Concepts of Programming Languages, CSCI 305, Fall 2021
Predict Sets EPS, FIRST, & FOLLOW and Creating Parsing Table for LL Parsing, pages 88-89, Oct. 18

EPS – epsilon predicate set – all of the non-terminals which reduce directly, or indirectly to epsilon.

Steps to create a parsing table for table-driven parsing:

Step 1. Complete an EPS, FIRST and FOLLOW table. This table has a row for each symbol: terminal and non-terminal. The sets are created using the 4 rules given near the bottom of page 88. Figure 2.24, on that page, give an algorithm for determining these values.

Step. 2 Using the EPS, FIRST and FOLLOW table, create the predict sets for the grammar using the rule:

$$\text{PREDICT}(A \rightarrow \alpha) \equiv \text{FIRST}(\alpha) \cup (\text{if EPS}(\alpha) \text{ then FOLLOW}(A) \text{ else } \Phi)$$

It is helpful to append the predict set to each production in the grammar.

Step. 3. Create the parsing table from the predict sets.

Rules for creating predict sets:
EPS(α) is true if $\alpha \Rightarrow^* \varepsilon$ and false otherwise.
Text writes: EPS(α) $\equiv$ if $\alpha \Rightarrow^* \varepsilon$ then true else false
In other words:

1. For all terminals c, EPS(c) = false.
2. For all productions α→ε, EPS(α) = true.

Do as long as more EPS values become true:

3. For production α→β, where $\beta \Rightarrow^* \varepsilon$, EPS(α) = true.

FIRST(α) is the set of all tokens that could be the start of an α.
Text writes: FIRST(α) $\equiv \{c: \alpha \Rightarrow^* c\ \beta\}$
In other words:

4. For all terminals c, FIRST(c) = c
5. For production α→β, where β begins with a terminal c, add c to FIRST(α).

Do as long as the sets keep increasing:

6. For production α→β, add FIRST(β) to FIRST(α).
7. For production α → β δ, where EPS(β)=true add FIRST(δ) to FIRST(α).

FOLLOW(α) is the set of all tokens that could come after an α in some valid program.
Text writes: FOLLOW(A) $\equiv \{c: S \Rightarrow^+ \alpha A c \beta\}$
In other words:

8. For production α→β where β contains BC add FIRST(C) to FOLLOW(B). (B and C may be terminals or non-terminals)

Do as long as the sets keep increasing:

9. For production α→β, where β ends with symbol B (terminal or non-terminal), add FOLLOW(α) to FOLLOW(B).
10. For production α→β, where β ends with variables BC, but $C \Rightarrow^* \varepsilon$, add FOLLOW(α) to FOLLOW(B).