**IB102 – úkol 9 – řešení** **Odevzdání: 5. 12. 2011**

Vypracoval: James Bond UČO: 007
Skupina: MI6

**1.** [**2 body**] Mějme následující jazyk:

$$L = \{w \in \{a, b, c, d\}^* \mid \#_a(w) = 2\#_b(w) \text{ a } \#_a(w) < \#_c(w)\}$$

Rozhodněte, zda je tento jazyk bezkontextový, a své rozhodnutí dokažte. (Pro důkaz toho, že je jazyk bezkontextový, stačí sestrojit příslušnou bezkontextovou gramatiku nebo zásobníkový automat.)

**Řešení:** Jazyk $L$ není bezkontextový. Dokážeme to pomocí lemmatu o vkládání (Pumping Lemma) pro bezkontextové jazyky.

Nechť $n$ je libovolné přirozené číslo. Zvolíme slovo $z = a^{2n}b^nc^{2n+1}$. Zřejmě platí $z \in L$ a $|z| > n$. Nyní prozkoumáme všechna rozdělení $z = uvwxy$ taková, že $|vwx| \leq n$ a $vx \neq \varepsilon$. Každé takové rozdělení je jednoho z těchto druhů:

- Část $v$ nebo $x$ obsahuje alespoň jedno $a$. Potom zřejmě ani $v$ ani $x$ neobsahují žádné $c$. Zvolíme $i = 2$, pak zřejmě platí $\#_a(uv^2wx^2y) \geq \#_c(uv^2wx^2y)$, a tedy $uv^2wx^2y \notin L$. (Pumpováním se zvětší počet $a$, ale počet $c$ se nezmění.)
- Části $v$ ani $x$ neobsahují žádné $a$, ale alespoň jedna z nich obsahuje alespoň jedno $b$. Zvolíme $i = 0$, pak zřejmě platí $\#_a(uv^2wx^2y) > 2\#_b(uv^2wx^2y)$, a tedy $uv^2wx^2y \notin L$. (Pumpováním se zmenší počet $b$, ale počet $a$ se nezmění.)
- Části $v$ ani $x$ neobsahují žádná $a$ ani žádná $b$, musí tedy obsahovat pouze symboly $c$. Zvolíme $i = 0$, pak zřejmě platí $\#_a(uv^2wx^2y) \geq \#_c(uv^2wx^2y)$, a tedy $uv^2wx^2y \notin L$. (Pumpováním se zmenší počet $c$, ale počet $a$ se nezmění.)

Je jasné, že tyto tři body pokrývají všechny možnosti, které mohou nastat. Ukázali jsme tedy, že pro každé rozdělení $z = uvwxy$ je možno najít $i$ takové, že $uv^iwx^iy \notin L$. Podle lemmatu o vkládání pro bezkontextové jazyky tedy $L$ není bezkontextový.

## IB102 – úkol 9 – řešení Odevzdání: 5. 12. 2011

Vypracoval: James Bond UČO: 007
Skupina: MI6

**2.** [**2 body**] Mějme gramatiku $G = (\{S, A, B, C, D\}, \{a, b\}, P, S)$, kde

$$\begin{aligned} P = \{ \quad S &\to Aa \mid BaC, \\ A &\to Sa \mid AD, \\ B &\to Sb \mid b, \\ C &\to CCa \mid b, \\ D &\to aD \mid bD \quad \}. \end{aligned}$$

Převeďte tuto gramatiku do Greibachové normální formy použitím algoritmů z přednášky.

*Poznámka:* Nezapomeňte si nejdřív zkontrolovat, zda je pro použití algoritmů gramatika ve vhodném vstupním tvaru.

**Řešení:** Pro provedení převodu na GNF potřebujeme, aby byla gramatika bez levé rekurze a vlastní. Algoritmus pro odstranění levé rekurze rovněž vyžaduje na vstupu vlastní gramatiku. Zadaná gramatika není vlastní, protože není redukovaná, obsahuje totiž nenormovaný neterminál. Provedeme tedy nejprve odstranění zbytečných neterminálů. Zřejmě jediný nenormovaný neterminál je $D$ a po jeho odstranění jsou všechny neterminály normované a dosažitelné. Máme tedy $G' = (\{S, A, B, C\}, \{a, b\}, P', S)$, kde

$$\begin{aligned} P' = \{ \quad S &\to Aa \mid BaC, \\ A &\to Sa, \\ B &\to Sb \mid b, \\ C &\to CCa \mid b \quad \}. \end{aligned}$$

Tato grmatika už je vlastní, můžeme tedy provést algoritmus pro odstranění levé rekurze. Nejprve si zvolíme uspořádání na neterminálech, například $S < A < B < C$. Dále provádíme substituci podle algoritmu. Pravidla $S$ se nemění:

$$S \to Aa \mid BaC$$

V pravidlech pro $A$ nejprve substituujeme za počáteční $S$:

$$A \to Aaa \mid BaCa$$

Následně odstraníme bezprostřední levou rekurzi:

$$\begin{aligned} A &\to BaCa \mid BaCaA' \\ A' &\to aa \mid aaA' \end{aligned}$$

V pravidlech pro $B$ nejprve substituujeme za počáteční $S$:

$$B \to Aab \mid BaCb \mid b$$

Následně substituujeme za počáteční $A$:

$$B \to BaCaab \mid BaCaA'ab \mid BaCb \mid b$$

A nakonec odstraníme bezprostřední levou rekurzi:

$$\begin{aligned} &B \to b \mid bB' \\ &B' \to aCaab \mid aCaA'ab \mid aCb \mid aCaabB' \mid aCaA'abB' \mid aCbB' \end{aligned}$$

V pravidlech pro $C$ pouze odstraníme bezprostřední levou rekurzi:

$$\begin{aligned} C &\to b \mid bC' \\ C' &\to Ca \mid CaC' \end{aligned}$$

Výsledná gramatika bez levé rekurze je $G'' = (\{S, A, A', B, B', C, C'\}, \{a, b\}, P'', S)$, kde

$$\begin{array}{rrcl} P'' = \{ & S & \to & Aa \mid BaC, \\ & A & \to & BaCa \mid BaCaA', \\ & A' & \to & aa \mid aaA', \\ & B & \to & b \mid bB', \\ & B' & \to & aCaab \mid aCaA'ab \mid aCb \mid aCaabB' \mid aCaA'abB' \mid aCbB', \\ & C & \to & b \mid bC', \\ & C' & \to & Ca \mid CaC' \quad \}. \end{array}$$

Tato gramatika je zřejmě i vlastní, můžeme tedy rovnou pokračovat algoritmem pro převod na GNF. Lineární uspořádání splňující podmínku v algoritmu je například $C' < B' < A' < S < A < B < C$. Provedeme tedy substituci podle algoritmu a nahradíme terminály na nepočátečních pozicích neterminály. Dostaneme tak výslednou gramatiku:

$$G''' = (\{S, A, A', B, B', C, C', a', b'\}, \{a, b\}, P''', S), \text{ kde}$$

$$\begin{array}{rrcl} P''' = \{ & C & \to & b \mid bC', \\ & B & \to & b \mid bB', \\ & A & \to & ba'Ca' \mid bB'a'Ca' \mid ba'Ca'A' \mid bB'a'Ca'A', \\ & S & \to & ba'Ca'a' \mid bB'a'Ca'a' \mid ba'Ca'A'a' \mid bB'a'Ca'A'a' \mid ba'C \mid bB'a'C, \\ & A' & \to & aa' \mid aa'A', \\ & B' & \to & aCa'a'b' \mid aCa'A'a'b' \mid aCb' \mid aCa'a'bxB' \mid aCa'A'a'bxB' \mid aCb'B', \\ & C' & \to & ba' \mid bC'a' \mid ba'C' \mid bC'a'C' \quad \}. \end{array}$$