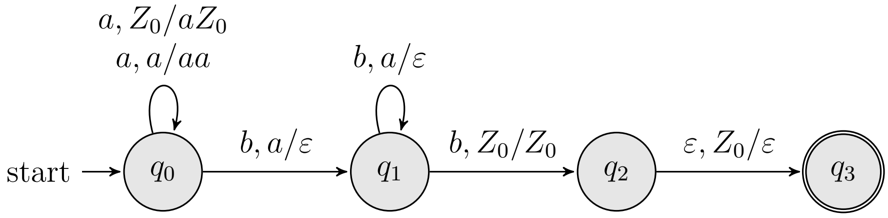
Constantin Lazari, Marco Wettstein 19. März 2013

1. Entwerfen deterministische PDA's die folgende Sprachen erkennen (akzeptieren durch Endzustand).

   (a) $\{a^n b^n | n > 0\}$

   Lösung:

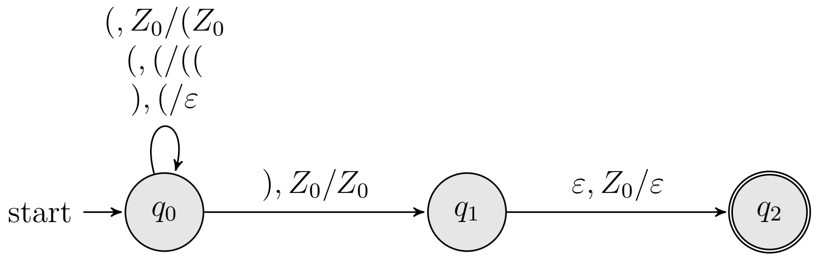
   $$P = (\{q_0, q_1, q_2\}, \{a, b\}, \{a, Z_0\}, \{(q_0, w, Z_0) \vdash^* (q_3, \varepsilon, \varepsilon)\}, q_0, Z_0, q_3)$$

   (b) Die Menge der wohlgeformten Klammerausdrücke (alle aufgehenden Klammern schliessen und es Schliessen nur Klammern, die vorhar aufgegangen sind.

   Lösung:

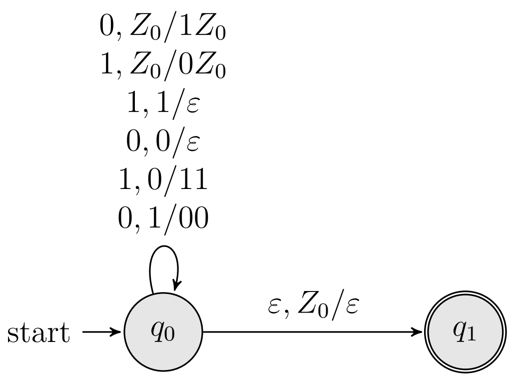
   $$P = (\{q_0, q_1\}, \{(,)\}, \{(, Z_0\}, \{(q_0, \omega, Z_0) \vdash^* (q_2, \varepsilon, \varepsilon)\}, q_0, Z_0, q_2)$$

   (c) Die Menge aller Wörter $w \in \{0, 1\}^*$ die die gleiche Anzahl Einsen und Nullen enthalten.

   Lösung:

   $$P = (\{q_0, q_1\}, \{0, 1\}, \{0, 1, Z_0\}, \{(q_0, \omega, Z_0) \vdash^* (q_1, \varepsilon, \varepsilon)\}, q_0, Z_0, q_1)$$

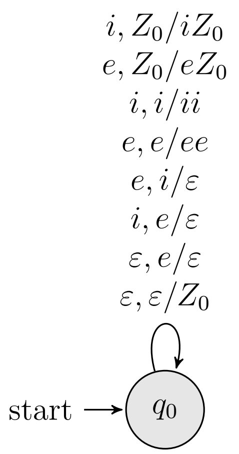
   

2. Gesucht sei ein PDA $P_{\text{if-else}}$, der eine Eingabefolge bestehend aus „if" und „else" genau dann akzeptiert (durch leeren Stack), wenn sie ein Präfix enthält, das mehr „else" als „if" aufweist. Dieses entspricht einer Syntaxverletzung in einer typischen Programmiersprache. Zur Vereinfachung repräsentiert das Symbol „i" ein „if" und das Symbol „e" ein „else".

   (a) Entwerfen Sie den PDA $P_{\text{if-else}}$ (Formale Notation und graphische Darstellung angeben)

Lösung:

$$\begin{aligned}
P_{\text{if-else}} &= (\{q_0, q_1\}, \{i, e\}, \{i, Z_0\}, \delta, q_0, Z_0) \\
\delta(q_0, i, Z_0) &= \{q_0, iZ_0\} \\
\delta(q_0, i, i) &= \{q_0, ii\} \\
\delta(q_0, i, e) &= \{q_0, \varepsilon\} \\
\delta(q_0, e, Z_0) &= \{q_0, eZ_0\} \\
\delta(q_0, e, i) &= \{q_0, \varepsilon\} \\
\delta(q_0, e, e) &= \{q_0, ee\} \\
\delta(q_0, \varepsilon, e) &= \{q_0, Z_0\} \\
\delta(q_0, \varepsilon, \varepsilon) &= \{q_0, Z_0\}
\end{aligned}$$



(b) Zeigen Sie, dass $P_{\text{if-else}}$ die Zeichenfolge $e, ieiee, iee, ieiiee$ und $iieee$ akzeptiert oder nicht akzeptiert.

i. e

Lösung:
$(q_0, e, Z_0) \vdash (q_0, \varepsilon, eZ0) \vdash (q_0, \varepsilon, Z_0) \Rightarrow$ Stack leer $\Rightarrow$ akzeptiert

ii. $ieiee$

Lösung:

$$(q_0, ieiee, Z_0) \vdash (q_0, eiee, iZ_0) \vdash (q_0, iee, Z_0) \vdash (q_0, ee, iZ_0) \vdash (q_0, e, Z_0) \\ \vdash (q_0, \varepsilon, eZ_0) \vdash (q_0, \varepsilon, Z_0)$$

Stack leer, also akzeptiert.

iii. iee

Lösung:

$$(q_0, iee, Z_0) \vdash (q_0, ee, iZ_0) \vdash (q_0, e, Z_0) \vdash (q_0, \varepsilon, eZ_0) \vdash (q_0, \varepsilon, Z_0)$$

Stack leer, also akzeptiert.

iv. $ieiiee$

Lösung:

$$(q_0, ieiiee, Z_0) \vdash (q_0, eiiee, iZ_0) \vdash (q_0, iiee, Z_0) \vdash (q_0, iee, iZ_0) \\ \vdash (q_0, ee, iiZ_0) \vdash (q_0, e, iZ_0) \vdash (q_0, \varepsilon, Z_0)$$

Kein Übergang definiert und Stack ist nicht leer, also nicht akzeptiert.

v. $iieee$

Lösung:

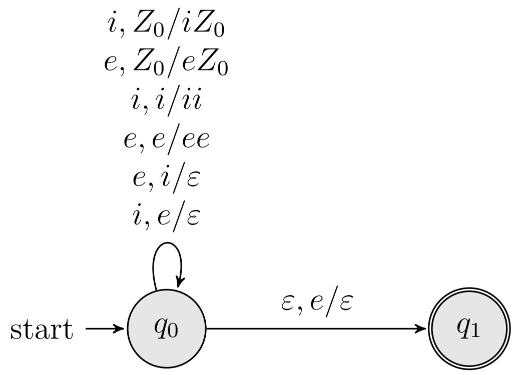
$$(q_0, iieee, Z_0) \vdash (q_0, ieee, iZ_0) \vdash (q_0, eee, iiZ_0) \vdash (q_0, ee, iZ_0) \\ \vdash (q_0, e, Z_0) \vdash (q_0, \varepsilon, eZ_0) \vdash (q_0, \varepsilon, Z_0)$$

Stack leer, also akzeptiert.

(c) Konstruieren Sie einen PDA $P_{\text{if-else-Z}}$, der dieselbe Sprache über einen akzeptierten Zustand erkennt.

Lösung:

$$
\begin{aligned}
P_{\text{if-else}} &= (\{q_0, q_1\}, \{i, e\}, \{i, Z_0\}, \delta, q_1, Z_0)\\
\delta(q_0, i, Z_0) &= \{q_0, iZ_0\}\\
\delta(q_0, i, i) &= \{q_0, ii\}\\
\delta(q_0, i, e) &= \{q_0, \varepsilon\}\\
\delta(q_0, e, Z_0) &= \{q_0, eZ_0\}\\
\delta(q_0, e, i) &= \{q_0, \varepsilon\}\\
\delta(q_0, e, e) &= \{q_0, ee\}\\
\delta(q_0, \varepsilon, e) &= \{q_1, \varepsilon\}
\end{aligned}
$$



(d) Zeigen Sie, dass $P_{\text{if-else-Z}}$ die Zeichenfolgen e, iee und $ieiiee$ akzeptiert oder nicht akzeptiert.

i. e

Lösung:
$(q_0, e, Z_0) \vdash (q_0, \varepsilon, eZ_0) \vdash (q_1, \varepsilon, Z_0)$ akzeptierter finaler Zustand

ii. iee

Lösung:

$$(q_0, iee, Z_0) \vdash (q_0, ee, iZ_0) \vdash (q_0, e, Z_0) \vdash (q_0, \varepsilon, eZ_0) \vdash (q_1, \varepsilon, Z_0)$$

akzeptierter finaler Zustand

iii. $ieiiee$

Lösung:

$(q_0, ieiiee, Z_0) \vdash (q_0, eiiee, iZ_0) \vdash (q_0, iiee, Z_0) \vdash (q_0, iee, iZ_0)$
$\vdash (q_0, ee, iiZ_0) \vdash (q_0, e, iZ_0) \vdash (q_0, \varepsilon, Z_0)$ kein Übergang, nicht akzeptiert

3. Eine KfG G mit Startsymbol S ist in Chomsky-Normalform, wenn alle ihre Produktionen von einer der folgenden Gestalt sind:

   - $A \to BC$, wobei B und C Variablen von G sind.
   - $A \to a$, wobei a ein Terminalsymbol von G ist.
   - $S \to \epsilon$, wenn diese Produktion vorkommt, wird ferner verlangt, dass die Variable S nie auf der rechten Seite einer Produktion vorkommen darf.

   Geben Sie eine KfG G an, die in Chomsky-Normalform ist und als Sprache genau die Palindrome über $\{x, y\}^*$ hat.

Lösung:

$$G = (V, T, P, S) = (\{S, A, B, C, X, Y\}, \{x, y\}, P, S)$$

mit den Produktionen P:

$$\begin{aligned}
S &\to \epsilon | BX | CY \\
A &\to BX | CY | x | y \\
B &\to XB | x \\
C &\to YA | y \\
X &\to x \\
Y &\to y
\end{aligned}$$