Technische Universität Chemnitz Sommersemester 2002
Professur Theoretische Informatik
Prof. Dr. A. Goerdt, Dipl.-Inf. F. Schädlich, Dipl.-Inf. A. Lanka

8. Übung „Theoretische Informatik II“
Lösungsvorschläge

Aufgabe 1

$M = (\{z\}, \Sigma, \Gamma, \delta, z, S)$
$\Gamma = V \cup \Sigma$

Für jede Regel $A \rightarrow \alpha \in R \quad (\alpha \in V \cup \Sigma^*)$ wird dem Kellerautomaten die Regel $\delta(z, \varepsilon, A) \rightarrow (z, \alpha)$ hinzugefügt. Ferner ergänzt man für die Terminalsymbole aus $\Sigma: \ \delta(z, a, a) \rightarrow (z, \varepsilon)$.
Dadurch erhält man folgenden Kellerautomaten, der L_4 erkennt:

$\delta(z, \varepsilon, S) = \{(z, aSb), (z, aaSb), (z, ab), (z, aab)\}$
$\delta(z, a, a) = \{(z, \varepsilon)\}$
$\delta(z, b, b) = \{(z, \varepsilon)\}$

Aufgabe 2

gegeben: Kellerautomat $M = (Z, \Sigma, \Gamma, \delta, z_0, \#)$
gesucht: äquivalente Grammatik $G = (V, \Sigma, P, S)$, also $L(G) = L(M)$

Idee:
Variablen von G sind Tripel (p, A, q) mit $p, q \in Z$ (Zustände) und $A \in \Gamma$.

Zur besseren Übersicht wird $\begin{pmatrix} p \\ A \\ q \end{pmatrix}$ verwendet.

Das Tripel (p, A, q) steht für die Wörter, die abgeleitet werden können, wenn der Kellerautomaten von Zustand p (wobei A oberstes Kellerzeichen ist) in den Zustand q überführt, wird. Dabei wird A von der Kellerspitze entfernt.
Der Kellerautomat darf maximal 2 Zeichen auf den Keller schreiben. Dies kann leicht durch das Umformen des Automaten und durch Hinzufügen neuer Zustände erreicht werden. Bei dieser Aufgabe ist die Umformung nicht nötig.

$$V = \left\{ \begin{pmatrix} p \\ A \\ q \end{pmatrix} : p, q \in Z, A \in \Gamma \right\} \cup \{S\}$$

Offensichtlich wird bei jedem Übergang $z_0 \rightarrow^* z$ ($z \in Z$), in dem der Keller geleert wird, ein Wort der Sprache erkannt, d. h. aus allen Variablen $\begin{pmatrix} z_0 \\ \# \\ z \end{pmatrix}$ können Wörter der Sprache abgeleitet werden.
$\Rightarrow$ erste Regeln

0.

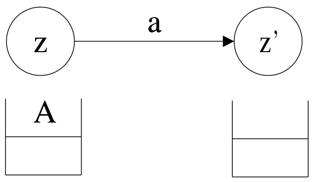
$$S \to \begin{pmatrix} z_1 \\ \# \\ z \end{pmatrix} \qquad \forall z \in Z.$$

Es werden nun drei Fälle unterschieden: 0/1/2 Symbole auf den Keller schreiben.

1.

$$\sigma(z, a, A) = \{(z', \varepsilon)\} \Rightarrow \begin{pmatrix} z \\ A \\ z' \end{pmatrix} \to a$$

0 Symbole schreiben (d. h. Kellerspitze löschen) für alle $z' \in Z$

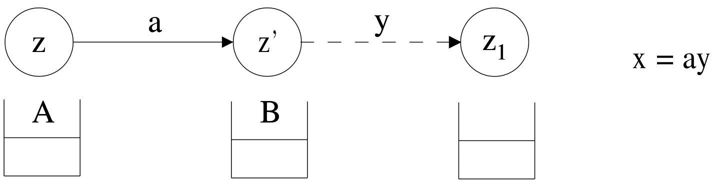


Anwenden auf die Regeln (3), (4), (5), (6) und (7).

2.

$$\sigma(z, a, A) = \{(z', B)\} \Rightarrow \begin{pmatrix} z \\ A \\ z_1 \end{pmatrix} \to a \begin{pmatrix} z' \\ B \\ z_1 \end{pmatrix}$$

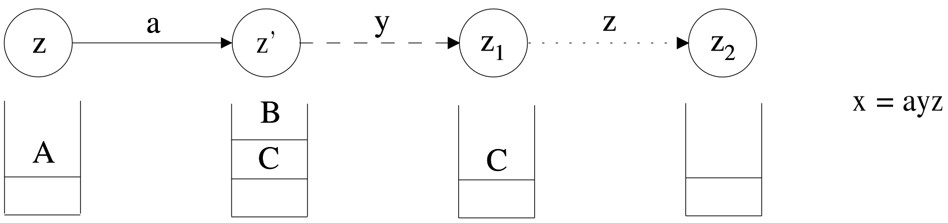
1 Symbol schreiben für alle $z_1 \in Z$



Anwenden auf Regel (1)3.

$$\sigma(z, a, A) = \{(z', BC)\} \Rightarrow \begin{pmatrix} z \\ A \\ z_2 \end{pmatrix} \to a \begin{pmatrix} z' \\ B \\ z_1 \end{pmatrix} \begin{pmatrix} z_1 \\ C \\ z_2 \end{pmatrix}$$

2 Symbole schreiben für alle $z_1, z_2 \in Z$



Anwenden auf Regel (2)

$$S \to \begin{pmatrix} z_0 \\ \# \\ z_0 \end{pmatrix} \;\Big|\; \begin{pmatrix} z_0 \\ \# \\ z_1 \end{pmatrix}$$

$$\begin{pmatrix} z_0 \\ \# \\ z_0 \end{pmatrix} \to \varepsilon \;\Big|\; a \begin{pmatrix} z_0 \\ a \\ z_0 \end{pmatrix}, \quad \begin{pmatrix} z_1 \\ \varepsilon \\ z_1 \end{pmatrix} \to \varepsilon$$

$$\begin{pmatrix} z_0 \\ \# \\ z_1 \end{pmatrix} \to a \begin{pmatrix} z_0 \\ a \\ z_1 \end{pmatrix}$$

$$\begin{pmatrix} z_0 \\ a \\ z_0 \end{pmatrix} \to a \begin{pmatrix} z_0 \\ a \\ z_0 \end{pmatrix} \begin{pmatrix} z_0 \\ a \\ z_0 \end{pmatrix} \Bigg| \; a \begin{pmatrix} z_0 \\ a \\ z_1 \end{pmatrix} \begin{pmatrix} z_1 \\ a \\ z_0 \end{pmatrix}$$

$$\begin{pmatrix} z_0 \\ a \\ z_1 \end{pmatrix} \to a \begin{pmatrix} z_0 \\ a \\ z_0 \end{pmatrix} \begin{pmatrix} z_0 \\ a \\ z_1 \end{pmatrix} \Bigg| \; a \begin{pmatrix} z_0 \\ a \\ z_1 \end{pmatrix} \begin{pmatrix} z_1 \\ a \\ z_1 \end{pmatrix}$$

$$\begin{pmatrix} z_0 \\ a \\ z_1 \end{pmatrix} \to b, \quad \begin{pmatrix} z_1 \\ a \\ z_1 \end{pmatrix} \to b, \quad \begin{pmatrix} z_0 \\ a \\ z_0 \end{pmatrix} \to \varepsilon$$

$$\begin{pmatrix} z_1 \\ a \\ z_1 \end{pmatrix}, \quad \begin{pmatrix} z_1 \\ \# \\ z_0 \end{pmatrix} \quad \text{können nach keinem Wort abgeleitet werden.}$$

5. Schritt (Umwandlung der Vektoren in Buchstaben):

$S \to A \mid B$
$A \to \varepsilon \mid aC$
$B \to aD$
$C \to \varepsilon \mid aCC \mid aDG$
$D \to aCD \mid aDH \mid b$
$H \to \varepsilon \mid b$
6. Schritt (Entfernen der ε-Übergänge):

$S \to \varepsilon \mid A \mid B \mid a \mid aC$
$A \to aC$
$B \to aD$
$C \to aCC \mid aDG \mid aC \mid a$
$D \to b \mid aCD \mid aD \mid aDb$

Aufgabe 3

$\Sigma = \{0, 1\}$
gegeben: eine positive Binärzahl
gesucht: die um 1 erniedrigte Binärzahl

Algorithmus:
Von rechts her alle Ziffern negieren, bis man auf eine „1“ trifft.

Zustände:
z_0 - Startzustand und zur rechten Ziffer laufen
z_1 - die aktuelle Stelle ändern und dann weiter nach links gehen
z_2 - die aktuelle Stelle nicht ändern und nur weiter nach links gehen
z_3 - die eventuell auftretende führende Null beseitigen
z_E - Endzustand

δ	0	1	$\Box$
z_0	$(z_0, 0, R)$	$(z_0, 1, R)$	$(z_1, \Box, L)$
z_1	$(z_1, 1, L)$	$(z_2, 0, L)$	$(z_1, \Box, N)$
z_2	$(z_2, 0, L)$	$(z_2, 1, L)$	$(z_3, \Box, R)$
z_3	$(z_E, \Box, R)$	$(z_E, 1, N)$	$(z_E, \Box, N)$

Aufgabe 4

Untersuchen wir zunächst das Verhalten der Turing-Maschine M auf den möglichen Eingaben der Länge 50. M hat für jede Eingabe eine Laufzeit $t_M(50) \leq 25$. Daraus können wir unmittelbar schlußfolgern, daß M nur die ersten 25 Zeichen der Eingabe lesen kann. Um weitere Zeichen zu lesen, bräuchte sie mindestens 26 Schritte.

Nehmen wir uns eine beliebige Eingabe der Länge 50 her. M rechnet wie gesagt nur mit den ersten 25 Zeichen dieser Eingabe. Daran ändert sich auch nichts, wenn wir am Ende der Eingabe zusätzliche Zeichen anhängen. Die Turingmaschine kann dies gar nicht bemerken und rechnet genauso, als hätten wir die Eingabe nicht verlängert. Die Laufzeit vom M ist daher unabhängig von der Länge der Eingabe immer ≤ 25. Wir haben also eine Konstante $C = 25$ gefunden.