Technische Universität München
Fakultät für Informatik
Prof. Tobias Nipkow, Ph.D.
Sascha Böhme, Lars Noschinski

Sommersemester 2011
Übungsblatt 3
30. Mai 2011

# Einführung in die Theoretische Informatik

*Abgabetermin: 30. Mai 2011* **vor der Vorlesung** *in die THEO-Briefkästen*

***Hinweis:*** *Bitte beachten Sie unbedingt die Hinweise zum Übungsablauf und zu den Aufgabentypen auf der THEO-Website (`http://theo.in.tum.de/`).*

## Hausaufgabe 1 (5 Punkte)

Wir betrachten einen Kaffeeautomaten mit folgender Spezifikation:

- Der Automat hat unendlich viel Wechselgeld und Zutaten zur Verfügung.
- Eingeworfen werden können 50-Cent- und 1-Euro-Münzen ($I_{50}$, $I_{100}$). Der Automat speichert ein Guthaben von höchstens zwei Euro. Wird durch den Einwurf einer weiteren Münze dieser Betrag überschritten, so wird diese Münze direkt zurückgegeben ($O_{50}$, $O_{100}$).
- Außer der Münzeingabe hat der Automat noch Knöpfe für Geldrückgabe ($G$) und einen extra großen Kaffee für $1,50$ EUR ($K$). Verlangt der Benutzer einen Kaffee und ist genug Guthaben im Automaten, so wird zunächst das gewünschte Getränk ausgegeben ($O_K$) und danach das Restguthaben ausgezahlt. Wurde nicht genug Geld eingeworfen, so passiert nichts.

  Bei einem Druck auf Geldrückgabe wird das eingeworfene Guthaben ausgezahlt.
- Die Auszahlung eines (Rest-)Guthabens erfolgt immer in 50-Cent-Münzen.

Modellieren Sie nun einen endlichen Automaten, der genau die Wörter akzeptiert, die eine gültige Interaktionssequenz mit dem Kaffeeautomaten darstellen und bei denen kein Guthaben im Automaten verbleibt.
Verwenden Sie dazu das Alphabet $\Sigma = \{I_{50}, I_{100}, O_{50}, O_{100}, G, K, O_K\}$.

## Hausaufgabe 2 (5 Punkte)

Die *Dropout-Wörter* zu einem Wort $w$ sind alle Wörter, die durch das Entfernen genau eines Buchstabens aus $w$ entstehen. Für eine Sprache $L$ nennen wir die Sprache $L_D$ die aus den Dropout-Wörtern der Wörter aus $L$ besteht, *Dropout-Sprache*. Formal schreiben wir: $L_D = \{uv \mid \exists a.\ uav \in L\}$.

1. Geben Sie je einen regulären Ausdruck für die Dropout-Sprachen zu $(ab)^*$ und $a((a|b)b)^*$ an.
2. Zeigen Sie, dass $L_D$ regulär ist für jede reguläre Sprache $L$.

## Hausaufgabe 3 (3 Punkte)

Wir bezeichnen die Menge er regulären Ausdrücke über einem Alphabet $\Sigma$ mit $\mathrm{RE}_\Sigma$.

1. Sei $a \in \Sigma$. Definieren Sie eine rekursive Funktion $contains_a : \mathrm{RE}_\Sigma \to \{\mathrm{true}, \mathrm{false}\}$, so dass gilt:

$$contains_a(\alpha) \iff L(\alpha) \neq \varnothing \wedge L(\alpha) \subseteq \{uav \mid u, v \in \Sigma^*\}$$

2. Definieren Sie zwei rekursive Funktionen

$$even : \mathrm{RE}_\Sigma \to \{\mathrm{true}, \mathrm{false}\} \quad \text{und} \quad odd : \mathrm{RE}_\Sigma \to \{\mathrm{true}, \mathrm{false}\}$$

so dass $even(\alpha) = \mathrm{true}$ gilt gdw. alle Wörter in $L(\alpha)$ gerade Länge haben und so dass $odd(\alpha) = \mathrm{true}$ gilt gdw. alle Wörter in $L(\alpha)$ ungerade Länge haben.

## Hausaufgabe 4 (3 Punkte)

Die Permutationssprache $L_P$ zu einer Sprache $L$ besteht aus allen Permutationen aller Wörter in $L$. Formal setzen wir:

$$L_P = \{u_{\pi(1)} \cdots u_{\pi(k)} \mid k \in \mathbb{N}, \pi \text{ Permutation und } u_1 \cdots u_k \in L\}$$

1. Geben Sie eine alternative, möglichst einfache Beschreibung der Permutationssprache zu der Sprache $L = L((ab)^*)$ an.
2. Verwenden Sie das Pumping-Lemma, um anhand dieser Sprache zu zeigen, dass reguläre Sprachen nicht abgeschlossen sind unter Permutation (d.h. für eine reguläre Sprache $L$ ist $L_P$ nicht zwangsläufig regulär).

## Hausaufgabe 5 (4 Punkte)

Widerlegen Sie die Regularität der folgenden Sprachen (mit Hilfe des Pumping-Lemmas):

1. $L_1 = \{w \in \{0,1\}^* \mid w^R = w\}$
2. $L_2 = \{w \in \{0,1\}^* \mid |w|_0 \geq |w|_1\}$

Dabei bedeutet $w^R$ die Umkehrung von $w$ (“rückwärtslesen”), und $|w|_a$ gibt die Anzahl der Zeichen $a$ in einem Wort $w$ an.

## Quiz 1

Beantworten Sie kurz die folgenden Fragen:

1. Wenn $L$ regulär ist und $L' \subseteq L$ gilt, ist dann auch $L'$ regulär?
2. Finden Sie ein Verfahren, welches für einen gegebenen NFA M entscheidet, ob $|L(M)| \leq 100$ gilt.
3. Ist die Sprache $L = \{a^n b^m \mid n+m \text{ gerade}\}$ regulär? Lässt sich das mit dem Pumping-Lemma zeigen?

## Tutoraufgabe 1

1. Sei $\Sigma = \{a, b, c, d, e\}$.
   Zeigen Sie, dass die Sprache $L = \{ab^{2i}cd^ie \mid i \in \mathbb{N}\}$ nicht regulär ist.
2. Sei $\Sigma = \{1\}$. Zeigen Sie, dass die Sprache $P = \{1^p \mid p \text{ prim}\}$ nicht regulär ist.

## Tutoraufgabe 2

Seien $A$, $B$ und $X$ Sprachen über $\Sigma$ mit $\epsilon \notin A$. Beweisen oder widerlegen Sie die Umkehrung von Arden's Lemma, d.h. $X = A^*B \Longrightarrow X = AX \cup B$.

## Tutoraufgabe 3

Wir betrachten den DFA $A = (Q, \Sigma, \delta, q_0, \{q_1, q_2\})$, dessen Zustandsübergangsfunktion $\delta$ gegeben ist durch

| $q_i$ | $\delta(q_i, 0)$ | $\delta(q_i, 1)$ |
|---|---|---|
| $q_0$ | $q_1$ | $q_2$ |
| $q_1$ | $q_0$ | $q_1$ |
| $q_2$ | $q_1$ | $q_0$ |

Stellen Sie zur Berechnung der Sprachen $L_i = \{w \in \Sigma^* \mid \hat{\delta}(q_i, w) \in \{q_1, q_2\}\}$ ein Gleichungssystem mit entsprechenden Unbekannten $X_i$ für die Sprachen $L_i$ auf und berechnen Sie mit Hilfe von Ardens Lemma alle $X_i$ als reguläre Ausdrücke.