

Zaawansowane języki programowania  
Laboratorium (Analiza składniowa - parsery zstępujące)

**Tutaj** znajduje się przykładowy parser w języku **ANSI C**.

**Zadanie 1** (1 pkt). Napisać parser zstępujący poniższych gramatyk:

a)  $\langle S \rangle \rightarrow a \langle S \rangle d \mid \langle B \rangle$   
 $\langle B \rangle \rightarrow b \langle B \rangle c \mid e$

b)  $\langle S \rangle \rightarrow \langle A \rangle \langle B \rangle$   
 $\langle A \rangle \rightarrow a \langle A \rangle \mid x$   
 $\langle B \rangle \rightarrow b \langle B \rangle \mid x$

Jakie języki generują następujące gramatyki?

Zasada **First<sub>n</sub>(α)** mówi dla jakich gramatyk można utworzyć parser zstępujący. Treść tej zasady wygląda następująco:

$$\text{First}_n(\alpha) = \{a_1 a_2 \dots a_n \mid \alpha \Rightarrow^* a_1 a_2 \dots a_n \beta\}.$$

Czyli **First<sub>n</sub>(α)** jest zbiorem n-literowych napisów terminalowych, od których mogą się zaczynać napisy wyprowadzalne z α, gdzie α jest produkcją.

Mówimy że gramatyka należy do **LL(n)** jeśli zachodzi następujący warunek: Dla nieterminali mających więcej niż jedną produkcję α|β|... zachodzi:

$$\text{First}_n(\alpha) \cap \text{First}_n(\beta) = \emptyset.$$

Przykład gramatyki należącej do **LL(1)**:

$$\langle A \rangle \rightarrow a \langle B \rangle \mid b \langle A \rangle b \mid \langle B \rangle b$$
$$\langle B \rangle \rightarrow c \langle B \rangle \mid d$$

Przykład gramatyki nie należącej do **LL(n)**:

$$\langle A \rangle \rightarrow \langle B \rangle \mid \langle A \rangle + \langle B \rangle$$
$$\langle B \rangle \rightarrow a \mid (A)$$

Można tą gramatykę zastąpić następująco:

$$\langle A \rangle \rightarrow \langle B \rangle \{+ \langle B \rangle\}^*$$
$$\langle B \rangle \rightarrow a \mid (A)$$

Zapis  $\{ \}^*$  oznacza zero lub więcej powtórzeń nawiasu klamrowego. Do obsługi  $\{ \}^*$  parser powinien używać pętli **while**.

**Zadanie 2** (1.5 pkt). Napisać parser zstępujący do powyższej gramatyki.