

# 第三章：语法分析

## 递归下降法



# 1. 递归下降法的基本原理

有文法:

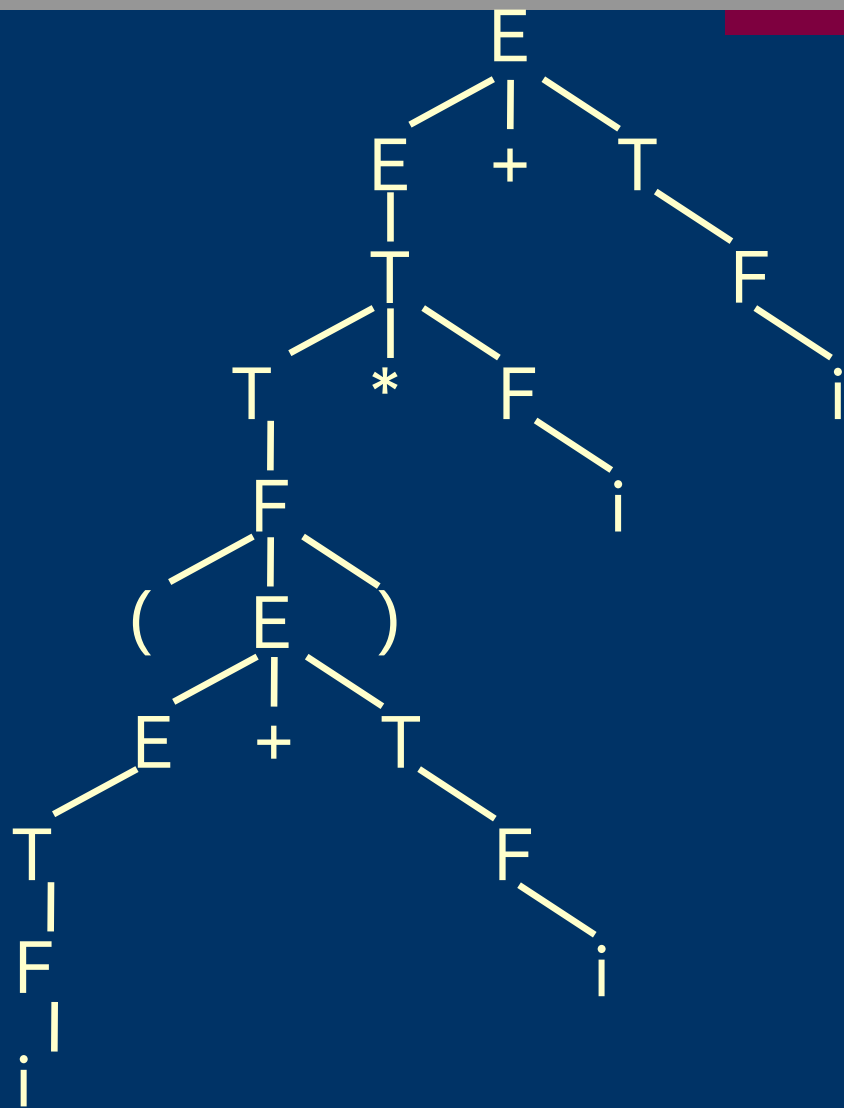
$E \rightarrow E+T \mid T$

$T \rightarrow T * F \mid F$

$F \rightarrow (E) \mid i$

有句子:

$(i+i)*i+i$



## 1.2 对文法的要求

- 为了保证推导的唯一性，对文法的要求与LL(1)文法相同。
  - 1. 文法不能是直接左递归的
  - 2. 以一个非终极符为左部的任意规则的predict集，交集为空。

## 2. 语法分析程序的构造

### □ 两个标准函数

1. ReadToken: 把输入流的头符读入变量 token 中

2. Match(a): if token=a then ReadToken  
else 出错

## 2. 语法分析程序的构造

有规则:  $\text{Stm} \rightarrow \text{while Exp do Stm}$

对应产生式右部的语法分析程序部分如下:

```
begin
  Match($while);
  Exp ();
  Match($do);
  Stm ();
end
```

## 2. 语法分析程序的构造

while  $x > y$  do if  $x > z$  then  $x := x + y$  else  $x := y$

Begin Match(\$while); Exp; Match(\$do); Stm  
End

## 2. 语法分析程序的构造

当产生式形如:  $A \rightarrow \beta_1 | \beta_2 | \dots | \beta_n$ , 则按下面的方法编写子程序A:

```
procedure A( )  
begin if token  $\in$  Predict( $A \rightarrow \beta_1$ ) then  $\theta(\beta_1)$  else  
      if token  $\in$  Predict( $A \rightarrow \beta_2$ ) then  $\theta(\beta_2)$  else  
      .....  
      if token  $\in$  Predict( $A \rightarrow \beta_n$ ) then  $\theta(\beta_n)$  else  
      error( )  
end
```

其中对  $\beta_i = X_1 X_2 \dots X_n$ ,  $\theta(\beta_i) = \theta'(X_1); \theta'(X_2); \dots; \theta'(X_n)$ ;

如果  $X \in V_N$ ,  $\theta'(X) = X()$ ;

如果  $X \in V_T$ ,  $\theta'(X) = \text{Match}(X)$ ; // 即  $\text{if}(\text{token} == X) \text{ReadToken}()$ ;

如果  $X = \varepsilon$ ,  $\theta(\varepsilon) = \text{skip}(\text{空语句})$ .

## 2. 语法分析程序的构造

□ 主程序:

```
void main(){  
    ReadToken(); S();  
    if (token=='#') 成功;  
    else 失败  
}
```



## 2. 语法分析程序的构造

### □ 具体构建流程

给定一个文法G

1. 求每条规则的Predict集
2. 写针对每个非终极符的函数
3. 写主函数

**优点:**  
构造简单

**缺点:**  
1. 频繁的函数调用影响效率  
2. 程序比较长

# 构造分析程序的例子

【1】  $E \rightarrow TE'$

【2】  $E' \rightarrow +TE'$

【3】  $E' \rightarrow \varepsilon$

【4】  $T \rightarrow FT'$

【5】  $T' \rightarrow *FT'$

【6】  $T' \rightarrow \varepsilon$

【7】  $F \rightarrow ( E )$

【8】  $F \rightarrow i$

{i, (}

{+}

{#, )}

{i, (}

{\*}

{+, #, )}

{(}

{i}

$(i+i)*i$

# 构造分析程序的例子

## □ 例子

假设有文法

$$Z \rightarrow a B a$$

$$B \rightarrow b B$$

$$B \rightarrow c$$

对于串abbca做验证

### 3.编译程序的自动生成

- 固定部分先构造好，如驱动程序
- 可变部分根据文法规则来求
- 两部分组合在一起