

Formale Grundlagen der Informatik I

7. Übungsblatt



TECHNISCHE
UNIVERSITÄT
DARMSTADT

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

SS 2011
25.05.11

Minitest Lösung

- a) Das Wortproblem für kontextfreie Sprachen ist entscheidbar. Richtig Falsch
Begründung: Sei $L \subseteq \Sigma^*$ eine kontextfreie Sprache. Falls $\varepsilon \notin L$ hat L eine Grammatik in Chomsky-Normalform (Satz 3.3.2) und das Wortproblem kann mit dem CYK-Algorithmus (3.3.13) entschieden werden. Falls $\varepsilon \in L$, dann kann das Wortproblem für $L \setminus \{\varepsilon\}$, wie oben, mit dem CYK-Algorithmus entschieden werden und das Wortproblem für $\{\varepsilon\}$ ist trivial.
 Alternativ kann man auch einen PDA konstruieren, der die Sprache erkennt (Satz 4.1.5).

- b) Gegeben sei die Turingmaschine \mathcal{M} über dem Alphabet $\{a, b\}$ mit Startzustand q_0 und folgender Übergangsfunktion:

δ	\square	a	b
q_0	$(\square, >, q_a)$		
q_a	(\square, \circ, q^+)	$(a, >, q_b)$	(b, \circ, q^-)
q_b	(\square, \circ, q^-)	(a, \circ, q^-)	$(b, >, q_a)$

Hält \mathcal{M} bei jeder Eingabe? Und wenn ja, welche Sprache erkennt \mathcal{M} ?

- \mathcal{M} hält immer und erkennt $L((a + b)^*)$.
- \mathcal{M} hält immer und erkennt $L((ab)^*)$.
- \mathcal{M} hält nicht bei jeder Eingabe.

Begründung: Die DTM hält immer: In jedem Schritt, in dem \mathcal{M} noch nicht hält, wird der Kopf nach rechts bewegt. D.h. irgendwann muss \square eingelesen werden, weil die Eingabe zu Ende ist. Nun gilt, dass \mathcal{M} nur im ersten Schritt in q_0 ist, d.h. irgendwann wird \square in dem Zustand q_a oder q_b eingelesen und damit geht die DTM in q^+ oder q^- und hält (wenn sie nicht schon vorher gehalten hat).

Die Maschine entscheidet $L((ab)^*)$, denn anhand von δ sieht man, dass für ein a ohne folgendes b die DTM in den nicht-akzeptierenden Zustand q^- übergeht, wobei wegen $\delta(q_a, a) = (a, >, q_b)$ nach Lesen eines a stets in den Zustand q_b gewechselt wird, in dem (wie eben begründet) als nächstes Zeichen nur ein b erlaubt ist.

- c) Für das Komplement $\bar{H} = \Sigma^* \setminus H$ des Halteproblems gilt:
- \bar{H} ist entscheidbar
 - \bar{H} ist nicht aufzählbar
 - es gibt eine Typ 3 Grammatik G mit $L(G) = \bar{H}$

Begründung: Für die ersten beiden Punkte, siehe Skript S. 67. Gäbe es eine solche Typ 3 Grammatik, die \bar{H} akzeptiert, so könnte man sie in eine Typ 3 Grammatik überführen, die H akzeptiert. Damit wäre H entscheidbar, was im Widerspruch zum Beweis aus der Vorlesung steht.

Gruppenübung

Aufgabe G1 (Konstruktion von Turingmaschinen)

Beschreiben Sie formal (d.h. unter Angabe der Komponenten Σ, Q, δ) jeweils eine 1-Band Turingmaschine für die folgenden Funktionen:

- Bei Eingabe der Binärdarstellung einer Zahl $n \in \mathbb{N}$ soll die Binärdarstellung der Zahl $2 \cdot n$ berechnet werden.
- Bei Eingabe der Binärdarstellung einer Zahl $n \in \mathbb{N}$ soll die Binärdarstellung der Zahl $n+3$ berechnet werden. Ihre Turingmaschine soll dabei, neben den Zuständen q_0 und q^+ nur höchstens drei weitere Zustände enthalten.

Tipp: Nehmen Sie zur Vereinfachung an, dass Sie die Zahl n in der Form $b_0 b_1 \dots b_k$ (von links nach rechts geschrieben) auf dem Band vorliegen haben, wobei $n = \sum_{i=0}^k b_i \cdot 2^i$, $b_i \in \{0, 1\}$.

Aufgabe G2 (Simulation von Turingmaschinen)

Beschreiben Sie informell wie eine deterministische Turingmaschine \mathcal{M}_0 für jeden der nachfolgenden Punkte konstruiert werden kann. Geben Sie die dazu benötigten Zustände von \mathcal{M}_0 an und beschreiben Sie deren Funktion.

- Gegeben eine feste Zeichenkette $c = c_1 \dots c_k \in \Sigma^*$. Die Maschine \mathcal{M}_0 lösche die Eingabe, schreibe c auf das Eingabeband, fahre zurück an die Startposition und halte.
- Gegeben eine feste Turingmaschine \mathcal{M} und eine feste Zeichenkette $c \in \Sigma^*$. Die Maschine \mathcal{M}_0 lösche ihre Eingabe und rechne anschliessend weiter wie die Maschine \mathcal{M} bei Eingabe der Konstanten c .
- Gegeben eine feste Turingmaschine \mathcal{M} . Die Maschine \mathcal{M}_0 simuliere \mathcal{M} auf Eingabe $\langle \mathcal{M} \rangle$ so dass

$$\begin{aligned} w \xrightarrow{\mathcal{M}_0} \infty &\iff \langle \mathcal{M} \rangle \xrightarrow{\mathcal{M}} q^+, \\ w \xrightarrow{\mathcal{M}_0} q^- &\iff \langle \mathcal{M} \rangle \xrightarrow{\mathcal{M}} q^- \end{aligned}$$

für alle Eingaben $w \in \Sigma^*$ der Maschine \mathcal{M}_0 . Sie dürfen hier verwenden, dass für alle Turingmaschinen \mathcal{M} solch eine Kodierung $\langle \mathcal{M} \rangle$ existiert¹.

Aufgabe G3 (Chomsky-Hierarchie)

Sei $\Sigma = \{a, b, c\}$. Zu welchem Niveau der Chomsky-Hierarchie gehören die folgenden Sprachen?

- $$\begin{aligned} L_1 &= \{w \in \Sigma^* : \text{zu jedem } a \text{ kann man eine spätere Stelle mit einem } b \text{ finden derart, dass jedes } b \text{ zu höchstens einem } a \text{ gehört}\} \\ L_2 &= \{w \in \Sigma^* : \text{wenn in } w \text{ ein } a \text{ vorkommt, dann gibt es eine spätere Stelle, an der ein } b \text{ steht, wobei dieses } b \text{ zu mehreren } a\text{'s gehören kann}\} \end{aligned}$$

Hausübung

Aufgabe H1 (Chomsky-Hierarchie)

(8 Punkte)

Welche der folgenden Sprachen über dem Alphabet $\Sigma = \{a, b, c\}$ sind (i) regulär, (ii) kontextfrei, aber nicht regulär, oder (iii) nicht kontextfrei? Begründen Sie Ihre Antwort!

- $$\begin{aligned} L_1 &= \{x \in \Sigma^* : |x|_a > 2 \cdot |x|_b\} \\ L_2 &= \{x \in \Sigma^* : |x|_a > 2 \cdot |x|_b > 2 \cdot |x|_c\} \\ L_3 &= \{x \in \Sigma^* : |x|_a > 2 \cdot |x|_b \text{ und } |x|_b \leq 1337\} \\ L_4 &= \{x \in \Sigma^* : |x|_a > 2 \cdot |x|_b \text{ und } |x|_b \geq 1337\} \end{aligned}$$

¹ siehe auch: Gödelnummer (<http://de.wikipedia.org/wiki/Gödelnummer>)

Aufgabe H2 (Postsches Korrespondenzproblem)

Im Postschen Korrespondenzproblem ist eine endliche Folge von Wortpaaren $(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)$ mit $x_i, y_i \in \Sigma^+$ gegeben. Gefragt ist, ob es eine Folge von Indizes $i_1, i_2, \dots, i_n \in \{1, 2, \dots, k\}$ ($n \geq 1$) gibt, mit

$$x_{i_1} x_{i_2} \dots x_{i_n} = y_{i_1} y_{i_2} \dots y_{i_n}.$$

Wenn eine solche Folge existiert, heißt diese eine Lösung des Korrespondenzproblems $(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)$. Man kann zeigen, dass man nicht rekursiv entscheiden kann, ob eine Lösung existiert. Zeigen Sie, dass das folgende Korrespondenzproblem keine Lösungen hat.

i	x_i	y_i
1	10	101
2	011	11
3	101	011