

FORMÁLNÍ JAZYKY A AUTOMATY I

Řešení cvičení 6.

1. Jazyk $\{a^n b^n \mid n \geq 1\}^+$ je generován gramatikou $G = (\{S, X, X_a, X_b, A, B\}, \{a, b\}, P, S)$, přičemž množina pravidel P je tvaru

$$\begin{aligned}
 P: \quad & S \rightarrow XS \mid XX \mid AX_b \mid AB \\
 & X \rightarrow AX_b \mid AB \\
 & X_b \rightarrow XB \\
 & A \rightarrow a \\
 & B \rightarrow b
 \end{aligned}$$

2. Necht' $G = (N, T, P, S)$ je gramatika v Chomského normálním tvaru. Gramatika G obsahuje dva typy pravidel: pravidla patřící do $N \times (N \cdot N)$ (nazvěme je neterminální) a pravidla z $N \times T$ (nazvěme je terminální). Necht' $S \Rightarrow_G w_1 \Rightarrow_G w_2 \dots \Rightarrow_G w_n = w$ je odvození délky n ; $w \in L(G)$. Protože se jedná o bezkontextovou gramatiku, ve které žádné pravidlo nemá na pravé straně terminální symbol, můžeme předpokládat, že v uvedeném odvození byla nejdříve aplikována neterminální pravidla a až potom terminální pravidla. Bud' tedy i maximální číslo takové, že $w_i \in N^+$. Protože se každý neterminál přepíše na právě jeden terminál, je $|w_i| = |w|$. Na odvození terminálního řetězce ze slova w_i je potřebných právě $|w|$ kroků. Slovo w_i bylo odvozeno z počátečního neterminálu aplikací neterminálních pravidel, z nichž každé zvětší délku řetězce právě o 1. Proto na odvození řetězce w_i bylo zapotřebí $|w| - 1$ kroků. Sečtením: $n = |w| + |w| - 1$ a tedy $\frac{n+1}{2} = |w|$.

3. Konstrukce gramatiky \overline{G} spočívá v následujících třech krocích.

(i) Množinu neterminálů N rozšíříme o nové neterminály, jeden pro každý terminální symbol z T , t.j. $\overline{N} = N \cup \{N_a \mid a \in T\}$.

(ii) Množinu pravidel P transformujeme tak, že každý výskyt terminálu a nahradíme příslušným neterminálem N_a . t.j.

$$\begin{aligned}
 \overline{P} = \{ & \alpha_1 \dots \alpha_k \rightarrow \beta_1 \dots \beta_l \mid x_1 \dots x_k \rightarrow y_1 \dots y_l \in P; \\
 & \alpha_i = x_i \text{ pro } x_i \in N; \\
 & \alpha_i = N_{x_i} \text{ pro } x_i \in T; \\
 & \beta_i = y_i \text{ pro } y_i \in N; \\
 & \beta_i = N_{y_i} \text{ pro } y_i \in T\}
 \end{aligned}$$

(iii) Nakonec do množiny pravidel \overline{P} přidáme pravidla, umožňující přepis nových neterminálů na terminály, t.j.

$$\overline{P} = \overline{P} \cup \{N_a \rightarrow a \mid \text{pro všechny neterminály } N_a \in \overline{N} - N\}.$$

Všechny nově přidaná pravidla jsou regulární. Transformací v bodě (ii) se může porušit regularita pravidel. Proto navržený algoritmus zkonstruuje k regulární gramatice G gramatiku \overline{G} , která nemusí být nutně regulární, ale jistě je bezkontextová. Pro ostatní gramatiky konstrukce zachovává typ gramatiky.

4. a) $L_1 \cap L_2 = \{a^1 b a^2 b a^3 b \dots a^j b \mid j \geq 2; j \text{ sudé}\}$.

b) Předpokládejme, že jazyk $L_1 \cap L_2$ je bezkontextový. Pak podle pumping lemy pro bezkontextové jazyky existuje číslo p takové, že pro všechna slova $w \in L_1 \cap L_2$, $|w| > p$ jsou splněny následující podmínky:

(a) $\exists u, v, x, y, z : w = uvxyz$,

(b) $|vy| > 0$; $|vxy| < p$,

(c) $\forall i \geq 0 : uv^i x y^i z \in L_1 \cap L_2$.

Zvolme například slovo $w = a^1 b a^2 b \dots a^p b$. Skusme najít slova u, v, x, y, z splňující 1.–3. Zřejmě $v \neq b$, $y \neq b$ (slovo z $L_1 \cap L_2$ nemůže obsahovat podřetězec bb). Kdyby $v = a^i$ (anebo $v = a^i$), tak napumpováním podle 3. se změní počet symbolů a v příslušné skupině a poruší se vztah mezi počtem symbolů a ve sousedních skupinách. Kdyby $v = a^i b a^j$ (anebo $v = a^i b a^j$), tak napumpováním podle 3. dostáváme slovo, ve kterém dvě po sobě jdoucí skupiny symbolů a mají stejnou délku.