### FIRST and FOLLOW

Lecture 8 Mon, Feb 7, 2005

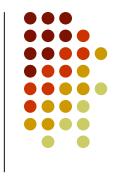


## **Left Factoring**



- A problem occurs when two productions for the same nonterminal begin with the same token.
- We cannot decide which production to use.
- This is not necessarily a problem since we could process the part they have in common, then make a decision based on what follows.

## **Left Factoring**



Consider the grammar

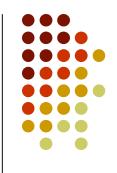
$$A \rightarrow \alpha \beta \mid \alpha \gamma$$
.

 We use left factorization to transform it into the form

$$A \rightarrow \alpha A'$$
  
 $A' \rightarrow \beta \mid \gamma$ .

 Now we can apply the productions immediately and unambiguously.

## **Example: Left Factoring**



In the earlier example, we had the productions

$$C \rightarrow id == num \mid id != num \mid id < num$$

 To perform left factoring, introduce a nonterminal C':

$$C \rightarrow id C'$$

$$C' \rightarrow == \text{num} \mid != \text{num} \mid < \text{num}$$

## **Example: Left Factoring**



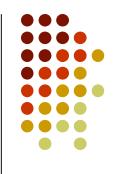
Consider the grammar of if statements.

$$S \rightarrow \text{if } C \text{ then } S \text{ else } S$$
  
| if  $C \text{ then } S$ 

We rewrite it as

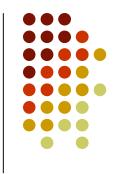
$$S \rightarrow$$
 if  $C$  then  $S S'$   
 $S' \rightarrow$  else  $S \mid \varepsilon$ .

## **LL Parsing Methods**



 LL parsing methods read the tokens from Left to right and parse them top-down according to a Leftmost derivation.

## **Table-Driven LL Parsing**



- To build the parsing table, we need the notion of nullability and the two functions
  - FIRST
  - FOLLOW

## **Nullability**



A nonterminal A is nullable if

$$A \Rightarrow^* \epsilon$$
.

Clearly, A is nullable if it has a production

$$A \rightarrow \epsilon$$
.

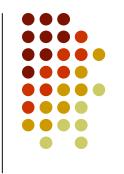
 But A is also nullable if there are, for example, productions

$$A \rightarrow BC$$
.

$$B \rightarrow A \mid aC \mid \varepsilon$$
.

$$C \rightarrow aB \mid Cb \mid \epsilon$$
.

## **Nullability**



 In other words, A is nullable if there is a production

$$A \rightarrow \varepsilon$$
,

or there is a production

$$A \rightarrow B_1 B_2 \dots B_n$$

where  $B_1$ ,  $B_2$ , ...,  $B_n$  are nullable.

## **Nullability**



In the grammar

$$E \rightarrow T E'$$
  
 $E' \rightarrow + T E' \mid \varepsilon$ .  
 $T \rightarrow F T'$   
 $T' \rightarrow * F T' \mid \varepsilon$ .  
 $F \rightarrow (E) \mid \text{id} \mid \text{num}$   
 $E' \text{ and } T' \text{ are nullable.}$ 

• E, T, and F are not nullable.

# **Summary**



Nonterminal	Nullable	
E	No	
E'	Yes	
T	No	
T'	Yes	
F	No	

#### FIRST and FOLLOW



- Given a grammar G, we may define the functions FIRST and FOLLOW on the strings of symbols of G.
  - FIRST( $\alpha$ ) is the set of all terminals that may appear as the *first* symbol in a replacement string of  $\alpha$ .
  - FOLLOW( $\alpha$ ) is the set of all terminals that may follow  $\alpha$  in a derivation.

#### **FIRST**



- For a grammar symbol X, FIRST(X) is defined as follows.
  - For every terminal X, FIRST(X) = {X}.
  - For every nonterminal X, if  $X \rightarrow Y_1 Y_2 ... Y_n$  is a production, then
    - FIRST( $Y_1$ )  $\subseteq$  FIRST(X).
    - Furthermore, if  $Y_1, Y_2, ..., Y_k$  are nullable, then FIRST( $Y_{k+1}$ )  $\subseteq$  FIRST(X).

#### **FIRST**



- We are concerned with FIRST(X) only for the nonterminals of the grammar.
- FIRST(X) for terminals is trivial.
- According to the definition, to determine FIRST(A), we must inspect all productions that have A on the left.

• Let the grammar be

$$E \rightarrow T E'$$
  
 $E' \rightarrow + T E' \mid \varepsilon$ .  
 $T \rightarrow F T'$   
 $T' \rightarrow * F T' \mid \varepsilon$ .  
 $F \rightarrow (E) \mid id \mid num$ 





- Find FIRST(E).
  - E occurs on the left in only one production  $E \rightarrow T E'$ .
  - Therefore,  $FIRST(T) \subseteq FIRST(E)$ .
  - Furthermore, T is not nullable.
  - Therefore, FIRST(*E*) = FIRST(*T*).
  - We have yet to determine FIRST(*T*).



- Find FIRST(*T*).
  - T occurs on the left in only one production  $T \rightarrow F T'$ .
  - Therefore,  $FIRST(F) \subseteq FIRST(T)$ .
  - Furthermore, F is not nullable.
  - Therefore, FIRST(*T*) = FIRST(*F*).
  - We have yet to determine FIRST(F).

- Find FIRST(F).
  - FIRST(*F*) = {(, id, num}.
- Therefore,
  - FIRST(*E*) = {(, id, num}.
  - FIRST(*T*) = {(, **id**, **num**}.



- Find FIRST(E').
  - FIRST(*E*') = {+}.
- Find FIRST(*T'*).
  - FIRST(*T'*) = {\*}.







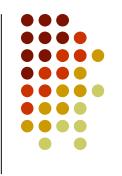
Nonterminal	Nullable	FIRST	
E	No	{(, id, num}	
E'	Yes	{+}	
T	No	{(, id, num}	
T'	Yes	{*}	
F	No	{(, id, num}	

#### **FOLLOW**



- For a grammar symbol X, FOLLOW(X) is defined as follows.
  - If S is the start symbol, then \$ ∈ FOLLOW(S).
  - If  $A \to \alpha B\beta$  is a production, then FIRST( $\beta$ )  $\subseteq$  FOLLOW(B).
  - If  $A \to \alpha B$  is a production, or  $A \to \alpha B\beta$  is a production and  $\beta$  is nullable, then FOLLOW(A)  $\subseteq$  FOLLOW(B).

#### **FOLLOW**

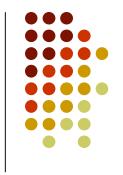


- We are concerned about FOLLOW(X) only for the nonterminals of the grammar.
- According to the definition, to determine FOLLOW(A), we must inspect all productions that have A on the right.



Let the grammar be

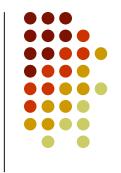
$$E \rightarrow T E'$$
  
 $E' \rightarrow + T E' \mid \varepsilon$ .  
 $T \rightarrow F T'$   
 $T' \rightarrow * F T' \mid \varepsilon$ .  
 $F \rightarrow (E) \mid id \mid num$ 



- Find FOLLOW(E).
  - E is the start symbol, therefore \$ ∈ FOLLOW(E).
  - E occurs on the right in only one production.

$$F \rightarrow (E)$$
.

• Therefore  $FOLLOW(E) = \{\$, \}$ .



- Find FOLLOW(E').
  - E'occurs on the right in two productions.

$$E \rightarrow T E'$$
  
 $E' \rightarrow + T E'$ .

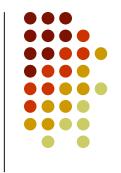
• Therefore,  $FOLLOW(E') = FOLLOW(E) = \{\$, \}$ .



- Find FOLLOW(T).
  - Toccurs on the right in two productions.

$$E \rightarrow T E'$$
  
 $E' \rightarrow + T E'$ .

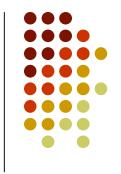
- Therefore, FOLLOW(*T*) contains FIRST(*E'*) = {+}.
- However, E' is nullable, therefore it also contains
   FOLLOW(E) = {\$, }} and FOLLOW(E') = {\$, }}.
- Therefore, FOLLOW(*T*) = {+, \$, )}.



- Find FOLLOW(T').
  - T' occurs on the right in two productions.

$$T \rightarrow F T'$$
 $T' \rightarrow * F T'$ 

• Therefore,  $FOLLOW(T) = FOLLOW(T) = \{\$, \}, +\}.$ 

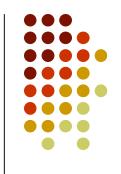


- Find FOLLOW(F).
  - Foccurs on the right in two productions.

$$T \rightarrow F T'$$
 $T' \rightarrow * F T'$ .

- Therefore, FOLLOW(F) contains FIRST(T') = {\*}.
- However, T' is nullable, therefore it also contains FOLLOW(T) = {+, \$, }} and FOLLOW(T) = {\$, }, +}.
- Therefore, FOLLOW(*F*) = {\*, \$, ), +}.

# **Summary**



Nonterminal	Nullable	FIRST	FOLLOW
E	No	{(, id, num}	{\$, )}
E'	Yes	{+}	{\$, )}
T	No	{(, id, num}	{\$, ), +}
T'	Yes	{*}	{\$, ), +}
F	No	{(, id, num}	{*, \$, ), +}

#### **Exercise**



The grammar

$$R \rightarrow R \cup R \mid RR \mid R^* \mid (R) \mid \mathbf{a} \mid \mathbf{b}$$
 generates all regular expressions on the alphabet  $\{\mathbf{a}, \mathbf{b}\}$ .

 Using the result of the exercise from the previous lecture, find FIRST(X) and FOLLOW(X) for each nonterminal X in the grammar.