

Wstęp do informatyki

Lista 12

Uwaga:

Algorytmy rozwiązujące zadania z tej listy mogą mieć postać pseudokodu, w szczególności mogą wykorzystywać abstrakcyjne typy danych jak kolejka, stos i in.

1. [1] Podaj gramatykę bezkontekstową opisującą dziesiętny zapis liczb całkowitych. Liczba taka może zaczynać się znakiem + lub – (ale nie jest on konieczny) po czym następuje niepusty ciąg cyfr. Zapis liczby nie może zawierać zbędnych wiodących zer, a liczba 0 nie powinna być poprzedzana znakiem + ani –.

Przykład: 0; 123; -15; +999 to poprawne zapisy liczb, natomiast +0; 01; +–3; +09; + są niepoprawne.

2. [1] Podaj bezkontekstową gramatykę opisującą zapis nieujemnych liczb rzeczywistych z przecinkiem dziesiętnym, w którym każda liczba ma tyle samo cyfr przed przecinkiem i po przecinku.

Przykład: 0,0; 0090,1117; 1,9 to poprawne zapisy, natomiast 0,00; 90,1117; 1,90 oraz zapis złożony jedynie z przecinka są niepoprawne w tym zadaniu.

3. [2] Wyrażenia należące do języka gramatyki $G(N, T, P, \langle \text{wyr} \rangle)$, gdzie zbiór produkcji w postaci BNF jest równy:

$$\langle \text{wyr} \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$$

$$\langle \text{wyr} \rangle ::= \langle \text{wyr} \rangle + \langle \text{wyr} \rangle$$

$$\langle \text{wyr} \rangle ::= \langle \text{wyr} \rangle - \langle \text{wyr} \rangle$$

$$\langle \text{wyr} \rangle ::= (\langle \text{wyr} \rangle)$$

mogą mieć wiele różnych drzew wyprowadzeń, Podaj przykład ilustrujący ten fakt, a następnie zaproponuj gramatykę, której język zawierać będzie tylko takie wyrażenia, które (dzięki nawiasom!) mają dokładnie jedno drzewo wyprowadzenia.

4. [1] Podaj jaki język definiuje gramatyka $G(N, T, P, S)$, gdzie $N = \{S, X, Y, P, R, A, B\}$, $T = \{a, b, c\}$ a zbiór produkcji jest następujący:

$$S \rightarrow X, S \rightarrow Y$$

$$X \rightarrow P C$$

$$Y \rightarrow A R$$

$$P \rightarrow a P b, P \rightarrow a b$$

$$R \rightarrow b R c, R \rightarrow b c$$

$$A \rightarrow A a, A \rightarrow a$$

$$C \rightarrow c C, C \rightarrow c$$

5. [1] Przeanalizuj działanie pierwszej wersji algorytmu konwersji wyrażenia do postaci ONP z wykładu (nie uwzględniającej priorytetów operatorów) na wyrażeniu postaci

$$a_1 p_1 a_2 p_2 \dots p_{k-1} a_k$$

dla $k > 1$, gdzie a_1, a_2, \dots, a_k to argumenty a p_1, p_2, \dots, p_{k-1} to operatory. Następnie:

- a) podaj któremu z poniższych sposobów nawiasowania odpowiada utworzona postać ONP:

$$(((a_1 p_1 a_2) p_2 a_3) \dots p_{k-1} a_k) \\ (a_1 p_1 \dots (a_{k-2} p_{k-2} (a_{k-1} p_{k-1} a_k)))$$

- b) wskaż modyfikację algorytmu, która spowoduje przekształcenie wyrażenia $a_1 p_1 a_2 p_2 \dots p_{k-1} a_k$ do postaci odpowiadającej nawiasowaniu z punktu a), którego nie wskazała(e)s w punkcie a).

6. [1] Podaj algorytm sprawdzający, czy podany na wejściu ciąg w_1, \dots, w_n gdzie

$$w_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, * \}$$

dla $i = 1, 2, \dots, n$ jest poprawnym wyrażeniem w postaci ONP. Uzasadnij poprawność swojego algorytmu, oszacuj jego złożoność czasową.

Wskazówki:

- spróbuj wykorzystać podany na wykładzie algorytm wyznaczania wartości wyrażenia w postaci ONP;
- przykłady napisów, które nie są wyrażeniami ONP: $5\ 6\ 7\ +$ (za dużo argumentów), $5\ +\ 6\ 7\ *$ (brak drugiego argumentu dla +).

7. [3] Opisz język $L(G)$ zdefiniowany przez gramatykę $G(N, T, P, S)$, gdzie $N = \{S, C\}$, $T = \{a, b, c\}$ a zbiór produkcji jest następujący:

$$\begin{aligned} S &\rightarrow S\ a\ S\ b\ S \mid S\ b\ S\ a\ S \mid C\ S \mid S\ C \mid \varepsilon \\ C &\rightarrow c\ C \mid \varepsilon \end{aligned}$$

Znaku „|” używamy jako alternatywy, tak jak w notacji BNF (np. $S \rightarrow X \mid Y$ oznacza dwie produkcje: $S \rightarrow X$ i $S \rightarrow Y$). Udowodnij, że wskazany przez Ciebie język jest równy $L(G)$.

Zadania dodatkowe, nieobowiązkowe (nie wliczają się do puli punktów do zdobycia na ćwiczeniach, punktacja została podana tylko jako informacja o trudności zadań wg wykładowcy)

8. [0] Wskaż, które z poniższych wyrażeń w postaci ONP są niepoprawne. Dla pozostałych podaj wartości i odpowiadające im wyrażenia w postaci standardowej (infiksowej):

$$\begin{array}{ll} 5\ 1\ +\ 2\ 3\ *\ 9\ +\ 8\ 7\ * & 1\ 2\ 3\ 4\ 5\ +\ +\ +\ + \\ 5\ 1\ 7\ +\ 2\ 3\ *\ 9\ +\ 8\ 7\ * & 1\ 2\ 3\ 4\ 5\ +\ +\ +\ + \end{array}$$

9. [0.5] Mamy gramatykę G definiującą składnię instrukcji języka C , za wyjątkiem instrukcji `for`. W szczególności gramatyka zawiera nieterminal $\langle \text{wyr} \rangle$, z którego można wyprowadzić wszystkie wyrażenia oraz nieterminal $\langle \text{ins} \rangle$ opisujący instrukcje. Podaj jak rozszerzyć taką gramatykę tak, aby dopuszczała również instrukcję `for`.
10. [1] Gramatyki podane na tej liście zadań i gramatyki tworzone w ramach rozwiązań zadań z listy zapisz w postaci BNF i EBNF (wykorzystaj dostępne w BNF i EBN dodatkowe notacje skracające opis gramatyk).
11. [3*] Wyznacz ile drzew wyprowadzeń w gramatyce z symbolem startowym S , zbiorem nieterminali $N = \{S\}$ i produkcjami

$$\begin{aligned} S &\rightarrow 0, \dots, S \rightarrow 9 \\ S &\rightarrow S\ +\ S \\ S &\rightarrow S\ *\ S \\ S &\rightarrow (S) \end{aligned}$$

może mieć słowo postaci

$$a_1\ p_1\ a_2\ p_2\ \dots\ p_{k-1}\ a_k$$

gdzie $a_1, a_2, \dots, a_k \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ oraz $p_1, p_2, \dots, p_{k-1} \in \{+, *\}$.