

Aufgabe 24

Es soll eine kontextfreie Grammatik konstruiert werden, die die Sprache

$$L \stackrel{\text{def}}{=} \{wcx \in \{a, b, c\}^* \mid w, x \in \{a, b\} \wedge w \neq x\}$$

akzeptiert.

Für das Wort $v = w_1w_2 \dots w_ncx_1x_2 \dots x_m \in L$ gilt:

1. w und x haben unterschiedliche Länge, also $|w| \neq |x|, n \neq m$ oder
2. $|w| = |x|$ und es existiert mindestens ein $i : 1 \leq i \leq n$ und $x_i \neq w_i$

Wir konstruieren eine Grammatik G , so daß für $v \in \{a, b, c\}^*$ mit $S \xrightarrow{*}_G v$ entweder (1) oder (2) gilt.

Zu (1): Sei zunächst $|w| > |x|$.

$$\begin{aligned} S &\rightarrow TS_L \\ S_L &\rightarrow TS_LT \mid TS_Lc \\ T &\rightarrow a \mid b \end{aligned}$$

Dann gilt: $S \Rightarrow TS_L \xRightarrow{*} Tw'cx \Rightarrow wcx$ mit $|w| > |x|$.

Analog: $|w| < |x|$.

$$\begin{aligned} S &\rightarrow S_RT \\ S_R &\rightarrow TS_RT \mid S_RTc \\ T &\rightarrow a \mid b \end{aligned}$$

Dann gilt: $S \Rightarrow S_RT \xRightarrow{*} wcx'T \Rightarrow wcx$ mit $|w| < |x|$.

Zu (2): $w_i = a, x_i = b$

$$\begin{aligned} S &\rightarrow S_abT' \\ S_a &\rightarrow TS_aT \mid aT'c \\ T' &\rightarrow T'T' \mid T \mid \varepsilon \\ T &\rightarrow a \mid b \end{aligned}$$

Dann gilt: $S_a \xRightarrow{*} T^n S_a T^n \Rightarrow T^n a T' c T^n$, d.h. vor einem a stehen genauso viele Terminalsymbole wie nach einem c .

Also:

$$\begin{aligned} S &\Rightarrow S_abT' \\ &\xRightarrow{i-1} T^{i-1} S_a T^{i-1} b T' \\ &\Rightarrow T^{i-1} a T' c T^{i-1} b T' \\ &\xRightarrow{*} w_1 w_2 \dots w_{i-1} a w_{i+1} \dots w_n c x_1 x_2 \dots x_{i-1} b x_{i+1} \dots x_m \\ &\rightsquigarrow w_i = a, x_i = b \end{aligned}$$

Analog: $w_i = b, x_i = a$

$$\begin{aligned} S &\rightarrow S_b|aT' \\ S_b &\rightarrow TS_bT|bT'c \\ T' &\rightarrow T'T'|T|\varepsilon \\ T &\rightarrow a|b \end{aligned}$$

Damit:

$$\begin{aligned} S &\Rightarrow S_baT' \\ &\stackrel{i-1}{\Rightarrow} T^{i-1}S_bT^{i-1}aT' \\ &\Rightarrow T^{i-1}bT'cT^{i-1}aT' \\ &\stackrel{*}{\Rightarrow} w_1w_2\dots w_{i-1}bw_{i+1}\dots w_ncx_1x_2\dots x_{i-1}ax_{i+1}\dots x_m \\ &\rightsquigarrow w_i = b, x_i = a \end{aligned}$$

Zusammenfassung:

$$G = (\{S, S_L, S_R, S_a, S_b, T', T, a, b, c\}, \{a, b, c\}, P, S)$$

$$P = \left\{ \begin{array}{l} S \rightarrow TS_L|S_RT|S_abT'|S_baT' \\ S_L \rightarrow TS_LT|TS_L|c \\ S_R \rightarrow TS_RT|S_RT|c \\ S_a \rightarrow TS_aT|aT'c \\ S_b \rightarrow TS_bT|bT'c \\ T' \rightarrow T'T'|T|\varepsilon \\ T \rightarrow a|b \end{array} \right\}$$

Aufgabe 25

Es soll ein PDA konstruiert werden, der $L_n \stackrel{\text{def}}{=} \{wcx \in \{a, b, c\}^* | w, x \in \{a, b\}^n \wedge w \neq x\}$ akzeptiert.

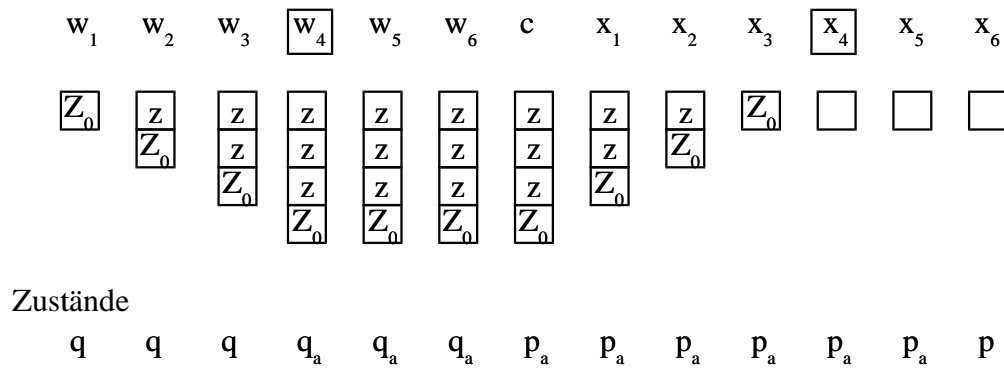
Jedes Wort besteht aus zwei gleich langen Teilen w und x , die durch ein c getrennt sind. w und x unterscheiden sich an mindestens einer Stelle.

Idee:

- Zähle die Länge $|w| = |x| = n$ über die Zustände
- Merke mit Hilfe des Kellers, wo sich w und x unterscheiden.
 - Stack aufbauen
 - „Entscheidung“: i -te Stelle $w_i = a$ und $x_i = b$ oder $w_i = b$ und $x_i = a$
 - Stack unberührt lassen bis c kommt
 - Stack abbauen bis zum Startsymbol

- Eingabesymbol $x_i = b$ (bzw a): Rest abzählen und Startsymbol beseitigen
- $X_i = a$ (bzw b): nicht akzeptieren

w und x unterscheiden sich bei $i = 4$



Definition: $M = (Q, \{a, b, c\}, \{z, Z_0\}, \delta, q_0, Z_0, \emptyset)$ mit

$Q = \{q_0, q_1, \dots, q_{n-1}, q_1^a, q_2^a, \dots, q_n^a, q_1^b, q_2^b, \dots, q_{n+1}^b, p_0, p_1, \dots, p_n, p_0^a, \dots, p_{n-1}^a, p_0^b, \dots, p_{n-1}^b\}$.

Offenbar: $|Q| = O(n)$.

Übergangsfunktion δ ist in der Bibliothek erhältlich.

Aufgabe 26

a) Um zu zeigen, daß G mehrdeutig ist, zeigen wir: `if expr then if expr then stmt else stmt` hat zwei verschiedene Linksableitungen.

1.

$$\begin{aligned}
 S &\Rightarrow \text{if expr then } S \text{ else } S \\
 &\Rightarrow \text{if expr then if expr then } S \text{ else } S \\
 &\Rightarrow \text{if expr then if expr then stmt else } S \\
 &\Rightarrow \text{if expr then if expr then stmt else stmt}
 \end{aligned}$$

2.

$$\begin{aligned}
 S &\Rightarrow \text{if expr then } S \\
 &\Rightarrow \text{if expr then if expr then } S \text{ else } S \\
 &\Rightarrow \text{if expr then if expr then stmt else } S \\
 &\Rightarrow \text{if expr then if expr then stmt else stmt}
 \end{aligned}$$

b) Betrachte Grammatik $G' = (\{S', S_M, S_N\}, \{\text{if, then, else, expr, stmt}\}, P', S')$ mit

$$P' = \left\{ \begin{array}{l} S' \rightarrow S_N | S_M \\ S_N \rightarrow \text{if exp then } S' | \text{if exp then } S_M \text{ else } S_N \\ S_M \rightarrow \text{if exp then } S_M \text{ else } S_M | \text{stmt} \end{array} \right\}$$

G' ist eindeutig.

c) $G = (\{E\}, \{(\cdot, \cdot), \cdot, +, \text{op}\}, P, E)$ mit $P = \{E \rightarrow E + E | E \cdot E | (E) | \text{op}\}$

G ist mehrdeutig, denn $\text{op} \cdot \text{op} + \text{op}$ besitzt 2 Linksableitungen.

$$1. E \Rightarrow E \cdot E \Rightarrow \text{op} \cdot E \Rightarrow \text{op} \cdot E + E \Rightarrow \text{op} \cdot \text{op} + E \Rightarrow \text{op} \cdot \text{op} + \text{op}$$

$$2. E \Rightarrow E + E \Rightarrow E \cdot E + E \Rightarrow \text{op} \cdot E + E \Rightarrow \text{op} \cdot \text{op} + E \Rightarrow \text{op} \cdot \text{op} + \text{op}$$

Betrachte $G' = (\{E\}, \{(\cdot, \cdot), \cdot, +, \text{op}\}, P', E)$ mit $P' = \{E \rightarrow \text{op} + E | \text{op} \cdot E | (E) + E | (E) \cdot E | (E) | \text{op}\}$.

G' ist eindeutig.