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

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

- We use *left factorization* to transform it into the form

$$A \to \alpha A'$$

$$A' \to \beta \mid \gamma.$$

- Now we can apply the productions immediately and unambiguously.

Example: Left Factoring

- In the earlier example, we had the productions

  $$C \rightarrow \textbf{id == num} \mid \textbf{id != num} \mid \textbf{id < num}$$

- To perform left factoring, introduce a nonterminal C':

  $$C \rightarrow \textbf{id}\ C'$$

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

Example: Left Factoring

- Consider the grammar of if statements.

  $S \rightarrow$ **if** C **then** S **else** S

  $\quad\quad\quad | $ **if** C **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 **L**eft to right and parse them top-down according to a **L**eftmost 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

$$A \Rightarrow^* \varepsilon.$$

- Clearly, A is nullable if it has a production

$$A \rightarrow \varepsilon.$$

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

$$A \rightarrow BC.$$

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

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

Nullability

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

$$A \to \varepsilon,$$

or there is a production

$$A \to B_1 B_2 \ldots B_n,$$

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

Nullability

- In the grammar

  $E \rightarrow T\ E'$

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

  $T \rightarrow F\ T'$

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

  $F \rightarrow (E) \mid \textbf{id} \mid \textbf{num}$

  E' and *T'* 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(α) is the set of all terminals that may appear as the *first* symbol in a replacement string of α.
  - FOLLOW(α) is the set of all terminals that may *follow* α 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 $X \to Y_1Y_2\ldots Y_n$ 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*.

Example: FIRST

- Let the grammar be

$$E \to T\ E'$$

$$E' \to +\ T\ E' \mid \varepsilon.$$

$$T \to F\ T'$$

$$T' \to *\ F\ T' \mid \varepsilon.$$

$$F \to (E) \mid \mathbf{id} \mid \mathbf{num}$$

Example: FIRST

- Find FIRST(E).
  - E occurs on the left in only one production

$$E \to T\,E'.$$

  - Therefore, FIRST(T) $\subseteq$ FIRST(E).
  - Furthermore, T is not nullable.
  - Therefore, FIRST(E) = FIRST(T).
  - We have yet to determine FIRST(T).

Example: FIRST

- Find FIRST(T).
  - T occurs on the left in only one production

$$T \to F\ T'.$$

  - Therefore, FIRST(F) $\subseteq$ FIRST(T).
  - Furthermore, F is not nullable.
  - Therefore, FIRST(T) = FIRST(F).
  - We have yet to determine FIRST(F).

Example: FIRST

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

Example: FIRST

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

Summary

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 $\$ \in$ FOLLOW(S).
  - If $A \rightarrow \alpha B\beta$ is a production, then FIRST(β) $\subseteq$ FOLLOW(B).
  - If $A \rightarrow \alpha B$ is a production, or $A \rightarrow \alpha B\beta$ is a production and β 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*.

Example: FOLLOW

- Let the grammar be

$$E \to T\,E'$$

$$E' \to +\,T\,E' \mid \varepsilon.$$

$$T \to F\,T'$$

$$T' \to *\,F\,T' \mid \varepsilon.$$

$$F \to (E) \mid \textbf{id} \mid \textbf{num}$$

Example: FOLLOW

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

$$F \to (E).$$

  - Therefore FOLLOW(E) = {$,)}.

Example: FOLLOW

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

$$E \to T\ E'$$

$$E' \to +\ T\ E'.$$

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

Example: FOLLOW

- Find FOLLOW(T).
  - T occurs 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) = {+, \$,)}.

Example: FOLLOW

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

$$T \rightarrow F\ T'$$

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

  - Therefore, FOLLOW(T') = FOLLOW(T) = {\$,), +}.

Example: FOLLOW

- Find FOLLOW(F).
  - F occurs 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

$$R \to R \cup R \mid RR \mid R^* \mid (R) \mid \mathbf{a} \mid \mathbf{b}$$

  generates all regular expressions on the alphabet {**a**, **b**}.
- Using the result of the exercise from the previous lecture, find FIRST(X) and FOLLOW(X) for each nonterminal X in the grammar.