

Algoritmo para calcular FIRST(X):

Para computar FIRST(X) para um símbolo X da gramática, aplicam-se as regras abaixo, até que não se possa adicionar mais terminais ou ϵ ao conjunto em questão.

- 1) Se a é um terminal, então $\text{FIRST}(a) = \{a\}$.
- 2) Se $X \rightarrow \epsilon$ é uma produção, então adicione ϵ a $\text{FIRST}(X)$.
- 3) Se $X \rightarrow Y_1 Y_2 \dots Y_k$ é uma produção e, para algum i , todos

$X \rightarrow Y_1 Y_2 \dots Y_{i-1}$ derivam ϵ , então $\text{FIRST}(Y_i)$ está em $\text{FIRST}(X)$, juntamente com todos os símbolos não- ϵ de $\text{FIRST}(Y_1)$, $\text{FIRST}(Y_2)$, ..., $\text{FIRST}(Y_{i-1})$. O símbolo ϵ será adicionado a $\text{FIRST}(X)$ apenas se todo $Y_j (j=1, 2, \dots, k)$ derivar ϵ .

Algoritmo para calcular FOLLOW(X):

1. $\text{FOLLOW}(S) := \{\$ \}$
2. repeat
3. foreach $p \in P$ do {
4. // varre as produções
5. case $p = A \rightarrow \alpha B \beta$ {
6. $\text{FOLLOW}(B) := \text{FOLLOW}(B) \cup \text{FIRST}(\beta) \setminus \{\epsilon\}$
7. if $\epsilon \in \text{FIRST}(\beta)$ then
8. $\text{FOLLOW}(B) := \text{FOLLOW}(B) \cup \text{FOLLOW}(A)$
9. end
10. }
11. case $p = A \rightarrow \alpha B$
12. $\text{FOLLOW}(B) := \text{FOLLOW}(B) \cup \text{FOLLOW}(A)$
13. }
14. until no change in any $\text{FOLLOW}(V)$

Algoritmo para construir uma tabela de análise preditiva:

Entrada: gramática G

Resultado: Tabela de Análise M

Método:

- 1) Para cada produção $A \rightarrow \alpha$ de G, execute os passos 2 e 3 (para criar a linha A da tabela M).
- 2) Para cada terminal a de $\text{FIRST}(\alpha)$, adicione a produção $A \rightarrow \alpha$ a $M[A, a]$.
- 3) Se $\text{FIRST}(\alpha)$ inclui a palavra vazia, então adicione $A \rightarrow \alpha$, a $M[A, b]$ para cada b em $\text{FOLLOW}(A)$.

Exemplo. Determinação das funções FIRST

Exemplo: Considere uma gramática para expressão lógica:

$$E \rightarrow TE'$$

$$E' \rightarrow \vee TE' \mid \epsilon$$

$$T \rightarrow FT'$$

$$T' \rightarrow \wedge FT' \mid \epsilon$$

$$F \rightarrow \neg F \mid \text{id}$$

obs.: Conjuntos FIRST: Convém iniciar pelos não-terminais mais simples, isto é, aqueles que derivam apenas formas sentenciais que iniciam por terminais.

- FIRST(F) = { \neg , id }
- FIRST(T') = { \wedge , ϵ }
- FIRST(E') = { \vee , ϵ }
- FIRST(T) = { \neg , id }
- FIRST(E) = { \neg , id }

Conjuntos FOLLOW:

$$\text{FOLLOW}(E) = \{ \$ \}$$

$$\text{FOLLOW}(E') = \{ \$ \}$$

$$\text{FOLLOW}(T) = \{ \vee, \$ \}$$

$$\text{FOLLOW}(T') = \{ \vee, \$ \}$$

$$\text{FOLLOW}(F) = \{ \vee, \wedge, \$ \}$$

Para $E \rightarrow TE'$	tem-se $\text{FIRST}(TE') = \{ \neg, \text{id} \}$	$M[E, \neg] = M[E, \text{id}] = E \rightarrow TE'$
Para $E' \rightarrow \vee TE'$	tem-se $\text{FIRST}(\vee TE') = \{ \vee \}$	$M[E', \vee] = E' \rightarrow \vee TE'$
Para $E' \rightarrow \epsilon$	tem-se $\text{FOLLOW}(E') = \{ \$ \}$	$M[E', \$] = E' \rightarrow \epsilon$
Para $T \rightarrow FT'$	tem-se $\text{FIRST}(FT') = \{ \neg, \text{id} \}$	$M[T, \neg] = M[T, \text{id}] = T \rightarrow FT'$
Para $T' \rightarrow \wedge FT'$	tem-se $\text{FIRST}(\wedge FT') = \{ \wedge \}$	$M[T', \wedge] = T' \rightarrow \wedge FT'$
Para $T' \rightarrow \epsilon$	tem-se $\text{FOLLOW}(T') = \{ \vee, \$ \}$	$M[T', \vee] = M[T', \$] = T' \rightarrow \epsilon$
Para $F \rightarrow \neg F$	tem-se $\text{FIRST}(\neg F) = \{ \neg \}$	$M[F, \neg] = F \rightarrow \neg F$
Para $F \rightarrow \text{id}$	tem-se $\text{FIRST}(\text{id}) = \{ \text{id} \}$	$M[F, \text{id}] = F \rightarrow \text{id}$

	id	\vee	\wedge	\neg	\$
E	$E \rightarrow TE'$			$E \rightarrow TE'$	
E'		$E' \rightarrow \vee TE'$			$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$	
T'		$T' \rightarrow \epsilon$	$T' \rightarrow \wedge FT'$		$T' \rightarrow \epsilon$
F	$F \rightarrow \text{id}$			$F \rightarrow \neg F$	