

## 8.2

Sei  $a = (Q, \Sigma, \Gamma, q_0, \delta, F)$  eine Turingmaschine mit  $Q = \{q_0, q_1, q_2, q_3, q_3', q_4, q_5\}$ ,  $\Sigma = \{a, b, c\}$ ,  $\Gamma = \{a, b, c, a', b', c', \square\}$  und  $F = \{q_4\}$ . Sei zudem  $\delta$  bestimmt durch:

$$\begin{aligned}
\delta(q_0, \square) &= (\square, r, q_0) \mid \text{ignoreiere Lücken auf der Suche nach } a \\
\delta(q_0, a) &= (a', r, q_1) \mid a \text{ wurde gefunden, suche ein } b \\
\delta(q_0, a') &= (a', r, q_0) \mid \text{suche das nächste } a \\
\delta(q_0, b') &= (b', l, q_3') \mid \text{alle } a \text{ wurden verarbeitet, teste ob das Wort korrekt ist} \\
\delta(q_1, a) &= (a, r, q_1) \mid \text{suche das nächste } b \\
\delta(q_1, a') &= (a', r, q_1) \mid \text{suche das nächste } b \\
\delta(q_1, b) &= (b', r, q_2) \mid b \text{ wurde gefunden, suche ein } c \\
\delta(q_2, b) &= (b, r, q_2) \mid \text{suche das nächste } c \\
\delta(q_2, b') &= (b', r, q_2) \mid \text{suche das nächste } c \\
\delta(q_2, c) &= (c', r, q_2) \mid c \text{ wurde gefunden, gehe wieder an den Anfang} \\
\forall X \in \Gamma \setminus \square : \delta(q_3, X) &= (X, l, q_3) \mid \text{gehe an den Anfang} \\
\delta(q_3, \square) &= (\square, r, q_0) \mid \text{Anfang des Wortes gefunden, suche das nächste } a \\
\forall X \in \Gamma \setminus \square : \delta(q_3', X) &= (X, l, q_3') \mid \text{gehe an den Anfang} \\
\delta(q_3', \square) &= (\square, r, q_4) \mid \text{Anfang des Wortes gefunden, überprüfe die Verarbeitung} \\
\forall X \in \Gamma \setminus (\square \cup \Sigma) : \delta(q_4, X) &= (X, r, q_4) \mid \text{überprüfe ob nur verarbeitete Symbole auftauchen} \\
\delta(q_4, \square) &= (\square, r, q_5) \mid \text{Ende des Wortes und kein Eingabesymbol wurde gefunden}
\end{aligned}$$

so gilt  $L(a) = \{a^n b^n c^n \mid n \in \mathbb{N}\}$

## 8.3

Sei  $\mathfrak{A}$  wie in der Aufgabenstellung gegeben, dann kann man die passende Grammatik  $G = (\{S, [q_0 Z_0 q_0], [q_0 X q_0], [q_0 X q_1]\}, \{0, 1\}, P, S)$  konstruieren. Dabei sei  $P$

wie folgt definiert:

$$\begin{aligned}
S &\rightarrow [q_0 Z_0 q_0] [q_0 Z_0 q_1] \\
[q_0 Z_0 q_0] &\rightarrow 0 [q_0 X q_0] [q_0 X q_1] \\
[q_0 X q_0] &\rightarrow 0 [q_0 X q_0] [q_0 X q_1] \\
[q_0 X q_1] &\rightarrow 1
\end{aligned}$$

## 8.4

### Behauptung

Der Schnitt einer kontextfreien Sprache mit einer regulären ist kontextfrei.

### Beweis

Sei  $L_1$  eine kontextfreie Sprache, so existiert ein Kellerautomat  $a_1$ , sodass  $L(a_1) = L_1$  gilt. Sei  $a_1 = (Q_1, \Sigma, \Gamma, q_{0_1}, Z_0, \Delta_1, F_1)$

Sei  $L_2$  eine reguläre Sprache, so existiert ein NEA  $a_{2'}$ , sodass  $L(a_{2'}) = L_2$  gilt. Da man nach Satz 1.19 zu jedem NEA  $a_{2'}$  einen DEA  $a_2$  finden kann, sodass gilt  $a_{2'} = a_2 \Rightarrow L(a_2) = L_2$ . Sei  $a_2 = (Q_2, \Sigma, q_{0_2}, \Delta_2, F_2)$

Nach Bemerkung 1.9 ist der Schnitt zweier NEAs der Schnitt beider Sprachen, demnach gilt es zu zeigen, dass  $L(a_1 \times a_2)$  kontextfrei ist. Nun lässt sich ein Produktautomaten  $a_3$  so konstruieren, dass gilt:  $a_1 \times a_2 = a_3 = (Q_1 \times Q_2, \Sigma, \Gamma, (q_{0_1} \times q_{0_2}), Z_0, \Delta, (F_1 \times F_2))$ , wobei  $\Delta$  definiert, sodass gilt

$$\forall ((q_1, a, \beta), (p_1, \gamma)) \in \Delta_1 \wedge (q_2, a, p_2) \in \Delta_2 : (((q_1, q_2), a, \beta), ((p_1, p_2), \gamma)) \in \Delta$$

und

$$\forall ((q_1, \epsilon, \beta), (p_1, \gamma)) \in \Delta_1 \wedge (q_2, \delta, q_2) \in \Delta_2 : (((q_1, q_2), \epsilon, \beta), ((p_1, q_2), \gamma)) \in \Delta$$

Damit ist  $a_3$  ein Kellerautomat, also ist  $L(a_3)$  kontextfrei, dadurch gilt die Behauptung.  $\square$