4.4.2 FIRST and FOLLOW

The construction of both top-down and bottom-up parsers is aided by two functions, FIRST and FOLLOW, associated with a grammar G. During top-down parsing, FIRST and FOLLOW allow us to choose which production to apply, based on the next input symbol. During panic-mode error recovery, sets of tokens produced by FOLLOW can be used as synchronizing tokens.

Define $FIRST(\alpha)$, where α is any string of grammar symbols, to be the set of terminals that begin strings derived from α. If $\alpha \overset{*}{\Rightarrow} \epsilon$, then ϵ is also in FIRST(α). For example, in Fig. 4.15, $A \overset{*}{\Rightarrow} c\gamma$, so c is in FIRST(A).

For a preview of how FIRST can be used during predictive parsing, consider two A-productions $A \to \alpha \mid \beta$, where FIRST(α) and FIRST(β) are disjoint sets. We can then choose between these A-productions by looking at the next input

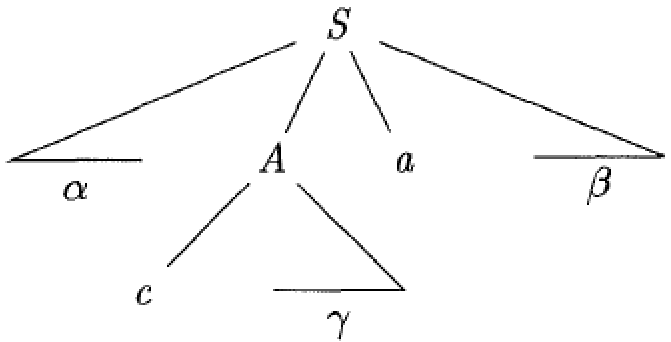

Figure 4.15: Terminal c is in $\text{FIRST}(A)$ and a is in $\text{FOLLOW}(A)$

symbol a, since a can be in at most one of $\text{FIRST}(\alpha)$ and $\text{FIRST}(\beta)$, not both. For instance, if a is in $\text{FIRST}(\beta)$ choose the production $A \to \beta$. This idea will be explored when LL(1) grammars are defined in Section 4.4.3.

Define *FOLLOW*(A), for nonterminal A, to be the set of terminals a that can appear immediately to the right of A in some sentential form; that is, the set of terminals a such that there exists a derivation of the form $S \stackrel{*}{\Rightarrow} \alpha A a \beta$, for some α and β, as in Fig. 4.15. Note that there may have been symbols between A and a, at some time during the derivation, but if so, they derived ϵ and disappeared. In addition, if A can be the rightmost symbol in some sentential form, then \$ is in FOLLOW(A); recall that \$ is a special "endmarker" symbol that is assumed not to be a symbol of any grammar.

To compute FIRST(X) for all grammar symbols X, apply the following rules until no more terminals or ϵ can be added to any FIRST set.

1. If X is a terminal, then FIRST$(X) = \{X\}$.

2. If X is a nonterminal and $X \to Y_1 Y_2 \cdots Y_k$ is a production for some $k \geq 1$, then place a in FIRST(X) if for some i, a is in FIRST(Y_i), and ϵ is in all of FIRST$(Y_1), \ldots,$ FIRST(Y_{i-1}); that is, $Y_1 \cdots Y_{i-1} \stackrel{*}{\Rightarrow} \epsilon$. If ϵ is in FIRST(Y_j) for all $j = 1, 2, \ldots, k$, then add ϵ to FIRST(X). For example, everything in FIRST(Y_1) is surely in FIRST(X). If Y_1 does not derive ϵ, then we add nothing more to FIRST(X), but if $Y_1 \stackrel{*}{\Rightarrow} \epsilon$, then we add FIRST$(Y_2)$, and so on.

3. If $X \to \epsilon$ is a production, then add ϵ to FIRST(X).

Now, we can compute FIRST for any string $X_1 X_2 \cdots X_n$ as follows. Add to $\text{FIRST}(X_1 X_2 \cdots X_n)$ all non-ϵ symbols of $\text{FIRST}(X_1)$. Also add the non-ϵ symbols of $\text{FIRST}(X_2)$, if ϵ is in $\text{FIRST}(X_1)$; the non-ϵ symbols of $\text{FIRST}(X_3)$, if ϵ is in $\text{FIRST}(X_1)$ and $\text{FIRST}(X_2)$; and so on. Finally, add ϵ to $\text{FIRST}(X_1 X_2 \cdots X_n)$ if, for all i, ϵ is in $\text{FIRST}(X_i)$.

To compute $\text{FOLLOW}(A)$ for all nonterminals A, apply the following rules until nothing can be added to any FOLLOW set.

1. Place \$ in $\text{FOLLOW}(S)$, where S is the start symbol, and \$ is the input right endmarker.

2. If there is a production $A \to \alpha B \beta$, then everything in FIRST(β) except ϵ is in FOLLOW(B).

3. If there is a production $A \to \alpha B$, or a production $A \to \alpha B \beta$, where FIRST(β) contains ϵ, then everything in FOLLOW(A) is in FOLLOW(B).

$$
\begin{array}{lcl}
E & \rightarrow & T\ E' \\
E' & \rightarrow & +\ T\ E'\ \mid\ \epsilon \\
T & \rightarrow & F\ T' \\
T' & \rightarrow & *\ F\ T'\ \mid\ \epsilon \\
F & \rightarrow & (\ E\ )\ \mid\ \mathbf{id}
\end{array}
\tag{4.28}
$$

Example 4.30: Consider again the non-left-recursive grammar (4.28). Then:

1. FIRST(F) = FIRST(T) = FIRST(E) = $\{(, \mathbf{id}\}$. To see why, note that the two productions for F have bodies that start with these two terminal symbols, **id** and the left parenthesis. T has only one production, and its body starts with F. Since F does not derive ϵ, FIRST(T) must be the same as FIRST(F). The same argument covers FIRST(E).

2. FIRST(E') = $\{+, \epsilon\}$. The reason is that one of the two productions for E' has a body that begins with terminal $+$, and the other's body is ϵ. Whenever a nonterminal derives ϵ, we place ϵ in FIRST for that nonterminal.

3. FIRST(T') = $\{*, \epsilon\}$. The reasoning is analogous to that for FIRST(E').

4. FOLLOW(E) = FOLLOW(E') = $\{), \$\}$. Since E is the start symbol, FOLLOW(E) must contain \$. The production body (E) explains why the right parenthesis is in FOLLOW(E). For E', note that this nonterminal appears only at the ends of bodies of E-productions. Thus, FOLLOW(E') must be the same as FOLLOW(E).

5. FOLLOW(T) = FOLLOW(T') = $\{+,), \$\}$. Notice that T appears in bodies only followed by E'. Thus, everything except ϵ that is in FIRST(E') must be in FOLLOW(T); that explains the symbol $+$. However, since FIRST(E') contains ϵ (i.e., $E' \overset{*}{\Rightarrow} \epsilon$), and E' is the entire string following T in the bodies of the E-productions, everything in FOLLOW(E) must also be in FOLLOW(T). That explains the symbols \$ and the right parenthesis. As for T', since it appears only at the ends of the T-productions, it must be that FOLLOW(T') = FOLLOW(T).

6. FOLLOW(F) = $\{+, *,), \$\}$. The reasoning is analogous to that for T in point (5).

First - Example

- $P \rightarrow i \mid c \mid n\ T\ S$
- $Q \rightarrow P \mid a\ S \mid b\ S\ c\ S\ T$
- $R \rightarrow b \mid \varepsilon$
- $S \rightarrow c \mid R\ n \mid \varepsilon$
- $T \rightarrow R\ S\ q$

- FIRST(P) = {i,c,n}
- FIRST(Q) = {i,c,n,a,b}
- FIRST(R) = {b,ε}
- FIRST(S) = {c,b,n,ε}
- FIRST(T) = {b,c,n,q}

First - Example

- $S \rightarrow a\ S\ e \mid S\ T\ S$
- $T \rightarrow R\ S\ e \mid Q$
- $R \rightarrow r\ S\ r \mid \varepsilon$
- $Q \rightarrow S\ T \mid \varepsilon$

- $\text{FIRST}(S) = \{a\}$
- $\text{FIRST}(R) = \{r, \varepsilon\}$
- $\text{FIRST}(T) = \{r, a, \varepsilon\}$
- $\text{FIRST}(Q) = \{a, \varepsilon\}$

Example

- $S \rightarrow a\ S\ e \mid \underline{B}$
- $B \rightarrow b\ B\ C\ f \mid \underline{C}$
- $C \rightarrow c\ C\ g \mid d \mid \varepsilon$

- FIRST(C) = $\{c,d,\varepsilon\}$
- FIRST(B) = $\{b,c,d,\varepsilon\}$
- FIRST(S) = $\{a,b,c,d,\varepsilon\}$

- FOLLOW(C) =

  $\{f,g\} \cup \text{FOLLOW}(B)$

  $= \{c,d,e,f,g,\$\}$

- FOLLOW(B) =

  $\{c,d,f\} \cup \text{FOLLOW}(S)$

  $= \{c,d,e,f,\$\}$

- FOLLOW(S) = $\{\$, e\}$

Example

- $S \rightarrow (A) \mid \varepsilon$
- $A \rightarrow T\ E$
- $E \rightarrow \&\ T\ E \mid \varepsilon$
- $T \rightarrow (\ A\ ) \mid a \mid b \mid c$

- $FIRST(T) = \{(,a,b,c\}$
- $FIRST(E) = \{\&, \varepsilon\ \}$
- $FIRST(A) = \{(,a,b,c\}$
- $FIRST(S) = \{(, \varepsilon\}$

- $FOLLOW(S) = \{\$\}$
- $FOLLOW(A) = \{\ )\ \}$
- $FOLLOW(E) = FOLLOW(A) = \{\ )\ \}$
- $FOLLOW(T) = FIRST(E) \cup FOLLOW(A) \cup FOLLOW(E) = \{\&,)\}$