بسم الله الرحمن الرحيم

Parsing:
Top-Down Parsing, Recursive Descent & Predictive Parser & LL(1)

Compilers - 002 - S08

Recall …

Continue …

How to write tests for selecting the appropriate production rule?

Basic Tools:

First: Let α be a string of grammar symbols. First(α) is the set that includes every terminal that appears leftmost in α or in any string originating from α.

NOTE: If $\alpha \Rightarrow \varepsilon$, then ε is First(α).

Follow: Let A be a non-terminal. Follow(A) is the set of terminals a that can appear directly to the right of A in some sentential form. ($S \Rightarrow \alpha A a \beta$, for some α and β).

NOTE: If $S \Rightarrow \alpha A$, then \$ is Follow(A).

Computing First Sets

Definition

$$\text{First}(X) = \{ t \mid X \to^* t\alpha \} \cup \{ \varepsilon \mid X \to^* \varepsilon \}$$

Intuition:

1. $\text{First}(t) = \{ t \}$
2. $\varepsilon \in \text{First}(X)$
   - if $X \to \varepsilon$
   - if $X \to A_1 \ldots A_n$ and $\varepsilon \in \text{First}(A_i)$ for $1 \le i \le n$
3. $\text{First}(\alpha) \subseteq \text{First}(X)$ if $X \to A_1 \ldots A_n\ \alpha$
   - and $\varepsilon \in \text{First}(A_i)$ for $1 \le i \le n$

Computing First Sets

- Compute First(X):
  - initialize:
    - if T is a terminal symbol then First (T) = {T}
    - if T is non-terminal then First(T) = { }
  - Calculate if e in First(X) for all X
  - while First(X) changes (for any X) do
    - for all X and all rules (X:= ABC...) do
      - First (X) := First(X) U First (ABC...)
        where First(ABC...) := F1 U F2 U F3 U ... and
        - » F1 := First (A)
        - » F2 := First (B), if A is Nullable; emptyset otherwise
        - » F3 := First (C), if A is Nullable & B is Nullable; emp...
        - » ...

First Sets. Example

- Recall the grammar

$$E \to T X \qquad\qquad X \to + E \mid \varepsilon$$
$$T \to (E) \mid \text{int } Y \qquad\qquad Y \to * T \mid \varepsilon$$

- First sets

First(( ) = { (}

First( )) = {) }

First(int) = { int }

First(+) = { + }

First(*) = { * }

First(T) = {int, (}

First(E) = {int, (}

First(X) = {+, ε }

First(Y) = {*, ε }

Computing Follow Sets

- Definition:

$$\text{Follow}(X) = \{ t \mid S \to^* \beta\ X\ t\ \delta \}$$

- Intuition
  - If $X \to A\ B$ then $\text{First}(B) \subseteq \text{Follow}(A)$ and $\text{Follow}(X) \subseteq \text{Follow}(B)$
- if $B \to^* \varepsilon$ then $\text{Follow}(X) \subseteq \text{Follow}(A)$

  - If S is the start symbol then $\$ \in \text{Follow}(S)$

Computing Follow Sets (Cont.)

Algorithm sketch:

1. $\$ \in Follow(S)$
2. $First(\beta) - \{\varepsilon\} \subseteq Follow(X)$
   - For each production $A \to \alpha\ X\ \beta$
3. $Follow(A) \subseteq Follow(X)$
   - For each production $A \to \alpha\ X\ \beta$ where $\varepsilon \in First(\beta)$

Computing Follow Sets

- Follow(X) is computed iteratively
  - base case:
    - initially, we assume nothing in particular follows X
      - (when computing, Follow (X) is initially { })
  - inductive case:
    - if (Y := s1 X s2) for any strings s1, s2 then
      - Follow (X) U= First (s2)
    - if (Y := s1 X s2) for any strings s1, s2 then
      - Follow (X) U= Follow(Y), if s2 is Nullable