

Aufgabe 1

a) Zu zeigen: $L_1 = \{ww \in \{0,1\}^*\}$ ist nicht regulär mit Hilfe der Abschlusseigenschaften für reguläre Sprachen.

Beweis: $L_{1_2} = L_1 \cap \{0^r 10^s 1 \mid r, s \geq 0\} = \{0^n 10^n 1 \mid n \geq 0\}$ ist regulär.

Definiert werde ein Homomorphismus $h_1(a) = 0, h_1(b) = 0, h_1(1) = 1$.

Dann ist $h_1^{-1}(L_{1_2}) \cap \{a^r 1b^s 1 \mid r, s \geq 0\} = \{a^n 1b^n 1 \mid n \geq 0\} = L_{1_3}$ regulär.

Homomorphismus $h_2(a) = a, h_2(b) = b, h_2(1) = \varepsilon$.

Dann ist $h_2(L_{1_3}) = \{a^n b^n \mid n \geq 0\}$ regulär \nexists

b) Zu zeigen: $L_2 = \{0^p \mid p \geq 2, p \text{ prim}\}$ ist nicht regulär.

Angenommen, L_2 wäre regulär. Dann gibt es nach dem Pumping-Lemma $n \in \mathbb{N}, w \in L_2, |w| \geq n, w = xyz, 0 < |y| < n : xy^i z \in L_2 \forall i \in \mathbb{N}$.

Sei p die kleinste Primzahl größer als n ; dann ist $0^p \in L_2$. $w = 0^p = xyz, 0 < |y| < p$ und $xy^{p+1}z \in L_2$, aber $|xy^{p+1}z| = |xyz| + |y|p = p + |y|p = p(1 + |y|)$. Dies ist eine zusammengesetzte Zahl und damit keine Primzahl \nexists

c) $L \subseteq \Sigma^*, L^*$ ist regulär. Ist dann auch L regulär? Benutzt werden kann, daß die Sprache $L = \{0^{n^2} \mid n \geq 0\}$ nicht regulär ist.

Behauptung: L ist nicht zwangsläufig regulär, wenn L^* regulär ist.

Beweis [Gegenbeispiel]: Zu widerlegen ist $L^* \text{ reg} \Rightarrow L \text{ reg} \Leftrightarrow L \text{ nicht reg} \Rightarrow L^* \text{ nicht reg}$.

Die Sprache $L = \{0^{n^2} \mid n \geq 0\}$ ist nicht regulär. Behauptung: $L^* = \{0^n \mid n \geq n\}$ ist regulär.

$$L^* = \bigcup_{i=0}^{\infty} \underbrace{L \cdot L \cdots L}_{i\text{-mal}}$$

Also ist $\varepsilon \in L^*$ und $0^n \in L^*$ für $n \geq 1$, da $0 \in L$. Damit ist L^* regulär \nexists

Aufgabe 2

Eine Grammatik $G = (V, \Sigma, P', S)$ heißt linksregulär kontextfrei, wenn ihre Produktionen von der Form $P = \{A \rightarrow Ba \mid a \in \Sigma\}$ mit $A, B \in V \setminus \Sigma, a \in \Sigma$. Zeige: Eine Sprache L ist regulär, wenn es eine linksreguläre Grammatik G gibt mit $L(G) = L$.

Beweis: Aus dem Skript ist bekannt, daß eine Sprache genau dann regulär ist, wenn eine rechtsreguläre Grammatik dazu existiert. Sei $G' = (V, \Sigma, P', S)$ mit $P' = \{A \rightarrow a \mid A \rightarrow a \in P \wedge a \in \Sigma \cup \{\varepsilon\}\} \cup \{A \rightarrow aB \mid A \rightarrow Ba \in P\}$. G' ist offenbar rechtsregulär.

Behauptung: $L(G) = L \Leftrightarrow L(G') = L^R$

Beweis: Mittels Induktion nach $|w|$ kann man zeigen: $A \xrightarrow[G]{*} w \Leftrightarrow A \xrightarrow[G']{*} w^R$ für $w \in \Sigma^*$ und $A \in V \setminus \Sigma$.

Dann gilt: $w \in L \Leftrightarrow w^R \in L^R$ und $w \in L(G) \Leftrightarrow S \xrightarrow{*}_G w \Leftrightarrow S \xrightarrow{*}_{G'} w^R \Leftrightarrow w^R \in L(G')$.

Also $L(G) = L \Leftrightarrow L(G') = L^R$. Zu einer gegebenen linksregulären Grammatik $L(G) = L$ existiert also eine rechtsreguläre Grammatik G' mit $L(G') = L^R$ und umgekehrt.

Behauptung: L ist regulär \Leftrightarrow es gibt eine linksreguläre Grammatik G mit $L(G) = L$

„ \Rightarrow “: L ist regulär $\Rightarrow L^R$ regulär \Rightarrow es existiert eine rechtsreguläre Grammatik G' mit $L(G') = L^R \Rightarrow$ es existiert eine linksreguläre Grammatik G mit $L(G) = (L^R)^R = L$

„ \Leftarrow “: Es existiert eine linksreguläre Grammatik G mit $L(G) = L \Rightarrow$ es existiert eine rechtsreguläre Grammatik G' mit $L(G') = L^R \Rightarrow L^R$ ist regulär $\Rightarrow (L^R)^R = L$ ist regulär.

Aufgabe 3

Zeige: Jede lineare Sprache L kann von einer Grammatik G erzeugt werden, deren Produktionen von der Form $A \rightarrow aB, A \rightarrow Ba$ oder $A \rightarrow a, A, B \in V \setminus \Sigma$ und $a \in \Sigma \cup \{\varepsilon\}$ sind.

Beweis: Sei L eine lineare Sprache. Dann gibt es eine lineare Grammatik G mit $L(G) = L$, und jede Produktion hat die Form

1. $A \rightarrow a, a \in \Sigma \cup \{\varepsilon\}$ oder
2. $A \rightarrow a_1 a_2 \dots a_n, a_i \in \Sigma_n$ oder
3. $A \rightarrow a_1 a_2 \dots a_n B b_1 b_2 \dots b_m, a_i, b_i \in \Sigma, B \in V \setminus \Sigma, n, m \geq 0$

1. *Schritt:* G in eine reduzierte Grammatik G' umwandeln. Beachte: $G' = (V', \Sigma, P', S')$ ist immer noch linear, denn die 3 Algorithmen verletzen die Linearität nicht. Wir haben also in G' keine zyklischen Produktionen, also gilt für Produktionen vom Typ 3: $n \geq 1$ oder $m \geq 1$.

2. *Schritt:* $P_1 := \{A \rightarrow a \mid A \rightarrow a \in P' \wedge a \in \Sigma \cup \{\varepsilon\}\} \cup \{A \rightarrow aB \mid A \rightarrow aB \in P' \wedge a \in \Sigma\} \cup \{A \rightarrow Ba \mid A \rightarrow Ba \in P' \wedge a \in \Sigma\}$

Produktionen aus $P' \setminus P_1$ sind vom Typ 2 oder 3. Wandle Produktionen vom Typ 2 wie folgt um:

$$\left. \begin{array}{l} A \rightarrow a_1 a_2 \dots a_n \in P' \setminus P_1, a_i \in \Sigma, n \geq 2 \\ A \rightarrow a_1 [a_2 \dots a_n] \\ [a_2 \dots a_n] \rightarrow a_2 [a_3 \dots a_n] \\ [a_{n-1} a_n] \rightarrow a_{n-1} [a_n] \\ [a_n] \rightarrow a_n \end{array} \right\} \text{neue Produktionen}$$

$[a_n \dots a_n]; [a_3 \dots b_n]; [a_{n-1}]; [a_n]$ sind neue Nichtterminale.

Produktionen vom Typ 3 werden wie folgt umgewandelt:

$$\begin{aligned}
 A &\rightarrow a_1 a_2 \dots a_n B b_1 b_2 \dots b_m \in P' \setminus P_1 \\
 A &\rightarrow a_1 [a_2 \dots a_n B b_1 \dots b_m] \\
 [a_2 \dots a_n B b_1 \dots b_m] &\rightarrow a_2 [a_3 \dots a_n B b_1 \dots b_m] \\
 &\vdots \\
 [a_n B b_1 \dots b_m] &\rightarrow a_n [B b_1 b_2 \dots b_m] \\
 [B b_1 b_2 \dots b_m] &\rightarrow [B b_1 \dots b_{m-1}] b_m \\
 [B b_1 \dots b_{m-1}] &\rightarrow [B b_1 \dots b_{m-2}] b_{m-1} \\
 &\vdots \\
 [B b_1] &\rightarrow B b_1
 \end{aligned}$$

Wir bilden nun eine neue Grammatik $G'' = (V'', \Sigma, P'', S')$ mit
 $V'' = V' \cup \{ \text{neue Nichtterminale aus Umwandlungen} \}$
 $P'' = P'_1 \cup \{ \text{neue Produktionen aus Umwandlungen} \}$

Offenbar gilt: $L(G'') = L$ nach Konstruktion und die Produktionen aus P'' sind in der gewünschten Normalform.