JAIO - Zadanie 2

Hubert Michalski

30 kwietnia 2023

1 Zadanie pierwsze

$$L_{\exists} = \{ab^{n_1}ab^{n_2}a \dots ab^{n_k}a \in \{a,b\}^* \mid \exists i \in \mathbb{N}.1 \leq i \leq k \land n_i = k\}$$

Gramatyka generująca język $L_{\exists}$ to $\mathcal{G}$ z symbolem startowym S:

$$\mathcal{G} = \begin{cases} S & \rightarrow & aLbRa \\ L & \rightarrow & BaLb \mid \varepsilon \\ R & \rightarrow & bRaB \mid \varepsilon \\ B & \rightarrow & Bb \mid \varepsilon \end{cases}$$

Można pokazać, że gramatyka jest poprawna, czyli zachodzi $L(\mathcal{G}) \subseteq L_{\exists}$. Zgodnie z definicją języka $L_{\exists}$ słowa posiadające $k+1$ znaków a muszą zawierać co najmniej jeden segment długości k znaków b. Słowo aba należy do $L_{\exists}$, ponieważ istnieje tylko jeden segment znaków b długości $n_1 = k = 1$. Dalej obserwujemy, że produkcje L i R na każdy nowo generowany znak a dodają do "skrajnego" segmentu (L do skrajnie prawego, a R do skrajnie lewego) znak b. Zatem zachowywany jest niezmiennik taki, że istnieje segment długości k znaków b podczas gdy słowo ma $k+1$ znaków a, a to oznacza, że słowa generowane przez tę gramatykę należą do języka $L_{\exists}$.

Aby udowodnić inkluzję $L_{\exists} \subseteq L(\mathcal{G})$ weźmy słowo $w \in L_{\exists}$ i skonstruujmy derywację tego słowa. Jeśli $w = aba$ szukana derywacja wygląda tak:

$$S \rightarrow aLbRa \rightarrow a\varepsilon bRa \rightarrow a\varepsilon b\varepsilon a$$

Przypuśćmy teraz, że $|w| > 3$. Bez straty ogólności załóżmy, że słowo w posiada $k+1$ liter a dla $k \in \mathbb{N}, k > 1$. Słowo w posiada co najmniej jeden segment znaków b długości k - oznaczmy przez i początek **pierwszego** takiego segmentu, czyli $w[i \dots i+k-1]$ wyznacza segment samych liter

b długości k. Dodatkowo oznaczmy liczbę liter a w prefiksie słowa w do indeksu i przez $l = \#_a(w[1 \dots i])$. Zatem widzimy, że słowo w jest postaci:

$$w = ab^{n_1}a \dots b^{n_{(l-1)}}ab^k ab^{n_{(l+1)}} \dots ab^{n_k}a$$

Zauważmy, że teraz jeśli chcemy otrzymać derywację słowa w wystarczy zastosować produkcję $L \to BaLb$ dokładnie $(l-1)$ razy, ponieważ podczas każdego kolejnego zastosowania produkcji w słowie pojawia się dokładnie jedna litera a oraz dokładnie jedna litera b w skrajnie prawym bloku, a produkcję $R \to bRaB$ dokładnie $(k+1) - 2 - (l-1) = k - l$ razy z analogicznej przyczyny. Pierwsze i ostatnie znaki a są produkowane z S. Ostatnim krokiem będzie rozwinięcie wszystkich nie-terminali B do oczekiwanej liczby znaków b w każdym segmencie. Skonstruujmy więc derywację dla danego słowa w:

$$S \to aLbRa \to a\mathbf{BaLb}bRa \to \dots \to a\underbrace{Ba\dots Ba}_{2\cdot(l-1)}\underbrace{b\dots b}_{(l-1)}bRa$$

Następnie postępujemy analogicznie z drugiej strony:

$$a\underbrace{Ba\dots Ba}_{2\cdot(l-1)}\underbrace{b\dots b}_{(l-1)}bRa \to \dots \to a\underbrace{Ba\dots Ba}_{2\cdot(l-1)}\underbrace{b\dots b}_{(l-1)}b\underbrace{b\dots b}_{(k-l)}\underbrace{aB\dots aB}_{2\cdot(k-l)}a$$

Teraz wystarczy rozwinąć wszystkie nie-terminale B do oczekiwanej liczby znaków b w każdym z segmentów słowa w:

$$a\underbrace{Ba\dots Ba}_{2\cdot(l-1)}\underbrace{b\dots b}_{k}\underbrace{aB\dots aB}_{2\cdot(k-l)}a \to \dots \to ab^{n_1}a \dots b^{n_{(l-1)}}ab^k ab^{n_{(l+1)}} \dots ab^{n_k}a$$

Otrzymujemy w ten sposób derywację dowolnego słowa $w \in L_\exists$ zatem gramatyka $\mathcal{G}$ generuje wszystkie słowa z tego języka. ■

2 Zadanie drugie

$$L_\forall = \{ab^{n_1}ab^{n_2}a\dots ab^{n_k}a \in \{a,b\}^* \mid \forall i \in \mathbb{N}.\ 1 \le i \le k \implies n_i = k\}$$

Udowodnijmy, że podany język nie jest bezkontekstowy z wykorzystaniem lematu o pompowaniu dla języków bezkontekstowych. Załóżmy, że $L_\forall$ jest bezkontekstowy i niech n będzie jak z lematu. Niech $w = ab^nab^n \dots ab^na$ (czyli $k = n$). Rozważmy faktoryzację $w = \mathit{prefix}\cdot\mathit{left}\cdot\mathit{infix}\cdot\mathit{right}\cdot\mathit{suffix}$ jak w lemacie. Przypomnijmy, że co najmniej jedno z *left*, *right* jest niepuste, zatem dokładnie jeden z poniższych przypadków zachodzi:

1. *left* i *right* zawierają same litery b
2. *left* lub *right* zawiera literę a

Przypuśćmy, że zachodzi przypadek pierwszy. Weźmy $w' = \mathit{prefix}\cdot\mathit{left}^2\cdot\mathit{infix}\cdot\mathit{right}^2\cdot\mathit{suffix}$, wtedy na pewno jeden z segmentów b, do których należały *left* i *right* jest większy od k, a ponieważ liczba liter a się nie zmieniła to liczba segmentów liter b dalej jest równa k. Zatem $w' \notin L_\forall$, ponieważ $\exists_i : n_i > k$ co jest sprzeczne z definicją języka.

Rozważmy teraz drugi przypadek i weźmy w' jak w pierwszym punkcie. Bez straty ogólności załóżmy, że litera a występuje w *left* (może wystąpić maksymalnie jedna ponieważ $|\mathit{left}\cdot\mathit{infix}\cdot\mathit{right}| \le n$). Zauważmy, że gdy napompujemy *left* to zwiększa się liczba liter a, co za tym idzie powstaje nowy segment liter b (być może pusty), czyli zwiększamy k do $k+1$. Łatwo zaobserwować, że nowo utworzony segment liter b musi mieć długość mniejszą niż $k+1$ (ponownie dlatego, że $|\mathit{left}\cdot\mathit{infix}\cdot\mathit{right}| \le n$). Co za tym idzie $\exists_i : n_i < k+1$, czyli $w' \notin L_\forall$. Zatem język $L_\forall$ nie jest bezkontekstowy. ■