

第五章 自顶向下语法分析方法

5.1 自顶向下分析的一般过程和问题

5.2 递归下降分析法

5.3 LL(1)文法及其分析程序

- FIRST和FOLLOW集定义和计算
- LL(1) 文法定义
- LL(1)分析程序实现

非LL(1)文法的改造

句型的分析

分析算法分类：

自上而下分析法：

从文法的开始符号出发，反复使用各种产生式，寻找与输入符号匹配的最左推导。

自下而上分析法：

从输入符号串开始，逐步进行归约（最右推导的逆过程），直至归约到文法的开始符号。

5.1自上而下语法分析的一般过程

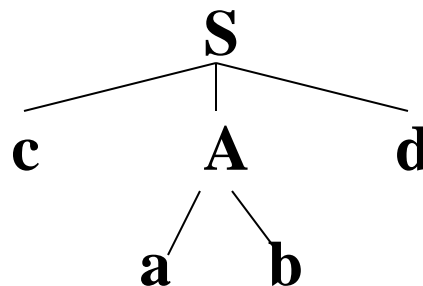
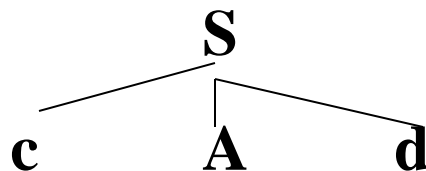
从文法的开始符号出发，反复使用各种产生式，寻找与输入符号匹配的最左推导。

文法G: $S \rightarrow cAd$ $A \rightarrow ab \mid a$

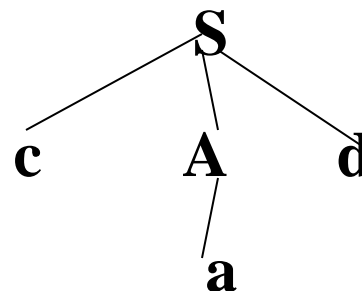
识别输入串 $w=cad$

试探: 推导过程: $S \Rightarrow cAd \Rightarrow cabd$

S



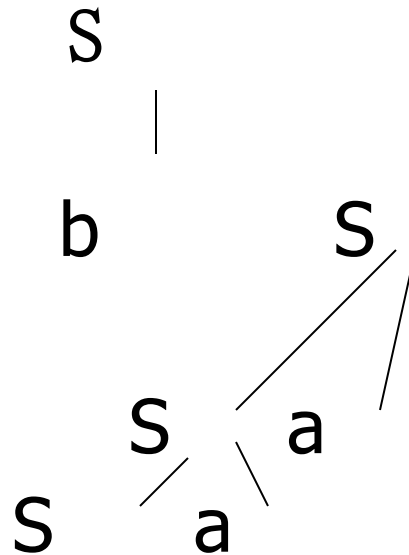
回溯: 试探推导过程: $S \Rightarrow cAd \Rightarrow cad$



自上而下语法分析的问题

$G[S]:$ $S \rightarrow Sa$
 $S \rightarrow b$ $L = \{ba^n \mid n \geq 1\}$

$W = baaa$



左递归 关于非终结符P的规则

直接左递归 若 $P \rightarrow P \alpha \mid \beta$

$\alpha, \beta \in V^*$ 且 β 不以 P 开头

间接左递归 若 $P \Rightarrow^* P \alpha$

$S \rightarrow Aa$

$A \rightarrow Sb$

$A \rightarrow b$

消除左递归

消除直接左递归

形如: $P \rightarrow P \alpha \mid \beta$

α 非 ϵ , β 不以 P 打头

改写为: $P \rightarrow \beta P'$

$P' \rightarrow \alpha P' \mid \epsilon$

消除左递归

例: $E \rightarrow E+T \mid T$

$T \rightarrow T*F \mid F$

$F \rightarrow (E) \mid i$

G[E]: (1) $E \rightarrow TE'$ (2) $E' \rightarrow +TE' \mid \varepsilon$

(3) $T \rightarrow FT'$ (4) $T' \rightarrow *FT' \mid \varepsilon$

(5) $F \rightarrow (E) \mid i$

消除左递归

一般情况:

$$A \rightarrow A \alpha_1 \mid A \alpha_2 \mid \dots \mid A \alpha_m \mid \beta_1 \mid \beta_2 \mid \dots \mid \beta_n$$

消除左递归后:

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

$$A' \rightarrow \alpha_1 A' \mid \alpha_2 A' \mid \dots \mid \alpha_m A' \mid \varepsilon$$

消除间接左递归要求文法:

1.无回路($A \xRightarrow{+} A$) 2.无空产生式

(1) 以某种顺序将文法非终结符排列 $A_1, A_2 \dots A_n$

(2) **for i:=1 to n do**

begin

for j:=1 to i-1 do

用 $A_i \rightarrow \alpha_1 \mid \alpha_2 r \dots \mid \alpha_k r$ 替代

形如 $A_i \rightarrow A_j r$ 的规则,其中

$A_j \rightarrow \alpha_1 \mid \alpha_2 \dots \mid \alpha_k$ 是关于 A_j 的全部产生式;

消除 A_i 规则的直接左递归;

end;

(3) 化简由2得到的文法

消除间接左递归

例：文法

$$\begin{array}{l} S \rightarrow Ac \mid c \\ B \rightarrow Sa \mid a \end{array} \quad A \rightarrow Bb \mid b$$

回溯问题

例: $S \rightarrow iEtS \mid iEtSeS \mid a$ $E \rightarrow b$

判断句子ibtaea

提取左公因子: $A \rightarrow \alpha \beta_1 \mid \alpha \beta_2$

变为: $A \rightarrow \alpha A'$

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

若 $A \rightarrow \alpha \beta_1 \mid \alpha \beta_2 \mid \dots \mid \alpha \beta_n \mid r$

变为: $A \rightarrow \alpha A' \mid r$

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

例: $S \rightarrow iEtS \mid iEtSeS \mid a$ $E \rightarrow b$

提取左公因子: $S \rightarrow iEtSS' \mid a$

$$S' \rightarrow eS \mid \varepsilon$$
$$E \rightarrow b$$

5.2 递归下降分析法

递归下降分析器：在消除了左递归和提取左公因子后，可以构造一个不带回溯的自上而下的分析程序。

这个程序由一组递归过程组成，每个过程对应文法的一个非终结符。

例：**G[E]:**

$$(1) \quad E \rightarrow TE' \quad (2) \quad E' \rightarrow +TE' \mid \varepsilon$$

$$(3) \quad T \rightarrow FT' \quad (4) \quad T' \rightarrow *FT' \mid \varepsilon$$

$$(5) \quad F \rightarrow (E) \mid i$$

Procedure E;

```
  Begin
    T;E'
  End;
```

Procedure T;

```
  Begin
    F;T'
  End;
```

Procedure F;

```
  IF sym= 'i' then advance
  else
    IF sym= '(' then
      Begin
        advance;
        E
      IF sym= ')' then advance
      else ERROR
    End;
  else ERROR;
```

Procedure E' ;

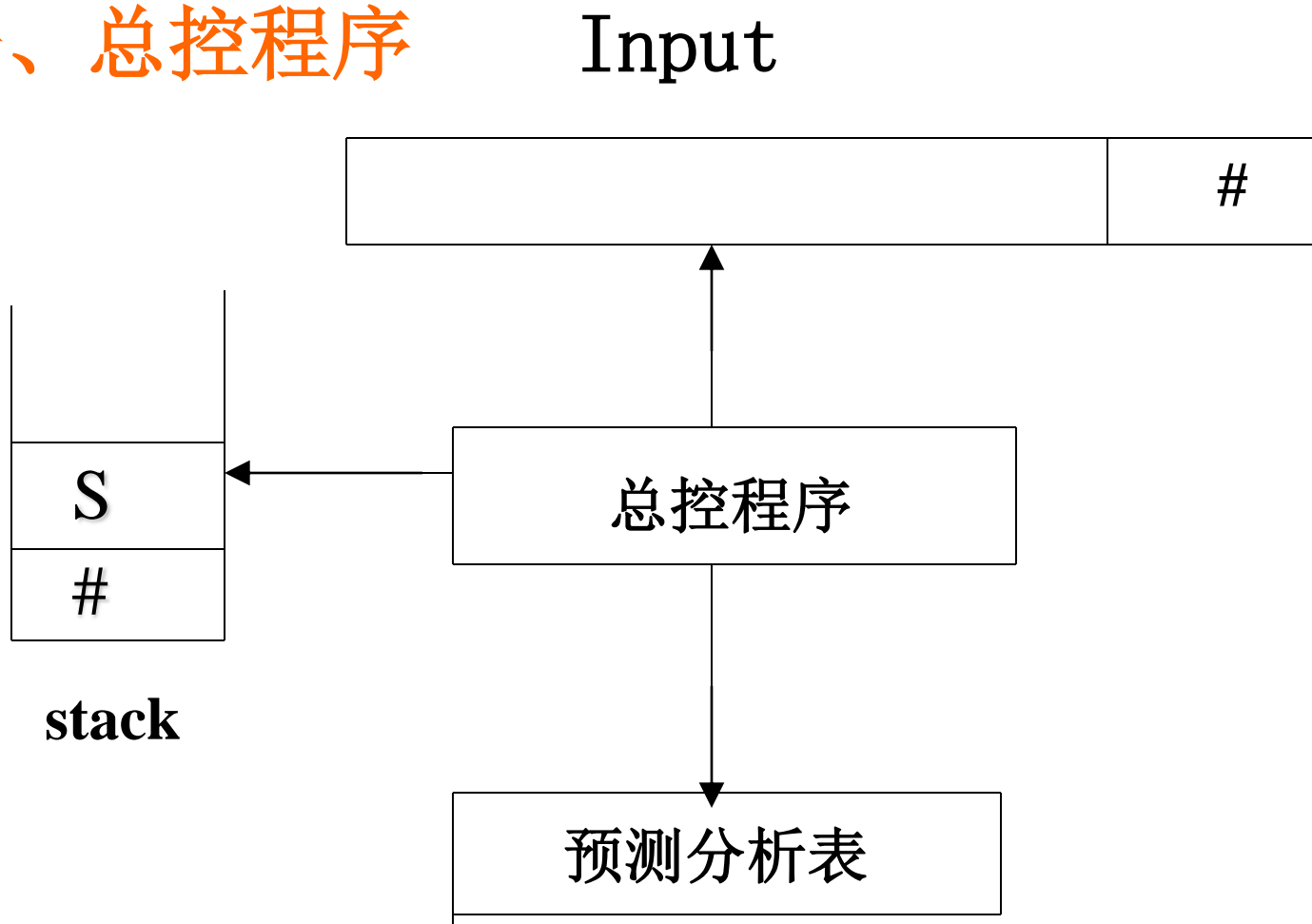
```
  IF sym= '+' then
    Begin
      advance;
      T;E'
    End;
```

Procedure T' ;

```
  IF sym= '*' then
    Begin
      advance;
      F;T'
    End;
```

5.3非递归的预测分析方法（LL（1））

一、总控程序



分析算法

BEGIN

首先把' #'然后把文法开始符号推入栈; 把第一个输入符号读进a; FLAG:=TRUE;

WHILE FLAG DO

BEGIN

把栈顶符号上托出去并放在X中;

IF $X \in V_t$ THEN IF $X=a$ THEN

把下一个输入符号读进a

ELSE ERROR

ELSE IF $X=\#$ THEN

IF $X=a$ THEN FLAG:=FALSE

ELSE ERROR

ELSE IF $M[X, a] = \{X \rightarrow X_1 X_2 \dots X_k\}$

THEN 把 X_k, X_{k-1}, \dots, X_1 一一推进栈

ELSE ERROR

END OF WHILE;

STOP/*分析成功, 过程完毕*/

END

例: $G[E]$: (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	+	*	()	#
E	(1)			(1)		
E'		(2)			(3)	(3)
T	(4)			(4)		
T'		(6)	(5)		(6)	(6)
F	(8)			(7)		

对输入 $i+i*i$ 的分析



二、预测分析表M的构造

---LL (1) 文法及其分析程序

FIRST集和FOLLOW集

设 $G=(V_T, V_N, S, P)$ 是上下文无关文法

$$\text{FIRST}(\alpha) = \{a \mid \alpha \xRightarrow{*} a\beta, a \in V_T, \alpha, \beta \in V^*\}$$

$$\text{FOLLOW}(A) = \{a \mid S \Rightarrow \cdots Aa\cdots, a \in V_T\}$$

计算FIRST集

1. 若 $X \in V_T$, 则 $\text{FIRST}(X) = \{X\}$
2. 若 $X \in V_N$, 且有产生式 $X \rightarrow a...$, 则把 a 加入到 $\text{FIRST}(X)$ 中; 若 $X \rightarrow \varepsilon$ 也是一条产生式, 则把 ε 也加到 $\text{FIRST}(X)$ 中.
3. 若 $X \rightarrow Y...$ 是一个产生式且 $Y \in V_N$, 则把 $\text{FIRST}(Y)$ 中的所有非 ε 元素都加到 $\text{FIRST}(X)$ 中; 若 $X \rightarrow Y_1 Y_2 \dots Y_K$ 是一个产生式, Y_1, Y_2, \dots, Y_{i-1} ($1 \leq i \leq k$) 都是非终结符, 而且, 当

$Y_1 \dots Y_{i-1} \xRightarrow{*} \varepsilon$, 则把 $\text{FIRST}(Y_{1 \leq j \leq i-1})$ 中所有非 ε 元素, $\text{FIRST}(Y_i)$ 都加到 $\text{FIRST}(X)$ 中; 特别是, 若所有的 $\text{FIRST}(Y_j, j=1, 2, \dots, K)$ 均含有 ε , 则把 ε 加到 $\text{FIRST}(X)$ 中.

例: $G[E]$:

- | | |
|--------------------------------|--|
| (1) $E \rightarrow TE'$ | (2) $E' \rightarrow +TE' \mid \varepsilon$ |
| (3) $T \rightarrow FT'$ | (4) $T' \rightarrow *FT' \mid \varepsilon$ |
| (5) $F \rightarrow (E) \mid i$ | |

计算FOLLOW集

1. 对于文法的开始符号 S , 置 $\#$ 于 $FOLLOW(S)$ 中;
2. 若 $A \rightarrow \alpha B \beta$ 是一个产生式, 则把 $FIRST(\beta) \setminus \{\epsilon\}$ 加至 $FOLLOW(B)$ 中;
3. 若 $A \rightarrow \alpha B$ 是一个产生式, 或 $A \rightarrow \alpha B \beta$ 是一个产生式而 $\beta \xRightarrow{*} \epsilon$ (即 $\epsilon \in FIRST(\beta)$), 则把 $FOLLOW(A)$ 加至 $FOLLOW(B)$ 中.

例: $G[E]$ (见上页)

预测分析表构造算法

1. 对文法G的每个产生式 $A \rightarrow \alpha$ 执行第二步和第三步;
2. 对每个终结符 $a \in \text{FIRST}(\alpha)$, 把 $A \rightarrow \alpha$ 加至 $M[A, a]$ 中,
3. 若 $\epsilon \in \text{FIRST}(\alpha)$, 则对任何 $b \in \text{FOLLOW}(A)$ 把 $A \rightarrow \alpha$ 加至 $M[A, b]$ 中,
4. 把所有无定义的 $M[A, a]$ 标上“出错标志”。

上例的预测分析表

三、LL (1) 文法:

一个文法**G**，若它的分析表**M**不含多重定义入口，则称为**LL (1)** 文法。

例:

文法 $S \rightarrow iEtS \mid iEtSeS \mid a$ $E \rightarrow b$

不是**LL (1)** 文法。

一个**LL (1)** 文法是无二义的。

一个文法G是LL(1)的，当且仅当对于G的每一个非终结符A的任何两个不同产生式 $A \rightarrow \alpha \mid \beta$ ，下面的条件成立：

1. $\text{FIRST}(\alpha) \cap \text{FIRST}(\beta) = \emptyset$ ，也就是 α 和 β 推导不出以同一个终结符a为首的符号串；它们不应该都能推出空字 ϵ 。

2. 假若 $\beta \Rightarrow^* \epsilon$ ，那么，

$\text{FIRST}(\alpha) \cap \text{FOLLOW}(A) = \emptyset$ 。也就是，若 $\beta \Rightarrow^* \epsilon$ ，则 α 所能推出的串的首符号不应在 $\text{FOLLOW}(A)$ 中。

结论：

LL(1)文法是无二义的。

非LL(1)文法的改造

消除左递归
提左公因子