



# Compiladores

## Aula 5

**Celso Olivete Júnior**

**`olivete@fct.unesp.br`**

# Na aula de hoje

## ☐ Analisador sintático descendente sem recursão

### ☐ Conjuntos *First* e *Follow*

✓ exercícios

### ☐ Montagem da tabela de análise sintática

✓ exercícios

# Analizador sintático descendente

- ❑ Produz uma **derivação mais à esquerda** – a partir de **uma gramática LL(1)** - para uma cadeia de entrada
- ❑ Tem como principal problema determinar, a cada passo, qual produção deve ser aplicada para substituir um o símbolo não-terminal
- ❑ Quando uma produção é escolhida, o restante do processo de análise consiste em **casar** os **símbolos terminais da produção com a cadeia de entrada**

## ☐ Gramática LL(1)

LL(1) = leitura **L**eft → right

+ derivação mais a esquerda (**L**eft)

+ uso de **1** token lookahead.

## ☐ Gramáticas LL(1) devem ser:

☐ Não recursiva à esquerda

☐ Fatorada a esquerda



# Analizador sintático descendente

## Conjuntos *First* e *Follow*

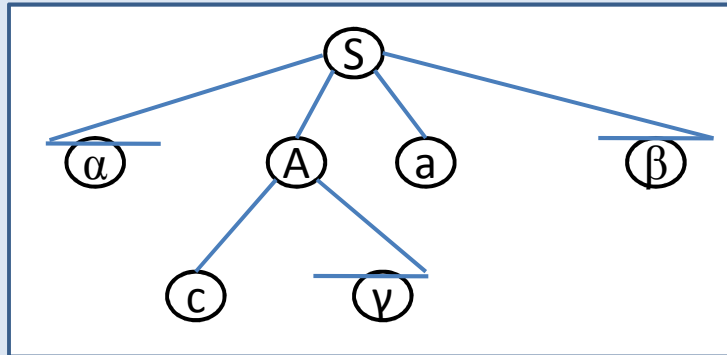
- ❑ São funções que auxiliam a construção de analisadores sintáticos
- ❑ Permitem escolher qual produção deve ser aplicada baseada no próximo símbolo de entrada

# Analizador sintático descendente

## *First e Follow*

- ❑ **First:** Define o conjunto de símbolos que iniciam derivações a partir de uma sequência de símbolos terminais e não-terminais

➤  $c$  está em **First**( $A$ )



- ❑ **Follow:** Define o conjunto de símbolos terminais que podem aparecer imediatamente à direita de um dado símbolo não terminal

➤  $a$  está em **Follow**( $A$ )

## Regras para o conjunto *First*

- ❑ Para calcular  $\text{FIRST}(X)$  de todos os símbolos  $X$  da gramática, as seguintes regras devem ser aplicadas até que não haja mais terminais ou  $\epsilon$ :
  1. Se  $X$  é um símbolo terminal, então  $\text{FIRST}(X) = \{X\}$
  2. Se  $X$  é um símbolo não-terminal e  $X ::= Y_1 Y_2 \dots Y_k$  é uma produção para algum  $k \geq 1$ , então:
    - acrescente  $a$  a  $\text{First}(X)$  se, para algum  $i$ ,  $a$  estiver em  $\text{FIRST}(Y_i)$ , e  $\epsilon$  estiver em todos os  $\text{FIRST}(Y_1), \dots, \text{FIRST}(Y_{i-1})$
    - adicione  $\epsilon$  se  $\epsilon$  está em  $\text{FIRST}(Y_j)$  para todo  $j = 1, 2, \dots, k$
    - Se  $Y_1$  não derivar  $\epsilon$ , nada mais deve ser acrescentado a  $\text{FIRST}(X)$
  3. Se  $X ::= \epsilon$  é uma produção, então acrescente  $\epsilon$  a  $\text{FIRST}(X)$

# Regras para o conjunto *First* ilustrando a regra 1

**First:** Define o conjunto de símbolos que iniciam derivações a partir de uma sequência de símbolos terminais e não-terminais

**Regra 1.** Se  $X$  é um símbolo terminal, então  $FIRST(X) = \{X\}$

## Gramática

1.  $S \rightarrow ABd$
2.  $A \rightarrow aA \mid \epsilon$
3.  $B \rightarrow bB \mid cA \mid AC$
4.  $C \rightarrow cB \mid \epsilon$

$$First(A) = First(a) \cup First(\epsilon)$$

$$\begin{array}{cc} \downarrow & \downarrow \\ \{a\} & \{\epsilon\} \end{array}$$

$$\downarrow \\ First(A) = \{a, \epsilon\}$$



# Regras para o conjunto *First* ilustrando a regra 2

**Regra 2.** Se  $X$  é um símbolo não-terminal e  $X ::= Y_1 Y_2 \dots Y_k$  é uma produção para algum  $k \geq 1$ , então:

- acrescente  $a$  a  $\text{First}(X)$  se, para algum  $i$ ,  $a$  estiver em  $\text{FIRST}(Y_i)$ , e  $\epsilon$  estiver em todos os  $\text{FIRST}(Y_1), \dots, \text{FIRST}(Y_{i-1})$
- adicione  $\epsilon$  se  $\epsilon$  está em  $\text{FIRST}(Y_j)$  para todo  $j = 1, 2, \dots, k$
- Se  $Y_1$  não derivar  $\epsilon$ , nada mais deve ser acrescentado a  $\text{FIRST}(X)$

## Gramática

1.  $S \rightarrow ABd$
2.  $A \rightarrow aA \mid \epsilon$
3.  $B \rightarrow bB \mid cA \mid AC$
4.  $C \rightarrow cB \mid \epsilon$

$\text{First}(S) = \text{First}(A)$

$\{a, \epsilon\}$

$\text{First}(B) = \text{First}(b) \cup \text{First}(c) \cup \text{First}(A)$

$\{b\}$

$\{c\}$

$\{a, \epsilon\}$

$\text{First}(S) = \{a, b, c, d\}$

Compiladores

Logo  $\text{First}(B)$  e  $\text{First}(d)$  deverão fazer parte de  $\text{First}(S)$

Como  $B$  não deriva em  $\epsilon$  então  $\epsilon$  é removido de  $\text{First}(S)$

## Regras para o conjunto *First* ilustrando a **regra 3**

**Regra 3.** Se  $X ::= \epsilon$  é uma produção, então  
acrescente  $\epsilon$  a  $\text{FIRST}(X)$

### Gramática

1.  $S \rightarrow ABd$
2.  $A \rightarrow aA \mid \epsilon$
3.  $B \rightarrow bB \mid cA \mid AC$
4.  $C \rightarrow cB \mid \epsilon$

$\text{First}(A)$



$\{a, \epsilon\}$

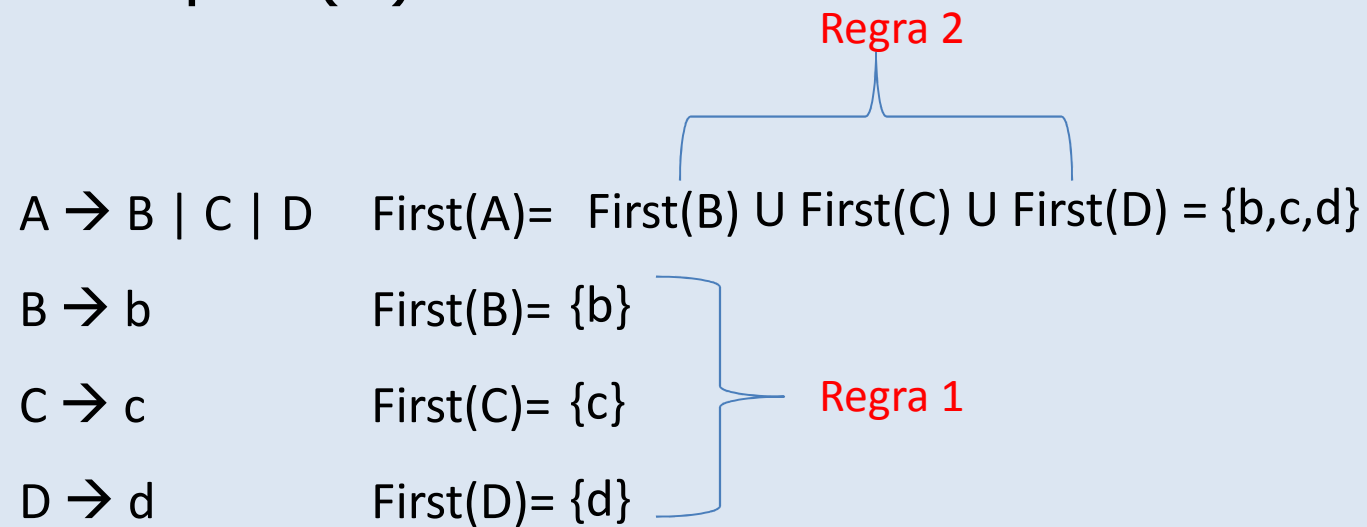
$\text{First}(C)$



$\{c, \epsilon\}$

# Conjunto *First*

## ❑ Exemplo (1)



# Conjunto *First*

## Aplicação

❑ Dada a gramática, como derivar a entrada **bde**?

$S \rightarrow ACE$   
 $A \rightarrow a \mid b \mid e$   
 $C \rightarrow c \mid d \mid e$   
 $E \rightarrow e$

```
S () {  
  switch {  
    case 'a':  
    case 'b':  
      A(); //expande regra A e tenta consumir o b  
      C(); //expande regra C e tenta consumir o d  
      E(); //expande regra E e consome e  
      break;  
    default:  
      abort("syntax_error");  
  }  
}
```

# Conjunto *First*

## □ Exemplo (2)

$E \rightarrow TE'$	$\text{First}(E) = \text{First}(T) = \{ (, id \}$
$E' \rightarrow +TE' \mid \epsilon$	$\text{First}(E') = \text{First}(+) \cup \text{First}(\epsilon) = \{ +, \epsilon \}$
$T \rightarrow FT'$	$\text{First}(T) = \text{First}(F) = \{ (, id \}$
$T' \rightarrow *FT' \mid \epsilon$	$\text{First}(T') = \text{First}(*) \cup \text{First}(\epsilon) = \{ *, \epsilon \}$
$F \rightarrow (E) \mid id$	$\text{First}(F) = \text{First}( ( ) \cup \text{First}(id) = \{ (, id \}$

Regra 2
Regra 3
Regra 1



unesp

# Conjunto *First*

## proc First( $\alpha$ : string de símbolos)

1. // seja  $\alpha = X_1 X_2 X_3 \dots X_n$
2. if  $X_1 \in T$
3.     then // caso simples onde  $X_1$  é um terminal
4.          $\text{First}(\alpha) := \{X_1\}$
5.     else { //  $X_1$  não é um terminal
6.          $i := 0$ ;  $\text{First}(\alpha) = \text{First}\{X_1\} \setminus \{\epsilon\}$ ;
7.         for ( $i=1$  ;  $i \leq n$  ;  $i++$ ) {
8.             if  $\epsilon$  is in  $\text{First}(X_1)$  and in  $\text{First}(X_2)$  and in...  $\text{First}(X_{i-1})$
9.              $\text{First}(\alpha) := \text{First}(\alpha) \cup \text{First}(X_i) \setminus \{\epsilon\}$
10.         }
11.     }
12. If  $(\alpha \rightarrow \epsilon)$  é uma produção
13.     then  $\text{First}(\alpha) := \text{First}(\alpha) \cup \{\epsilon\}$

# Exercícios (I)

- ❑ Encontre o conjunto *First* (primeiro) para os símbolos não-terminais de cada uma das gramáticas:

G1	G2	G3	G4
$S \rightarrow AB$ $A \rightarrow aA \mid a$ $B \rightarrow bB \mid b$	$S \rightarrow ABd$ $A \rightarrow aA \mid \epsilon$ $B \rightarrow bB \mid cA \mid AC$ $C \rightarrow cB \mid \epsilon$	$S \rightarrow A \mid BC$ $A \rightarrow aAS \mid D$ $B \rightarrow bB \mid fAC \mid \epsilon$ $C \rightarrow cC$ $D \rightarrow gD \mid C \mid \epsilon$	$S \rightarrow XYZ$ $X \rightarrow aXb \mid \epsilon$ $Y \rightarrow cYZcX \mid d$ $Z \rightarrow eZYe \mid f$

# Exercícios (I)

## resolução

❑ Encontre o conjunto *First* (primeiro) para os símbolos não-terminais de cada uma das gramáticas:

G1	G2	G3	G4
$S \rightarrow AB$ $A \rightarrow aA \mid a$ $B \rightarrow bB \mid b$ $\text{First}(S) = \{ a \}$ $\text{First}(A) = \{ a \}$ $\text{First}(B) = \{ b \}$	$S \rightarrow ABd$ $A \rightarrow aA \mid \epsilon$ $B \rightarrow bB \mid cA \mid AC$ $C \rightarrow cB \mid \epsilon$ $\text{First}(S) = \{ b, c, a, d \}$ $\text{First}(A) = \{ a, \epsilon \}$ $\text{First}(B) = \{ b, c, a, \epsilon \}$ $\text{First}(C) = \{ c, \epsilon \}$	$S \rightarrow A \mid BC$ $A \rightarrow aAS \mid D$ $B \rightarrow bB \mid fAC \mid \epsilon$ $C \rightarrow cC$ $D \rightarrow gD \mid C \mid \epsilon$ $\text{First}(S) = \{ a, g, c, b, f \}$ $\text{First}(A) = \{ a, g, c, \epsilon \}$ $\text{First}(B) = \{ b, f, \epsilon \}$ $\text{First}(C) = \{ c \}$ $\text{First}(D) = \{ g, c, \epsilon \}$	$S \rightarrow XYZ$ $X \rightarrow aXb \mid \epsilon$ $Y \rightarrow cYZcX \mid d$ $Z \rightarrow eZYe \mid f$ $\text{First}(X) = \{ a, \epsilon \}$ $\text{First}(Y) = \{ c, d \}$ $\text{First}(Z) = \{ e, f \}$ $\text{First}(S) = \{ a, c, d \}$



## Conjunto *Follow*

- ❑ **Follow** é o conjunto de terminais que podem aparecer à direita de um não-terminal  $A$  em uma sentença válida.
- ❑  $\$$  passa a denotar um terminal “virtual” que marca o fim da entrada (EOF)

## Regras para o conjunto *Follow*

- ❑ Para calcular FOLLOW(X) de todos os não-terminais A, as seguintes regras devem ser aplicadas até que nada mais possa ser acrescentado a nenhum dos conjuntos FOLLOW:
  1. Se X é o símbolo inicial da gramática coloque \$ em FOLLOW(X), onde \$ é o marcador de fim da entrada
  2. Se houver uma produção  $A \rightarrow \alpha X \beta$ , então tudo em FIRST( $\beta$ ) exceto  $\epsilon$  está em FOLLOW(X)
  3. Se houver uma produção  $A \rightarrow \alpha X$ , ou uma produção  $A \rightarrow \alpha X \beta$ , onde o FIRST( $\beta$ ) contém  $\epsilon$ , então inclua o FOLLOW(A) em FOLLOW(X)

# Conjunto *Follow*

## ❑ Exemplo (1)

$S \rightarrow A B$   
 $A \rightarrow c \mid \epsilon$   
 $B \rightarrow cbB \mid ca$

$\text{First}(A) = \text{First}(c) \cup \text{First}(\epsilon) = \{c, \epsilon\}$

$\text{First}(B) = \{c\}$

$\text{First}(S) = \text{First}(A) \cup \text{First}(B) - \epsilon = \{c\}$

$\text{Follow}(A) = \text{First}(B) = \{c\}$

Regra 2

$\text{Follow}(B) = \{\$ \}$

$\text{Follow}(S) = \{\$ \}$

Início da G  
→ Regra 1

# Conjunto *Follow*

## Exemplo 2

 $E \rightarrow TE'$ 
 $E' \rightarrow +TE' \mid \epsilon$ 
 $T \rightarrow FT'$ 
 $T' \rightarrow *FT' \mid \epsilon$ 
 $F \rightarrow (E) \mid id$ 
 $First(E) = First(T) = \{ (, id \}$ 
 $First(E') = First(+) \cup First(\epsilon) = \{ +, \epsilon \}$ 
 $First(T) = First(F) = \{ (, id \}$ 
 $First(T') = First(*) \cup First(\epsilon) = \{ *, \epsilon \}$ 
 $First(F) = First(() \cup First(id) = \{ (, id \}$ 

Aparece apenas nas extremidades de E

 $Follow(E) = First() \cup \{ \$ \} = \{ ), \$ \}$ 
 $Follow(E') = Follow(E) = \{ ), \$ \}$ 
 $Follow(T) = First(E') \cup Follow(E') = \{ +, ), \$ \}$ 
 $Follow(T') = Follow(T) = \{ +, ), \$ \}$ 
 $Follow(F) = First(T') \cup Follow(T') = \{ *, +, ), \$ \}$ 

1. Se X é o símbolo inicial da gramática coloque \$ em FOLLOW(X), onde \$ é o marcador de fim da entrada
2. Se houver uma produção  $A \rightarrow \alpha X \beta$ , então tudo em FIRST( $\beta$ ) exceto  $\epsilon$  está em FOLLOW(X)
3. Se houver uma produção  $A \rightarrow \alpha X$ , ou uma produção  $A \rightarrow \alpha X \beta$ , onde o FIRST( $\beta$ ) contém  $\epsilon$ , então inclua o FOLLOW(A) em FOLLOW(X)

## Conjunto *Follow*

proc Follow( $A \in N$ )

```
1. Follow(S) := {$};
2. Repeat
3.   foreach  $p \in P$  do {
4.     // Varre as produções
5.     case  $p == A \rightarrow \alpha B \beta$  {
6.       Follow(B) := Follow(B)  $\cup$  First( $\beta$ ) \ { $\epsilon$ };
7.       if  $\epsilon \in$  First( $\beta$ ) then
8.         Follow(B) := Follow(B)  $\cup$  Follow(A);
9.       end
10.    }
11.    case  $p == A \rightarrow \alpha B$ 
12.      Follow(B) := Follow(B)  $\cup$  Follow(A);
13.  }
14. Until no change in any Follow(N)
```

## Exercícios (II)

- ❑ Encontre o conjunto *Follow* (seguidor) para os símbolos não-terminais de cada uma das gramáticas:

G1
$S \rightarrow AB$
$A \rightarrow aA \mid a$
$B \rightarrow bB \mid b$

G2
$S \rightarrow ABd$
$A \rightarrow aA \mid \epsilon$
$B \rightarrow bB \mid cA \mid AC$
$C \rightarrow cB \mid \epsilon$

G3
$S \rightarrow A \mid BC$
$A \rightarrow aAS \mid D$
$B \rightarrow bB \mid fAC \mid \epsilon$
$C \rightarrow cC$
$D \rightarrow gD \mid C \mid \epsilon$

G4
$S \rightarrow XYZ$
$X \rightarrow aXb \mid \epsilon$
$Y \rightarrow cYZcX \mid d$
$Z \rightarrow eZYe \mid f$

# Analísadores sintáticos preditivos

- ❑ Não necessitam de retrocesso
- ❑ O símbolo da cadeia de entrada, em análise, é suficiente para determinar qual regra de produção deve ser escolhida
- ❑ São construídos utilizando gramáticas **LL(1)**
  - Cadeia de entrada analisada da esquerda para a direita (**L**eft-to-right)
  - A derivação das produções é feita mais a esquerda (**L**eftmost)
  - A cada passo é observado um (**1**) símbolo a frente para determinação de que ação deve ser tomada

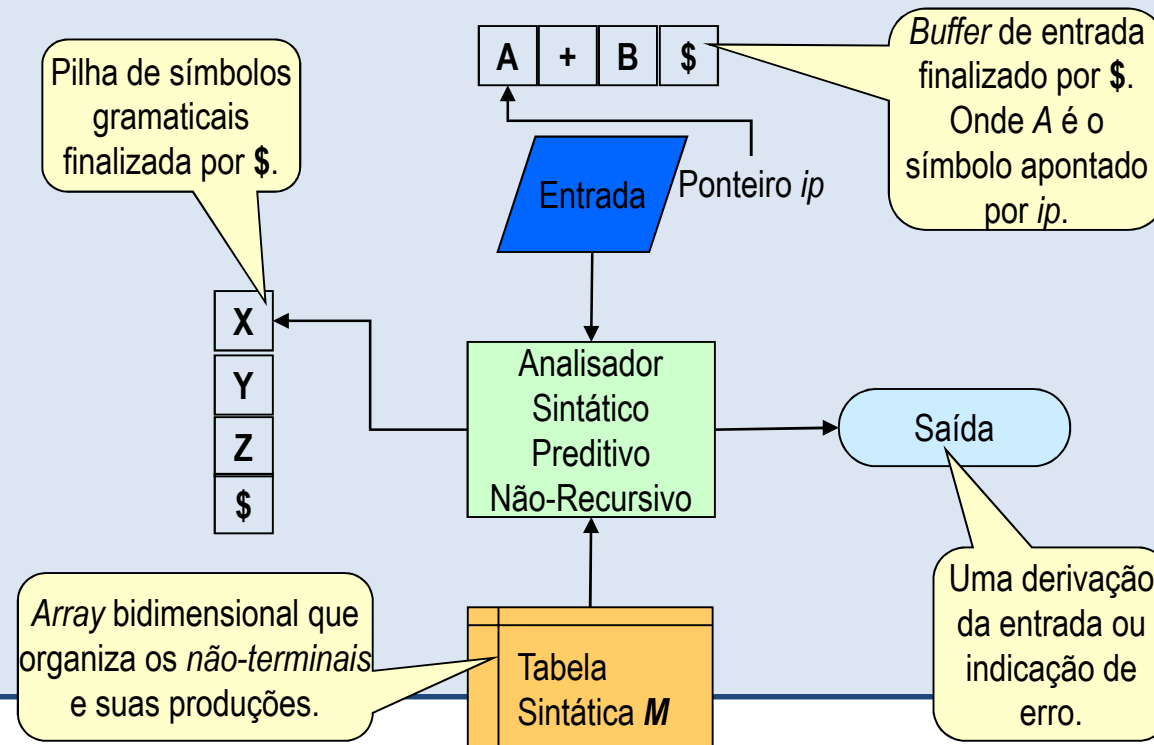
## □ Condições:

- Gramática sem ambiguidade
- Gramática sem recursão a esquerda
- Gramática fatorada
- Construir os conjuntos **First** e **Follow**
  - ✓ Permitem escolher qual produção deve ser aplicada baseada no próximo símbolo de entrada



# Reconhecedor sintático preditivo estrutura

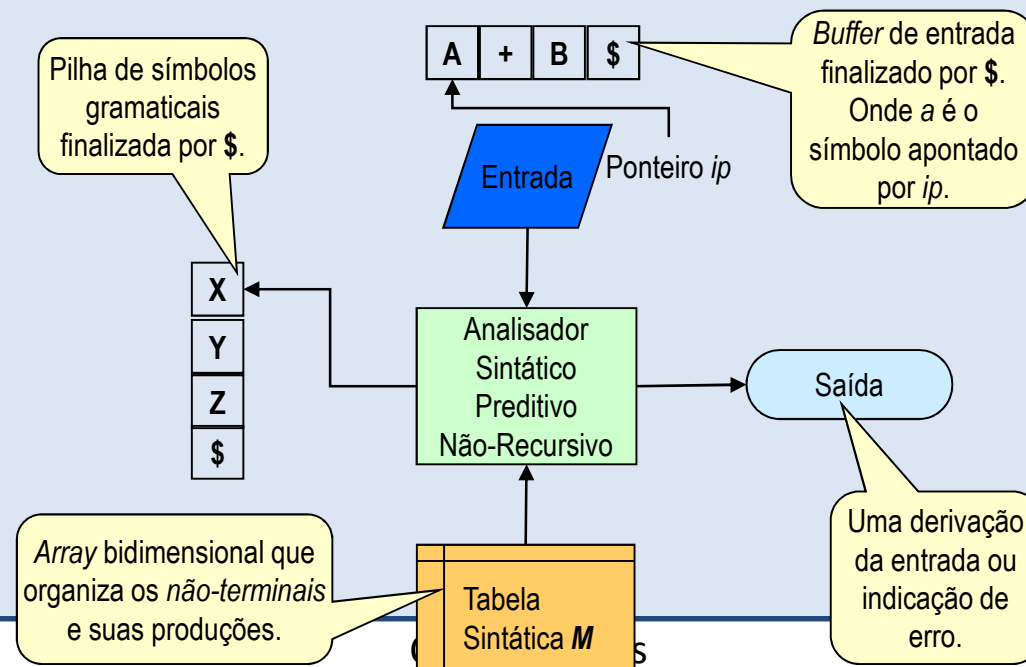
- ❑ Um *buffer* de entrada → \$ delimita o fim
- ❑ Um fluxo de **saída**
- ❑ Uma pilha cujo fundo é marcado por \$
  - ❑ Inicializada com o símbolo de início da gramática
- ❑ Uma tabela sintática preditiva



# Reconhecedor sintático preditivo

## funcionamento do *parser*

- ❑ Seja **X** o símbolo no topo da pilha
- ❑ Seja **a** o símbolo de entrada (que é terminal) a analisar
  1. Se  $X = \$$  e  $a = \$ \rightarrow$  finaliza e reconhece a entrada.
  2. Se  $X = a \neq \$ \rightarrow$  desempilha X e avança de um símbolo na entrada.
  3. Se X é não-terminal: Consulta a tabela  $M(X, a)$ 
    - $\rightarrow$  Se for vazia: **ERRO**
    - $\rightarrow$  Se contém  $X ::= UVW$ , então substitui na pilha X por UVW (U no topo).



# Exemplo 1:

## Construção da tabela preditiva $M(X,t)$

### ❑ Tabela bi-dimensional

- Dimensão 1: não-terminal  $X$
- Dimensão 2: caractere de entrada (terminal)  $t$
- A entrada  $(X,t)$  contém a regra da produção a aplicar  $\rightarrow$  obtida a partir dos conjuntos **First** e **Follow**

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S				
A				
B				

$S \rightarrow cAa$   
 $A \rightarrow cB \mid B$   
 $B \rightarrow bcB \mid \epsilon$

$\text{First}(A) = \{b, c, \epsilon\}$   
 $\text{First}(B) = \{b, \epsilon\}$   
 $\text{First}(S) = \{c\}$

$\text{Follow}(A) = \{a\}$   
 $\text{Follow}(B) = \text{First}(B) - \epsilon = \{a\}$   
 $\text{Follow}(S) = \{\$ \}$

# Exemplo 1:

## Construção da tabela preditiva $M(X,t)$

### ❑ Tabela bi-dimensional

- Dimensão 1: não-terminal  $X$
- Dimensão 2: caractere de entrada (terminal)  $t$
- A entrada  $(X,t)$  contém a regra da produção a aplicar  $\rightarrow$  obtida a partir dos conjuntos **First** e **Follow**

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S			$S \rightarrow cAa$	
A				
B				

$S \rightarrow cAa$   
 $A \rightarrow cB \mid B$   
 $B \rightarrow bcB \mid \epsilon$

$\text{First}(A) = \{b, c, \epsilon\}$   
 $\text{First}(B) = \{b, \epsilon\}$   
 $\text{First}(S) = \{c\}$

$\text{Follow}(A) = \{a\}$   
 $\text{Follow}(B) = \text{First}(B) - \epsilon = \{a\}$   
 $\text{Follow}(S) = \{\$ \}$

# Exemplo 1:

## Construção da tabela preditiva $M(X,t)$

### ❑ Tabela bi-dimensional

- Dimensão 1: não-terminal  $X$
- Dimensão 2: caractere de entrada (terminal)  $t$
- A entrada  $(X,t)$  contém a regra da produção a aplicar  $\rightarrow$  obtida a partir dos conjuntos **First** e **Follow**

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S			$S \rightarrow cAa$	
A	$A \rightarrow B$	$A \rightarrow B$		
B				

$S \rightarrow cAa$   
 $A \rightarrow cB \mid B$   
 $B \rightarrow bcB \mid \epsilon$

$\text{First}(A) = \{b, c, \epsilon\}$   
 $\text{First}(B) = \{b, \epsilon\}$   
 $\text{First}(S) = \{c\}$

$\text{Follow}(A) = \{a\}$   
 $\text{Follow}(B) = \text{First}(B) - \epsilon = \{a\}$   
 $\text{Follow}(S) = \{\$ \}$

# Exemplo 1:

## Construção da tabela preditiva $M(X,t)$

### ❑ Tabela bi-dimensional

- Dimensão 1: não-terminal  $X$
- Dimensão 2: caractere de entrada (terminal)  $t$
- A entrada  $(X,t)$  contém a regra da produção a aplicar  $\rightarrow$  obtida a partir dos conjuntos **First** e **Follow**

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S			$S \rightarrow cAa$	
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	
B				

$S \rightarrow cAa$   
 $A \rightarrow cB \mid B$   
 $B \rightarrow bcB \mid \epsilon$

$\text{First}(A) = \{b, c, \epsilon\}$   
 $\text{First}(B) = \{b, \epsilon\}$   
 $\text{First}(S) = \{c\}$

$\text{Follow}(A) = \{a\}$   
 $\text{Follow}(B) = \text{First}(B) - \epsilon = \{a\}$   
 $\text{Follow}(S) = \{\$ \}$

# Exemplo 1:

## Construção da tabela preditiva $M(X,t)$

### ❑ Tabela bi-dimensional

- Dimensão 1: não-terminal  $X$
- Dimensão 2: caractere de entrada (terminal)  $t$
- A entrada  $(X,t)$  contém a regra da produção a aplicar  $\rightarrow$  obtida a partir dos conjuntos **First** e **Follow**

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S			$S \rightarrow cAa$	
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	
B	$B \rightarrow \epsilon$			

$S \rightarrow cAa$   
 $A \rightarrow cB \mid B$   
 $B \rightarrow bcB \mid \epsilon$

$\text{First}(A) = \{b, c, \epsilon\}$   
 $\text{First}(B) = \{b, \epsilon\}$   
 $\text{First}(S) = \{c\}$

$\text{Follow}(A) = \{a\}$   
 $\text{Follow}(B) = \text{First}(B) - \epsilon = \{a\}$   
 $\text{Follow}(S) = \{\$ \}$

# Exemplo 1:

## Construção da tabela preditiva $M(X,t)$

### ❑ Tabela bi-dimensional

- Dimensão 1: não-terminal  $X$
- Dimensão 2: caractere de entrada (terminal)  $t$
- A entrada  $(X,t)$  contém a regra da produção a aplicar  $\rightarrow$  obtida a partir dos conjuntos **First** e **Follow**

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S			$S \rightarrow cAa$	
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	
B	$B \rightarrow \epsilon$	$B \rightarrow bcB$		

$S \rightarrow cAa$   
 $A \rightarrow cB \mid B$   
 $B \rightarrow bcB \mid \epsilon$

$\text{First}(A) = \{b, c, \epsilon\}$   
 $\text{First}(B) = \{b, \epsilon\}$   
 $\text{First}(S) = \{c\}$

$\text{Follow}(A) = \{a\}$   
 $\text{Follow}(B) = \text{First}(B) - \epsilon = \{a\}$   
 $\text{Follow}(S) = \{\$ \}$



# Exemplo 1:

## Construção da tabela preditiva $M(X,t)$

### ❑ Tabela bi-dimensional

- Dimensão 1: não-terminal  $X$
- Dimensão 2: caractere de entrada (terminal)  $t$
- A entrada  $(X,t)$  contém a regra da produção a aplicar  $\rightarrow$  obtida a partir dos conjuntos **First** e **Follow**

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S	ERRO	ERRO	$S \rightarrow cAa$	ACEITA
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	ERRO
B	$B \rightarrow \epsilon$	$B \rightarrow bcB$	ERRO	ERRO

$S \rightarrow cAa$   
 $A \rightarrow cB \mid B$   
 $B \rightarrow bcB \mid \epsilon$

$\text{First}(A) = \{b, c, \epsilon\}$   
 $\text{First}(B) = \{b, \epsilon\}$   
 $\text{First}(S) = \{c\}$

$\text{Follow}(A) = \{a\}$   
 $\text{Follow}(B) = \text{First}(B) - \epsilon = \{a\}$   
 $\text{Follow}(S) = \{\$ \}$

# Usando a tabela de análise sintática

❑ Entrada:  
"cbca"

Pilha	Entrada	Ação
S\$	cbca\$	$S \rightarrow cAa$

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S	ERRO	ERRO	$S \rightarrow cAa$	ACEITA
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	ERRO
B	$B \rightarrow \epsilon$	$B \rightarrow bcB$	ERRO	ERRO

# Usando a tabela de análise sintática

❑ Entrada:  
"cbca"

Pilha	Entrada	Ação
S\$	cbca\$	$S \rightarrow cAa$
cAa\$	cbca\$	Casar c

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S	ERRO	ERRO	$S \rightarrow cAa$	ACEITA
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	ERRO
B	$B \rightarrow \epsilon$	$B \rightarrow bcB$	ERRO	ERRO

# Usando a tabela de análise sintática

❑ Entrada:  
"cbca"

Pilha	Entrada	Ação
S\$	cbca\$	$S \rightarrow cAa$
cAa\$	cbca\$	Casar c
Aa\$	bca\$	$A \rightarrow B$

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S	ERRO	ERRO	$S \rightarrow cAa$	ACEITA
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	ERRO
B	$B \rightarrow \epsilon$	$B \rightarrow bcB$	ERRO	ERRO

# Usando a tabela de análise sintática

❑ Entrada: "cbca"

Pilha	Entrada	Ação
S\$	cbca\$	$S \rightarrow cAa$
cAa\$	cbca\$	Casar c
Aa\$	bca\$	$A \rightarrow B$
Ba\$	bca\$	$B \rightarrow bcB$

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S	ERRO	ERRO	$S \rightarrow cAa$	ACEITA
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	ERRO
B	$B \rightarrow \epsilon$	$B \rightarrow bcB$	ERRO	ERRO

# Usando a tabela de análise sintática

❑ Entrada:  
"cbca"

Pilha	Entrada	Ação
S\$	cbca\$	$S \rightarrow cAa$
cAa\$	cbca\$	Casar c
Aa\$	bca\$	$A \rightarrow B$
Ba\$	bca\$	$B \rightarrow bcB$
bcBa\$	bca\$	Casar b

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S	ERRO	ERRO	$S \rightarrow cAa$	ACEITA
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	ERRO
B	$B \rightarrow \epsilon$	$B \rightarrow bcB$	ERRO	ERRO

# Usando a tabela de análise sintática

❑ Entrada:  
"cbca"

Pilha	Entrada	Ação
S\$	cbca\$	$S \rightarrow cAa$
cAa\$	cbca\$	Casar c
Aa\$	bca\$	$A \rightarrow B$
Ba\$	bca\$	$B \rightarrow bcB$
bcBa\$	bca\$	Casar b
cBa\$	ca\$	Casar c

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S	ERRO	ERRO	$S \rightarrow cAa$	ACEITA
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	ERRO
B	$B \rightarrow \epsilon$	$B \rightarrow bcB$	ERRO	ERRO

# Usando a tabela de análise sintática

❑ Entrada:  
"cbca"

Pilha	Entrada	Ação
S\$	cbca\$	$S \rightarrow cAa$
cAa\$	cbca\$	Casar c
Aa\$	bca\$	$A \rightarrow B$
Ba\$	bca\$	$B \rightarrow bcB$
bcBa\$	bca\$	Casar b
cBa\$	ca\$	Casar c
Ba\$	a\$	$B \rightarrow \epsilon$

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S	ERRO	ERRO	$S \rightarrow cAa$	ACEITA
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	ERRO
B	$B \rightarrow \epsilon$	$B \rightarrow bcB$	ERRO	ERRO



# Usando a tabela de análise sintática

❑ Entrada:  
"cbca"

Pilha	Entrada	Ação
S\$	cbca\$	$S \rightarrow cAa$
cAa\$	cbca\$	Casar c
Aa\$	bca\$	$A \rightarrow B$
Ba\$	bca\$	$B \rightarrow bcB$
bcBa\$	bca\$	Casar b
cBa\$	ca\$	Casar c
Ba\$	a\$	$B \rightarrow \epsilon$
a\$	a\$	Casar a

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S	ERRO	ERRO	$S \rightarrow cAa$	ACEITA
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	ERRO
B	$B \rightarrow \epsilon$	$B \rightarrow bcB$	ERRO	ERRO

# Usando a tabela de análise sintática

❑ Entrada:  
"cbca"

Pilha	Entrada	Ação
S\$	cbca\$	$S \rightarrow cAa$
cAa\$	cbca\$	Casar c
Aa\$	bca\$	$A \rightarrow B$
Ba\$	bca\$	$B \rightarrow bcB$
bcBa\$	bca\$	Casar b
cBa\$	ca\$	Casar c
Ba\$	a\$	$B \rightarrow \epsilon$
a\$	a\$	Casar a
\$	\$	Casar \$ $\rightarrow$ <b>aceitou</b>

Não Terminal	Símbolo de Entrada			
	a	b	c	\$
S	ERRO	ERRO	$S \rightarrow cAa$	ACEITA
A	$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow cB$	ERRO
B	$B \rightarrow \epsilon$	$B \rightarrow bcB$	ERRO	ERRO

## Exemplo 2: Conjuntos *First* e *Follow*

$E \rightarrow TE'$

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

$T \rightarrow FT'$

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

$F \rightarrow (E) \mid id$

$First(E) = First(T) = \{ (, id \}$

$First(E') = First(+) \cup First(\epsilon) = \{ +, \epsilon \}$

$First(T) = First(F) = \{ (, id \}$

$First(T') = First(*) \cup First(\epsilon) = \{ *, \epsilon \}$

$First(F) = First(() \cup First(id) = \{ (, id \}$

$Follow(E) = FIRST(,) \cup \{ \$ \} = \{ ), \$ \}$

$Follow(E') = Follow(E) = \{ ), \$ \}$

$Follow(T) = First(E') \cup Follow(E') = \{ +, ), \$ \}$

$Follow(T') = Follow(T) = \{ +, ), \$ \}$

$Follow(F) = First(T') \cup Follow(T') = \{ *, +, ), \$ \}$

# Algoritmo para construção da tabela de análise sintática

- ❑ Para cada produção  $A \rightarrow \alpha$  da gramática faça:
- 1.** Para cada terminal  $a$  em  $\text{FIRST}(A)$ , inclua  $A \rightarrow \alpha$  em  $M[A, a]$
  - 2.** Se  $\text{FIRST}(\alpha)$  inclui a palavra vazia, então adicione  $A \rightarrow \epsilon$  a  $M[A, b]$  para cada  $b$  em  $\text{FOLLOW}(A)$

Não Terminal	Símbolo de Entrada					
	Id	+	*	(	)	\$
E	$E ::= TE' \quad 1.$	ERRO	ERRO	$E ::= TE' \quad 1.$	ERRO	ACEITA
E'	ERRO	$E' ::= +TE' \quad 1.$	ERRO	ERRO	$E' ::= \epsilon \quad 2.$	$E' ::= \epsilon \quad 2.$
T	$T ::= FT' \quad 1.$	ERRO	ERRO	$T ::= FT' \quad 1.$	ERRO	ERRO
T'	ERRO	$T' ::= \epsilon \quad 2.$	$T' ::= *FT' \quad 1.$	ERRO	$T' ::= \epsilon \quad 2.$	$T' ::= \epsilon \quad 2.$
F	$F ::= id \quad 1.$	ERRO	ERRO	$F ::= (E) \quad 1.$	ERRO	ERRO

## Exercício (3)

- ❑ A partir dos conjuntos First e Follow da gramática G4, construa a tabela de análise sintática para a entrada: "abcdfcf".

G4
$S \rightarrow XYZ$
$X \rightarrow aXb \mid \epsilon$
$Y \rightarrow cYZcX \mid d$
$Z \rightarrow eZYe \mid f$

- ❑ Construa a tabela de análise sintática para as gramáticas G1, G2 e G3 (considere entradas com no mínimo 4 caracteres)