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

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

2. Nechť $G = (N, T, P, S)$ 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). Nechť $S \Longrightarrow_G w_1 \Longrightarrow_G w_2 \ldots \Longrightarrow_G w_n = w$ 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 neterminálová pravidla a až potom terminálová pravidla. Buď 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 řetězu ze slova w_i je potřebných právě $|w|$ kroků. Slovo w_i bylo odvozeno z počátečného neterminálu aplikací neterminálových pravidel, z nichž každé zvětší délku řetězu právě o 1. Proto na odvození řetězu 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.

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

   (iii) Nakonec do množiny pravidel $\overline{P}$ přidáme pravidla, umožňující přepis nových neterminálů na terminály, t.j.
   $\overline{P} = \overline{P} \cup \{N_a \longrightarrow a \mid \text{pro všechny neterminály } N_a \in \overline{N} - N\}$.

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

4. **a)** $L_1 \cap L_2 = \{a^1ba^2ba^3b \ldots a^jb \mid j \geq 2; j \text{ sudé }\}$.
   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) $\forall i \geq 0 \ : \ uv^ixy^iz \in L_1 \cap L_2$.
   Zvolme například slovo $w = a^1ba^2b \ldots a^pb$. 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 napumpováváním podle 3. se změní počet symbolů a v příslušné skupině a poruší se vztah mezi počtem symbolů a ve sousedních skupinách. Kdyby $v = a^iba^j$ (anebo $v = a^iba^j$), tak napumpováváním podle 3. dostávame slovo, ve kterém dvě po sobě jdoucí skupiny symvolů a mají stejnou délku.