CSE443 Compilers

Dr. Carl Alphonce
alphonce@buffalo.edu
343 Davis Hall

Phases of a compiler

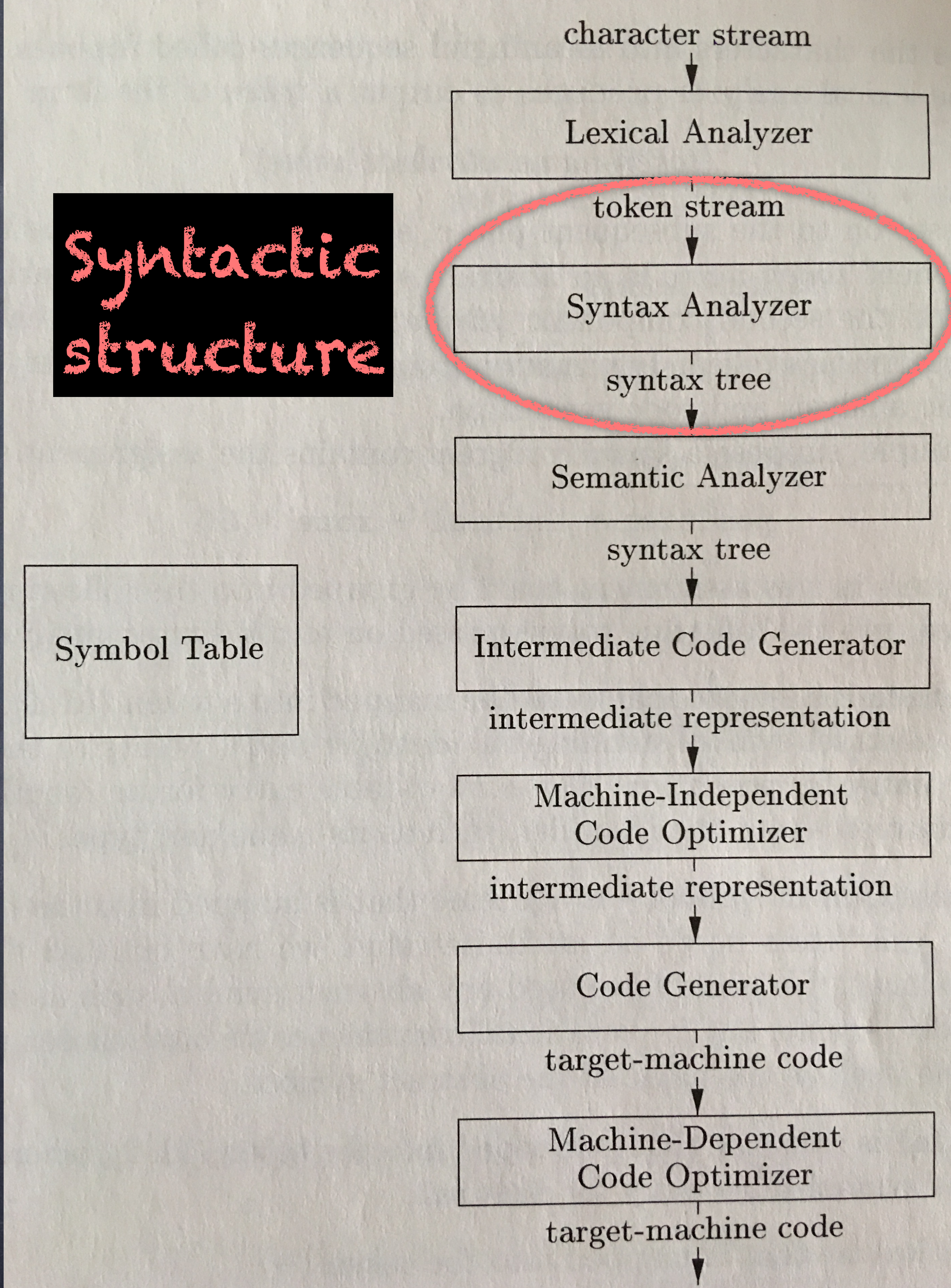

Figure 1.6, page 5 of text

Left factoring

- If two (or more) rules share a prefix then their FIRST sets do not distinguish between rule alternatives.
- If there is a choice point later in the rule, rewrite rule by factoring common prefix
- Example: rewrite

$$A \rightarrow \alpha\ \beta_1 \mid \alpha\ \beta_2$$

- as

$$A \rightarrow \alpha\ A'$$

$$A' \rightarrow \beta_1 \mid \beta_2$$

Predictive parsing:

a special case of recursive-descent parsing that does not require backtracking

Each non-terminal A ∈ N has an associated procedure:

```
void A() {
    choose an A-production A -> X1 X2 ... Xk
    for (i = 1 to k) {
        if (xi ∈ N) {
            call xi()
        }
        else if (xi = current input symbol) {
            advance input to next symbol
        }
        else error
    }
}
```

Predictive parsing:

a special case of recursive-descent parsing that does not require backtracking

Each non-terminal A ∈ N has an associated procedure:

```
void A() {
    choose an A-production A -> X1 X2 ... Xk
    for (i = 1 to k) {
        if (xi ∈ N) {
            call xi()
        }
        else if (xi = current input symbol) {
            advance input to next symbol
        }
        else error
    }
}
```

There is non-determinism in choice of production. If "wrong" choice is made the parser will need to revisit its choice by backtracking.

A predictive parser can always make the correct choice here.

FIRST(X)

- if $X \in T$ then $FIRST(X) = \{ X \}$
- if $X \in N$ and $X \rightarrow Y_1 Y_2 \ldots Y_k \in P$ for $k \geq 1$, then
  - add $a \in T$ to $FIRST(X)$ if $\exists i$ s.t. $a \in FIRST(Y_i)$ and $\varepsilon \in FIRST(Y_j)$ $\forall j < i$ (i.e. $Y_1 Y_2 \ldots Y_{i-1} \Rightarrow^* \varepsilon$)
  - if $\varepsilon \in FIRST(Y_j)$ $\forall j <= k$ add ε to $FIRST(X)$
  - if $X \rightarrow \varepsilon \in P$, then add ε to $FIRST(X)$

FOLLOW(X)

- Place \$ in FOLLOW(S), where S is the start symbol (\$ is an end marker)
- if $A \rightarrow \alpha B \beta \in P$, then $FIRST(\beta) - \{\varepsilon\}$ is in FOLLOW(B)
- if $A \rightarrow \alpha B \in P$ or $A \rightarrow \alpha B \beta \in P$ where $\varepsilon \in FIRST(\beta)$, then everything in FOLLOW(A) is in FOLLOW(B)

Table-driven predictive parsing Algorithm 4.32 (p. 224)

- INPUT: Grammar G = (N,T,P,S)
- OUTPUT: Parsing table M
- For each production A -> α of G:

1. For each terminal a ∈ FIRST(α), add A -> α to M[A,a]
2. If ε ∈ FIRST(α), then for each terminal b in FOLLOW(A), add A -> α to M[A,b]
3. If ε ∈ FIRST(α) and $ ∈ FOLLOW(A), add A -> α to M[A,$]