

## Zbiory FIRST i FOLLOW

Założmy, że w gramatyce znajduje się ciąg produkcji:

$$A ::= \gamma_1 \mid \dots \mid \gamma_n$$

Zbiór “FIRST” ( symboli startowych/początkowych ? ):

**FIRST**( $\gamma_i$ ) = {  $x$ :  $x\beta$  wywodzi się z  $\gamma_i$  , gdzie  $x$  należy do  $T$ , a  $\beta$  jest słowem }

Zbiór “FOLLOW” ( symboli końcowych ? ):

**FOLLOW**(**A**) = {  $x$ :  $\beta A \delta$  wywodzi się z  $S$ , gdzie  $x \in \text{FIRST}(\delta)$ , a  $\beta$  i  $\delta$  są słowami }

*Przykład:*

$$X ::= aAy \mid Az \quad A ::= B \mid C \mid d \quad B ::= e \mid \epsilon \quad C ::= f \mid \epsilon$$

$$\text{FIRST}(B) = \{e\} \quad \text{FOLLOW}(A) = \{y,z\}$$

## Wyznaczanie symboli startowych

- Jeżeli  $x$  jest terminalem, to  $FIRST(x) = \{x\}$
- Jeżeli w gramatyce istnieje produkcja  $A ::= \epsilon$ , to  $FIRST(A) = FIRST(A) + \{\epsilon\}$
- Jeżeli  $X$  jest nieterminalem, to dla wszystkich produkcji postaci:

$$X ::= Y_1 Y_2 \dots Y_n$$

- $FIRST(X) = FIRST(X) + FIRST(Y_1)$
- Jeżeli z  $Y_1$  da się wywieść  $\epsilon$ , to dodatkowo

$$FIRST(X) = FIRST(X) + FIRST(Y_2)$$

( Postępuj tak, dopóki z  $Y_i$  da się wywieść  $\epsilon$  )

- Jeżeli  $\epsilon$  należy do każdego zbioru  $FIRST(Y_i)$   $i=1,2,\dots,n$ ,

$$\text{to } FIRST(X) = FIRST(X) + \{\epsilon\}$$

$$FIRST(E) = FIRST(T) = FIRST(F) = \{ ( , id \}$$

$$FIRST(E') = \{ + , \epsilon \} \quad FIRST(T') = \{ * , \epsilon \}$$

## Wyznaczanie symboli końcowych

- $\text{FOLLOW}(S) = \{ eof \}$

- Jeżeli istnieje produkcja  $A ::= \alpha B \beta$ , to:

$$\text{FOLLOW}(B) = \text{FOLLOW}(B) + (\text{FIRST}(\beta) - \{\epsilon\})$$

- Jeżeli istnieje produkcja  $A ::= \alpha B$  lub produkcja  $A ::= \alpha B \beta$  i  $\epsilon$  należy do  $\text{FIRST}(\beta)$ , to:

$$\text{FOLLOW}(B) = \text{FOLLOW}(B) + \text{FOLLOW}(A)$$

$E ::= TE'$

$T ::= FT'$

$E' ::= +TE' \mid \epsilon$

$T' ::= *FT' \mid \epsilon$

$F ::= (E) \mid id$

$\text{FOLLOW}(E) = \{ ), \mathbf{EOF} \}$

$\text{FOLLOW}(T) = \{ +, ), \mathbf{EOF} \}$

$\text{FOLLOW}(F) = \{ +, *, ), \mathbf{EOF} \}$

## Tablice parsingu dla parsera LL(1)

1. Dla każdej produkcji postaci:  $A ::= \alpha$  wykonaj:

a) Dla każdego symbolu terminalnego  $a$  ze zbioru  $FIRST(\alpha)$

$$TP[A,a] = A ::= \alpha$$

b) Jeżeli  $\epsilon$  należy do  $FIRST(\alpha)$ , to:

$$TP[A,b] = A ::= \alpha$$

dla każdego symbolu terminalnego  $b$  ze zbioru  $FOLLOW(A)$

c) Jeżeli  $\epsilon$  należy do  $FIRST(\alpha)$  oraz EOF do  $FOLLOW(A)$ , to:

$$TP[A,EOF] = A ::= \alpha$$

2. Pozostałe miejsca w tablicy oznacz jako błędne.

## Przykłady zastosowań w parserach

- *Alternatywa* rekurencyjnych
- *Zero lub więcej wystąpień*
- *Wystąpienie opcjonalne*

Mechanizm “panic mode” z zastosowaniem zbiorów FIRST i FOLLOW

Używamy zbiorów jako elementów synchronizujących przy pomijaniu symboli z wejścia

- Przykłady dla rzeczywistych gramatyk
- Dyskusja na temat innych metod odzyskiwania kontroli

# Budowa drzewa wyvodu w analizatorze rekurencyjnym

- Reprezentacja węzłów (programowanie strukturalne i obiektowe)
- Poddrzewa instrukcji i deklaracji
- Kolejność działań w wyrażeniach arytmetycznych
- Reakcja na błędy
- Dwa rodzaje drzew:
  - Syntax (parse) tree
  - Abstract Syntax Tree