = \emptyset \overset{\text{I.H.}}{\iff} \mathit{empty}(\alpha) \land \mathit{empty}(\beta) = \mathit{empty}(\alpha \mid \beta)$.

Übung und Nachbereitung

Fokusaufgabe Ü3.5. (*Pumping Lemma*)

Beweisen Sie für jede der folgenden Sprachen mithilfe des Pumping Lemmas, dass sie *nicht* regulär sind.

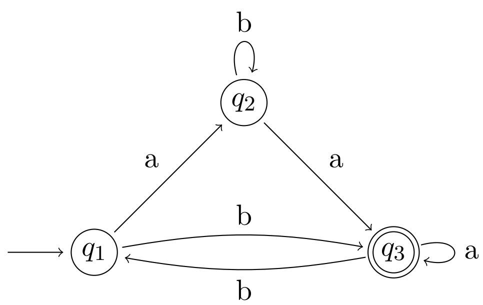
(a) $L_1 = \{w \in \{0,1\}^* \mid w = w^R\}$

(b) $L_2 = \{w \in \{0,1\}^* \mid |w|_0 \geq |w|_1\}$

(c) $L_5 = \{a^{2^i} \mid i \geq 0\}$

Übungsaufgabe Ü3.6. (*Ardens Lemma*)

Gegeben sei folgender Automat $M = (\{q_1, q_2, q_3\}, \{a, b\}, \delta, q_1, \{q_3\})$:



Berechnen Sie mit dem Gauß-Verfahren und Ardens Lemma einen regulären Ausdruck α mit $L(\alpha) = L(M)$.

Lösungsskizze. Gleichungssystem:

$$X_1 \equiv aX_2 \mid bX_3 \tag{1}$$
$$X_2 \equiv aX_3 \mid bX_2 \tag{2}$$
$$X_3 \equiv aX_3 \mid bX_1 \mid \boldsymbol{\epsilon} \tag{3}$$

Gleichung (2) nach X_2 auflösen und in (1) einsetzen:

$$X_2 \equiv b^*aX_3 \tag{4}$$
$$X_1 \equiv ab^*aX_3 \mid bX_3 \equiv (ab^*a \mid b)X_3 \tag{5}$$

Gleichung (5) in (3) einsetzen und auflösen:

$$X_3 \equiv aX_3 \mid b(ab^*a \mid b)X_3 \mid \boldsymbol{\epsilon} \tag{6}$$
$$\equiv (a \mid b(ab^*a \mid b))^* \tag{7}$$
$$\equiv (a \mid bb \mid bab^*a)^* \tag{8}$$

Einsetzen in (5):

$$X_1 \equiv \alpha \equiv (ab^*a \mid b)(a \mid bb \mid bab^*a)^* \tag{9}$$

Übungsaufgabe Ü3.7. (*Strukturelle Induktion: Klausuraufgabe von 2020*)

(a) Sei Σ ein Alphabet. Geben Sie die Rekursionsgleichungen für eine rekursive Prozedur contains(a, r) an, die für einen Buchstaben $a \in \Sigma$ und regulären Ausdruck r berechnet, ob a in jedem Wort aus $L(r)$ vorkommt.
Es soll also für alle $w \in L(r)$ gelten, dass $(\exists u, v \in \Sigma^* \,.\, w = uav) =: P(w)$.

(b) Beweisen Sie mit Hilfe von struktureller Induktion, dass ihre Prozedur korrekt ist. Sie dürfen dabei den Konkatenationsfall (d.h. $r = r_1 r_2$) weglassen. Kennzeichnen Sie dabei die Induktionshypothesen und deren Anwendungen deutlich.

Lösungsskizze.

(a)

$$\begin{aligned}
\text{contains}(a, \emptyset) &= \textbf{true}\\
\text{contains}(a, \epsilon) &= \textbf{false}\\
\text{contains}(a, x) &= (x = a)\\
\text{contains}(a, r_1|r_2) &= \text{contains}(a, r_1) \wedge \text{contains}(a, r_2)\\
\text{contains}(a, r_1 r_2) &= \text{contains}(a, r_1) \vee \text{contains}(a, r_2)\\
\text{contains}(a, r^*) &= \textbf{false}
\end{aligned}$$

(b)

- **Fall** $r = \emptyset$
  Es gilt contains$(a, \emptyset) = \textbf{true} \iff \forall w \in L(\emptyset).\ P(w)$, da $L(\emptyset) = \emptyset$.
- **Fall** $r = \epsilon$
  Wir haben $L(r) = \{\epsilon\}$ und contains$(a, \epsilon) = \textbf{false} \iff P(\epsilon) \iff \forall w \in L(\epsilon).\ P(w)$.

- **Fall** $r = x$
  Wenn $a = x$, dann gilt $L(x) = \{a\}$ und somit $\text{contains}(a, x) = \textbf{true} \iff P(a) \iff \forall w \in L(x).\ P(w)$. Ansonsten haben wir $\text{contains}(a, x) = \textbf{false} \iff P(x) \iff \forall w \in L(x).\ P(w)$, da $L(x) = \{x\}$.
- **Fall** $r = r_1|r_2$
  Als Induktionshypothesen erhalten wir $\text{contains}(a, r_1) \iff \forall w \in L(r_1).\ P(w)$ und $\text{contains}(a, r_2) \iff \forall w \in L(r_2).\ P(w)$. Damit haben wir

$$\begin{aligned}
\text{contains}(a, r_1|r_2) &\iff \text{contains}(a, r_1) \wedge \text{contains}(a, r_2) \\
&\overset{I.H.}{\iff} (\forall w \in L(r_1).\ P(w)) \wedge (\forall w \in L(r_2).\ P(w)) \\
&\iff \forall w \in L(r_1) \cup L(r_2).\ P(w) \\
&\iff \forall w \in L(r_1|r_2).\ P(w)
\end{aligned}$$

- **Fall** $r = r_1 r_2$ (optional)
  Als Induktionshypothesen erhalten wir $\text{contains}(a, r_1) \iff \forall w \in L(r_1).\ P(w)$ und $\text{contains}(a, r_2) \iff \forall w \in L(r_2).\ P(w)$. Wenn $L(r_1) = \emptyset$ oder $L(r_2) = \emptyset$, dann gilt $L(r_1 r_2) = \emptyset$ und die Korrektheit folgt analog zum Fall $r = \emptyset$. Ansonsten gilt

$$\begin{aligned}
\text{contains}(a, r_1 r_2) &\iff \text{contains}(a, r_1) \vee \text{contains}(a, r_2) \\
&\overset{I.H.}{\iff} (\forall w \in L(r_1).\ P(w)) \vee (\forall w \in L(r_2).\ P(w)) \\
&\iff (\forall w \in L(r_1 r_2).\ \exists y \in L(r_2).\ \exists u, v.\ w = uavy) \vee \\
&\qquad (\forall w \in L(r_1 r_2).\ \exists y \in L(r_1).\ \exists u, v.\ w = yuav) \\
&\iff (\forall w \in L(r_1 r_2).\ \exists u, v.\ w = uav) \vee \\
&\qquad (\forall w \in L(r_1 r_2).\ \exists u, v.\ w = uav) \\
&\iff \forall w \in L(r_1 r_2).\ \exists u, v.\ w = uav
\end{aligned}$$

  Dabei gilt die dritte Äquivalenz aufgrund der Annahmen $L(r_1) \neq \emptyset$ und $L(r_2) \neq \emptyset$.
- **Fall** $r = r_1^*$
  Dann gilt $\epsilon \in L(r_1^*)$ und da $P(\epsilon) = \textbf{false}$, ist die Definition $\text{contains}(a, r^*) = \textbf{false}$ korrekt.