
编译原理

- 第一章 编译程序概述
 - 第二章 PL/0编译程序的实现
 - 第三章 文法和语言
 - 第四章 词法分析
 - 第五章 自顶向下语法分析方法
 - 第六章 自底向上优先分析方法
 - 第七章 LR分析方法
 - 第八章 语法制导翻译和中间代码生成
 - 第九章 符号表
 - 第一〇章 代码优化
 - 第十一章 代码生成
-

第5章 自顶向下语法分析方法

- 一、确定的自顶向下分析思想
 - 二、LL(1)文法的判别
 - 三、某些非LL(1)文法到LL(1)文法等价变换
 - 四、不确定的自顶向下分析思想
 - 五、确定的自顶向下分析方法
-

五、确定的自顶向下分析方法

1、递归子程序法：

基本思想是：对应文法中每个非终结符编写一个递归过程，每个过程的功能是识别由该非终结符推出的串。

要求文法满足LL(1)文法。是比较简单直观易于构造的一种语法分析方法。

PL/0编译程序的语法分析部分就是采用的递归子程序法。

例：递归子程序实现 表达式的语法分析

表达式的EBNF

$\langle \text{表达式} \rangle ::= [+|-] \langle \text{项} \rangle \{ (+|-) \langle \text{项} \rangle \}$
 $\langle \text{项} \rangle ::= \langle \text{因子} \rangle \{ (*|/) \langle \text{因子} \rangle \}$
 $\langle \text{因子} \rangle ::= \underline{\text{ident}} | \underline{\text{number}} | ' (\langle \text{表达式} \rangle) '$

• $\langle \text{表达式} \rangle ::= [+|-] \langle \text{项} \rangle \{ (+|-) \langle \text{项} \rangle \}$

```
- procedure expr;  
- begin  
  
-   if sym in [ plus, minus ] then  
-   begin  
-     getsym; term;  
-   end  
-   else term;  
  
-   while sym in [plus, minus] do  
-   begin  
-     getsym; term;  
-   end  
  
- end;
```

$\langle \text{项} \rangle ::= \langle \text{因子} \rangle \{ (*|/) \langle \text{因子} \rangle \}$

- Procedure term;
 - begin
 - factor;
 - while sym in [times, slash] do
 - begin
 - getsym;
 - factor;
 - end
 - end;
-

〈因子〉 ::= ident | number | ‘ (’ 〈表达式〉 ‘) ’

- Procedure factor;
- begin
- if sym=ident
- then getsym
- else
- if sym=number
- then getsym
- else
- if sym=‘ (’
- then begin
- getsym;
- expr;
- if sym=‘) ’
- then getsym
- else error
- end
- else error
- end;

五、确定的自顶向下分析方法

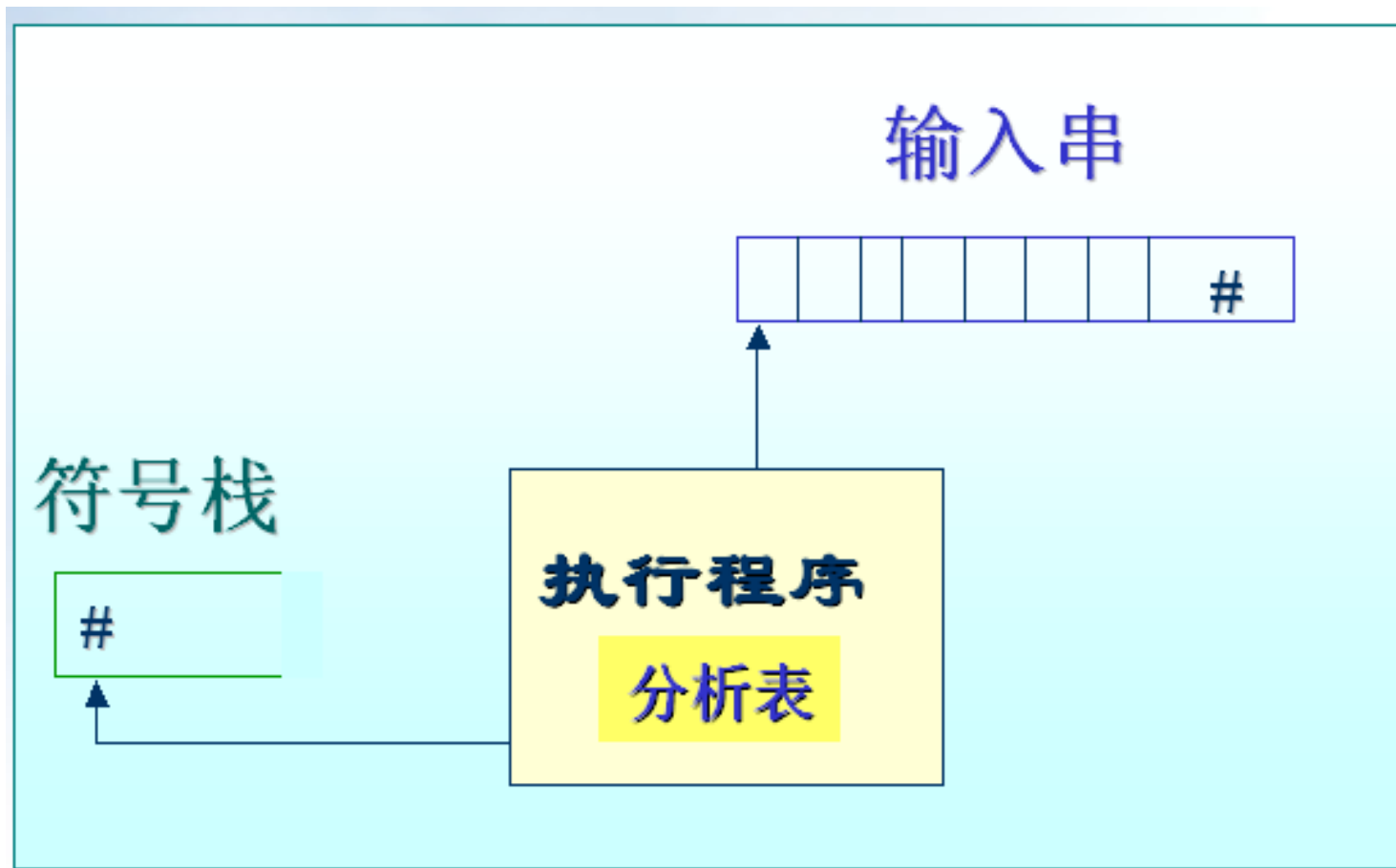
2、预测分析方法：

自顶向下分析的另一种方法

(1)预测分析器的组成：

{ 预测分析程序
先进后出栈
预测分析表—与文法有关

表驱动预测分析程序模型



-
- 预测分析表可用一个矩阵表示。
 - 矩阵元素 $M[A,a]$ 中的 A 表示非终结符， a 表示终结符或句子结束符 $\#$ ，矩阵元素 $M[A,a]$ 中的内容是一条关于 A 的产生式，表明当用非终结符 A 向下推导时，面临输入符 a 时，所采用的候选产生式；
 - 当元素内容无产生式时，表明用 A 为左部向下推导时遇到了不该出现的符号，因此元素内容为转向出错处理。
-

(2)分析过程:

X: 当前栈顶符号
a: 当前输入符号

‘#’‘S’进栈，当前终结符送a

上托栈顶符号放入X

若产生式为
 $X \rightarrow x_1 x_2 \dots x_n$
按逆序即
 $x_n \dots x_2 x_1$ 入栈

M(X,a)是
产生式吗?

error

$X \in V_T?$

y

$X = a?$

y

读下一个
符号

n

$X = \#?$

y

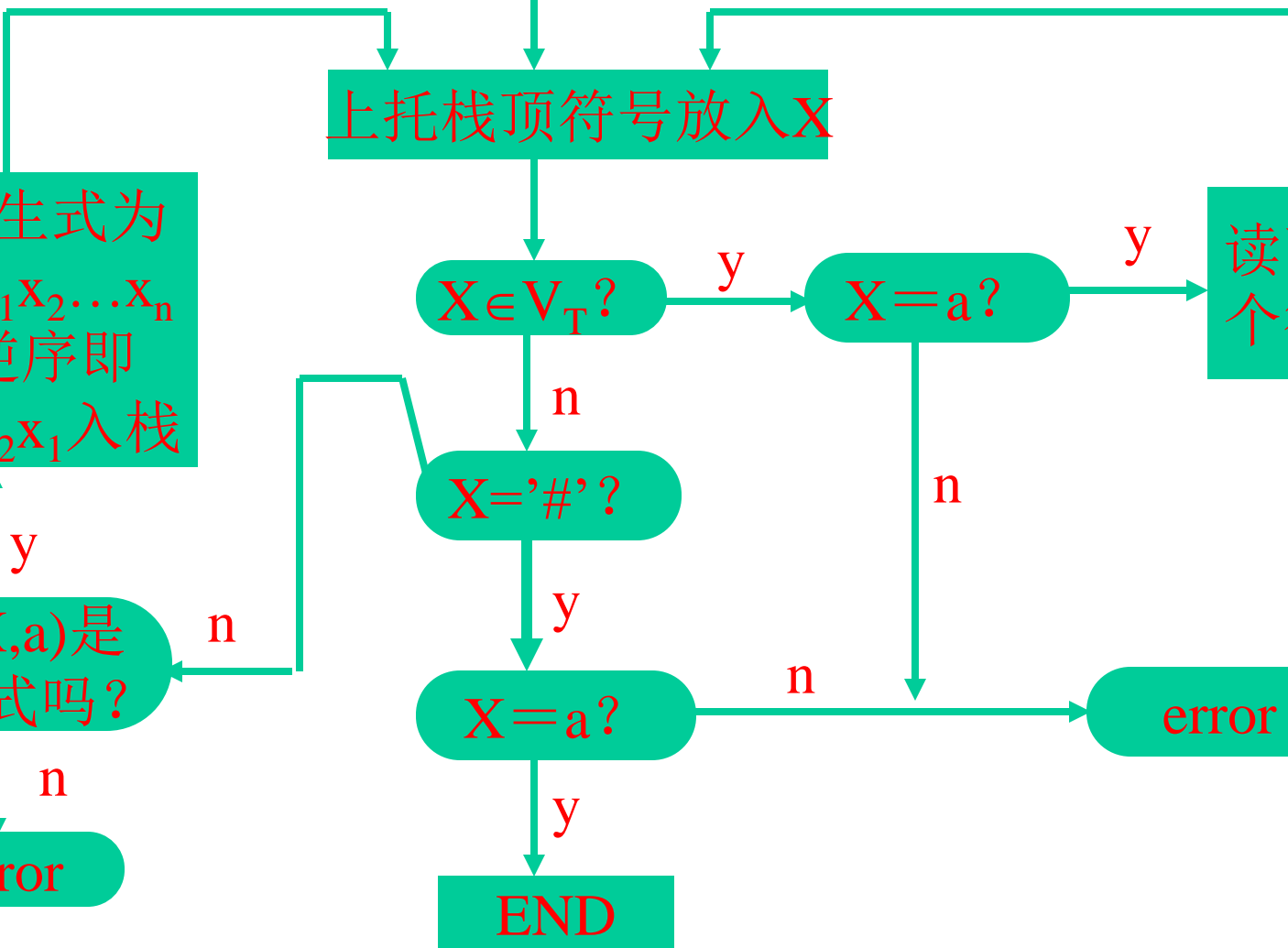
$X = a?$

n

error

y

END



执行程序主要实现如下操作:

1.把#和文法开始符号S推进栈,并读入输入串的第一个符a,重复下述过程直到正常结束或出错.

2.测定栈顶符号X和当前输入符号a,执行如下操作:

- (1)若 $X=a=\#$, 分析成功, 停止。S匹配输入串成功.
 - (2)若 $X=a\neq\#$, 把X推出栈, 再读入下一个符号。
 - (3)若 $X\in V_n$, 查分析表M (详细步骤见下页)
-

(3)若 $X \in V_n$, 查分析表M。

a) $M[X, a] = X ::= UVW$

则将X弹出栈, 将UVW压入

注: U在栈顶 (最左推导)

b) $M[X, a] = \text{error}$ 转出错处理

c) $M[X, a] = X ::= \epsilon$ ---a为X的后继符号

则将X弹出栈 (不读下一符号)

继续分析。

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 $M[X, a]=\{X \rightarrow X_1X_2 \dots X_K\}$
- THEN 把 X_K, X_{K-1}, \dots, X_1 ——推进栈
- ELSE ERROR
- END OF WHILE;
- STOP/*分析成功，过程完毕*/
- END

-
- 预测分析表可用一个矩阵表示。矩阵元素 $M[A,a]$ 中的 A 表示非终结符， a 表示终结符或句子结束符 $\#$ ，矩阵元素 $M[A,a]$ 中的内容是一条关于 A 的产生式，表明当用非终结符 A 向下推导时，面临输入符 a 时，所采用的候选产生式，当元素内容无产生式时，表明用 A 为左部向下推导时遇到了不该出现的符号，因此元素内容为转向出错处理。
 - 如何构造预测分析表？
 - 如何根据预测分析表构造自动的分析程序？
-

如何构造预测分析表?

基本思想是:

当语法中某一**非终结符**呈现在**栈顶**时,根据当前的**输入符号**,分析表应指示要用该**非终结符**里的哪一条规则去匹配输入串(即进行下一步最左推导)

根据这个思想,我们不难把构造分析表算法构造出来!

终结符号

非
终
结
符
号

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;
2. 对每个终结符 $a \in \text{SELECT}(A \rightarrow \alpha)$, 把 $A \rightarrow \alpha$ 加至 $M[A, a]$ 中,
3. 所有无定义的 $M[A, a]$ 标上“出错标志”。

可以证明, 一个文法 G 的预测分析表不含多重入口, 当且仅当该文法是LL(1)的

(3)实例分析：给定文法，构造预测分析表，并针对输入串*i+i*i#*构造预测分析过程。

例：文法为： $E \rightarrow E+T \mid T$

$T \rightarrow T * F \mid F$

$F \rightarrow i \mid (E)$

步骤：

(1) 判断文法是否为LL(1)文法。

✓如果文法中含有左递归，必须先消除左递归：

(2) 构造预测分析表： $\text{Select}(A \rightarrow \alpha)$

(3) 列出预测分析过程

(3)实例分析：给定文法，构造预测分析表，并针对输入串*i+i*i*#构造预测分析过程。

例：文法为： $E \rightarrow E+T \mid T$

$T \rightarrow T * F \mid F$

$F \rightarrow i \mid (E)$

$S \rightarrow Sa$
 $S \rightarrow b$

$\left\{ \begin{array}{l} S \rightarrow bS' \\ S' \rightarrow aS' \mid \epsilon \end{array} \right.$

$E \rightarrow E+T$

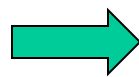
$E \rightarrow T$

构造步骤有：

① 判断文法是否为LL(1)文法。

✓ 由于文法中含有左递归，所以必须先消除左递归：

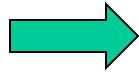
$E \rightarrow E+T \mid T$



$E \rightarrow TE'$

$E' \rightarrow +TE' \mid \epsilon$

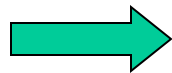
$$\mathbf{T \rightarrow T * F \mid F}$$



$$\mathbf{T \rightarrow FT'}$$

$$\mathbf{T' \rightarrow *FT' \mid \varepsilon}$$

文法变为:



$$\mathbf{E \rightarrow TE'}$$

$$\mathbf{E' \rightarrow +TE' \mid \varepsilon}$$

$$\mathbf{T \rightarrow FT'}$$

$$\mathbf{T' \rightarrow *FT' \mid \varepsilon}$$

$$\mathbf{F \rightarrow i \mid (E)}$$

一个文法符号串的first集合计算方法:

如果文法符号串 $\alpha \in V^*$, $\alpha = X_1 X_2 \dots X_n$,

1、当 $X_1 \xRightarrow{*} \varepsilon$, 则 $\text{first}(\alpha) = \text{first}(X_1)$

2、当对任何 j ($1 \leq j \leq i-1$, $2 \leq i \leq n$), $\varepsilon \in \text{first}(X_j)$

则 $\text{first}(\alpha) = (\text{first}(X_1) - \{\varepsilon\}) \cup (\text{first}(X_2) - \{\varepsilon\})$
 $\cup \dots \cup (\text{first}(X_{i-1}) - \{\varepsilon\}) \cup \text{first}(X_i)$

3、当 $\text{first}(X_j)$ 都含有 ε 时 ($1 \leq j \leq n$), 则

$\text{first}(\alpha) = \text{first}(X_1) \cup \text{first}(X_2) \cup \dots \cup \text{first}(X_j) \cup \{\varepsilon\}$

构造FOLLOW集合的算法

- (1) 若S为开始符号,则把“#”加入FOLLOW(S)中
 - (2) 若 $A \rightarrow \alpha B \beta$ ($\beta \neq \epsilon$),则把 $\text{FIRST}(\beta) - \{\epsilon\}$ 加入FOLLOW(B)
 - (3) 若 $A \rightarrow \alpha B$ 或 $A \rightarrow \alpha B \beta$, 且 $\beta \Rightarrow \epsilon$ 则把FOLLOW(A)加入FOLLOW(B)
-

✓ 求First集合:

$\text{First}(E) = \{ (, i \}$

$\text{First}(E') = \{ +, \varepsilon \}$

$\text{First}(T) = \{ (, i \}$

$\text{First}(T') = \{ *, \varepsilon \}$

$\text{First}(F) = \{ (, i \}$

✓ 求Follow集:

$\text{Follow}(E) = \{), \# \}$

$\text{Follow}(E') = \{), \# \}$

$\text{Follow}(T) = \{ +,), \# \}$

$\text{Follow}(T') = \{ +,), \# \}$

$\text{Follow}(F) = \{ *, +,), \# \}$

$\because E \rightarrow TE'$
 $T \not\Rightarrow \varepsilon, E' \not\Rightarrow \varepsilon$
 $\therefore \text{First}(E) = \text{first}(T)$
 $= \text{first}(F) = \{ (, i \}$

$E \Rightarrow TE' \Rightarrow FT'E' \Rightarrow (E)T'E'$

$E \Rightarrow TE' \Rightarrow FT'E' \Rightarrow (E)T'E'$
 $\Rightarrow (TE')T'E'$

$E \Rightarrow TE' \Rightarrow T+TE' \Rightarrow T+T$
 $\Rightarrow T+FT' \Rightarrow T+(E)T' \Rightarrow$
 $T+(TE')T' \Rightarrow T+(T)T'$

$E \Rightarrow TE' \Rightarrow FT'E' \Rightarrow FT'$
 $\Rightarrow F*FT' \Rightarrow F*(E)T'$
 $\Rightarrow F*(TE')T' \Rightarrow F*(FT'E')T'$
 $\Rightarrow F*(FT')T'$

求Select集:

$$\text{Select}(E \rightarrow TE') = \{ (, i \}$$
$$\text{Select}(E' \rightarrow +TE') = \{ + \}$$
$$\text{Select}(E' \rightarrow \varepsilon) = \{), \# \}$$
$$\text{Select}(T \rightarrow FT') = \{ (, i \}$$
$$\text{Select}(T' \rightarrow *FT') = \{ * \}$$
$$\text{Select}(T' \rightarrow \varepsilon) = \{ +,), \# \}$$
$$\text{Select}(F \rightarrow i) = \{ i \}$$
$$\text{Select}(F \rightarrow (E)) = \{ (\}$$

由上可知有相同左部产生式的Select集合的交集为空, 所以文法是LL(1)

②构造预测分析表：

方法：对每个终结符或#用a表示。

若 $a \in \text{Select}(A \rightarrow a)$ ，则 $A \rightarrow a$ 放入 $M[A, a]$ 中。

	i	+	*	()	#
E	$\rightarrow TE'$			$\rightarrow TE'$		
E'		$\rightarrow +TE'$			$\rightarrow \varepsilon$	$\rightarrow \varepsilon$
T	$\rightarrow FT'$			$\rightarrow FT'$		
T'		$\rightarrow \varepsilon$	$\rightarrow *FT'$		$\rightarrow \varepsilon$	$\rightarrow \varepsilon$
F	$\rightarrow i$			$\rightarrow (E)$		

③ 对于某句子的分析过程：

下面用预测分析程序，栈和预测分析表对输入串 $i+i*i\#$ 进行分析，给出栈的变化过程如下：

步骤	分析栈	剩余输入串	所用产生式
1	#E	$i+i*i\#$	$E \rightarrow TE'$
2	#E'T	$i+i*i\#$	$T \rightarrow FT'$
3	#E'T'F	$i+i*i\#$	$F \rightarrow i$
4	#E'T'i	$i+i*i\#$	i匹配
5	#E'T'	$+i*i\#$	$T' \rightarrow \varepsilon$
6	#E'	$+i*i\#$	$E' \rightarrow +TE'$

7	#E'T+	+i*i#	+匹配
8	#E'T	i*i#	T→FT'
9	#E'T'F	i*i#	F→i
10	#E'T'i	i*i#	i匹配
11	#E'T'	*i#	T'→*FT'
12	#E'T'F*	*i#	*匹配
13	#E'T'F	i#	F→i
14	#E'T'i	i#	i匹配
15	#E'T'	#	T'→ε
16	#E'	#	E'→ε
17	#	#	接受

非LL(1)文法

- 二义性文法肯定不是LL(1)文法，例如：

$$G[S]: S \rightarrow iEtSS' \mid a \quad S' \rightarrow eS \mid \varepsilon \quad E \rightarrow b$$

- $\text{Follow}(S') = \{e, \$\}$
- 其相应的分析表如下表

	a	b	e	i	t	\$
S	$S \rightarrow a$			$S \rightarrow iEtSS'$		
S'			$S' \rightarrow eS$ $S' \rightarrow \varepsilon$			$S' \rightarrow \varepsilon$
E		$E \rightarrow b$				

例题分析

• P96 例题

例题 1 已知文法 $G[S]: S \rightarrow aH$

$$H \rightarrow aMd \mid d$$
$$M \rightarrow Ab \mid \epsilon$$
$$A \rightarrow aM \mid \epsilon$$

1. 判断 $G[S]$ 是否为 LL(1) 文法, 若是, 请构造相应的 LL(1) 预测分析表。
2. 若 $G[S]$ 是 LL(1) 文法, 请给出对输入串 $aaabd\#$ 的预测分析过程, 并说明该输入串是否是 $G[S]$ 的句子。

1. 首先计算文法中能推出‘ ϵ ’的非终结符和它们的 FIRST 集和 FOLLOW 集。

(1) 检查 $G[S]$ 的每个产生式可得到非终结符 M 能推出‘ ϵ ’。

(2) 非终结符的 FIRST 集和 FOLLOW 集如表 5.5 所示。

表 5.5 FIRST 集和 FOLLOW 集

非终结符	FIRST 集	FOLLOW 集
S	{a}	{#}
H	{a,d}	{#}
M	{a,e, ϵ }	{d,b}
A	{a,e}	{b}

(3) $G[S]$ 是否为 LL(1) 文法的判断。

检查相同左部产生式 select 集的交集是否为‘ \emptyset ’。

由于 $\text{select}(H \rightarrow aMd) \cap \text{select}(H \rightarrow d) = \{a\} \cap \{d\} = \emptyset$

$\text{select}(M \rightarrow Ab) \cap \text{select}(M \rightarrow \epsilon) = \{a,e\} \cap \{d,b\} = \emptyset$

$\text{select}(A \rightarrow aM) \cap \text{select}(A \rightarrow e) = \{a\} \cap \{e\} = \emptyset$

所以该文法是 LL(1) 文法。

(4) 构造的 LL(1)预测分析表如表 5.6 所示。

表 5.6 LL(1)分析表

	<i>a</i>	<i>d</i>	<i>b</i>	<i>e</i>	#
<i>S</i>	$\rightarrow aH$				
<i>H</i>	$\rightarrow aMd$	$\rightarrow d$			
<i>M</i>	$\rightarrow Ab$	$\rightarrow \epsilon$	$\rightarrow \epsilon$	$\rightarrow Ab$	
<i>A</i>	$\rightarrow aM$			$\rightarrow e$	

2. 对输入串 *aaabd* # 的预测分析过程如下：

步骤	分析栈	剩余输入串	推导用产生式或匹配
1	# <i>S</i>	<i>aaabd</i> #	$S \rightarrow aH$
2	# <i>Ha</i>	<i>aaabd</i> #	' <i>a</i> ' 匹配
3	# <i>H</i>	<i>aabd</i> #	$H \rightarrow aMd$
4	# <i>dMa</i>	<i>aabd</i> #	' <i>a</i> ' 匹配
5	# <i>dM</i>	<i>abd</i> #	$M \rightarrow Ab$
6	# <i>dbA</i>	<i>abd</i> #	$A \rightarrow aM$
7	# <i>dbMa</i>	<i>abd</i> #	' <i>a</i> ' 匹配
8	# <i>dbM</i>	<i>bd</i> #	$M \rightarrow \epsilon$
9	# <i>db</i>	<i>bd</i> #	' <i>b</i> ' 匹配
10	# <i>d</i>	<i>d</i> #	' <i>d</i> ' 匹配
11	#	#	分析成功

例题分析

• P97 例题3

例题 3 判断文法 $G[S]$: $S \rightarrow Ab | Ba$

$A \rightarrow aA | a$

$B \rightarrow a$

是 LL(1) 的吗? 若不是, 请改写为等价的 $G'[S]$, 证明改写后的文法是否为 LL(1) 的。

表 5.7 预测分析表

	b	a	#
S		$\rightarrow Ab$	
		$\rightarrow Ba$	
A		$\rightarrow aA'$	
A'	$\rightarrow \epsilon$	$\rightarrow A$	
B		$\rightarrow a$	

本章要求掌握的内容

- 给定文法G，判断是否是LL(1)文法
 - 非LL(1)文法到LL(1)文法的转换（提取左公因子和消除左递归）
 - 构造LL(1)文法的预测分析表，写出给定输入串的预测分析过程
-

课后作业

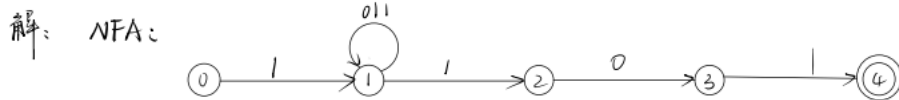
- 上次作业：
 - P100 第2题 (1) 、 (2)
 - P101 第7题 (1) (2) (3)

 - P99 1
-

第四音作业

- P72 : 1 (1)
 , 2, 5
- P73: 8, 9

1. 构造下列正规式相应的 DFA:
 (1) $1(0|1)^*101$



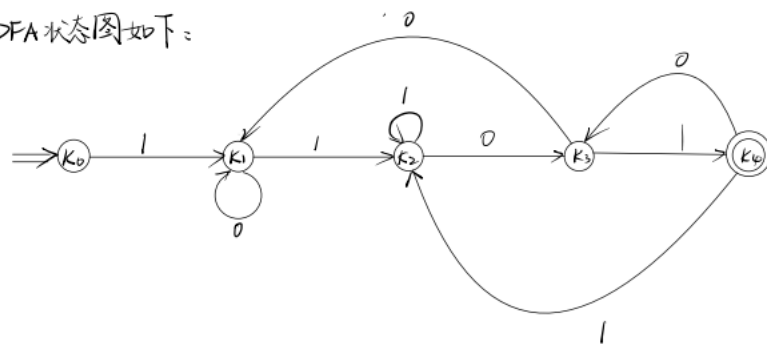
NFA \rightarrow DFA:

- $K_0 = \{0\}$
- $move(K_0, 1) = \{1\}$
- $K_1 = \{1\}$
- $move(K_1, 1) = \{1, 2\}$
- $move(K_1, 0) = \{1\}$
- $K_2 = \{1, 2\}$
- $move(K_2, 1) = \{1, 2\}$
- $move(K_2, 0) = \{1, 3\}$
- $K_3 = \{1, 3\}$
- $move(K_3, 1) = \{1, 2, 4\}$
- $move(K_3, 0) = \{1\}$
- $K_4 = \{1, 2, 4\}$
- $move(K_4, 1) = \{1, 2\}$
- $move(K_4, 0) = \{1, 3\}$

DFA矩阵表示如下:

K	$move(K, 0)$	$move(K, 1)$	
$K_0 = \{0\}$		K_1	0
$K_1 = \{1\}$	K_1	K_2	0
$K_2 = \{1, 2\}$	K_3	K_2	0
$K_3 = \{1, 3\}$	K_1	K_4	0
$K_4 = \{1, 2, 4\}$	K_3	K_2	1

DFA状态图如下:



第四章作业

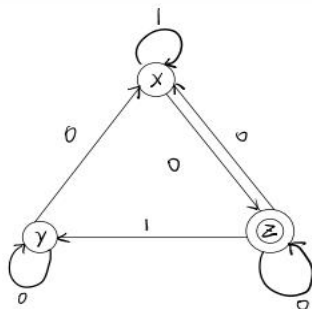
• P72 : 1 (1)

, 2, 5

• P73: 8, 9

2. 已知 $NFA = (\{x, y, z\}, \{0, 1\}, M, \{x\}, \{z\})$, 其中: $M(x, 0) = \{z\}$, $M(y, 0) = \{x, y\}$, $M(z, 0) = \{x, z\}$, $M(x, 1) = \{x\}$, $M(y, 1) = \emptyset$, $M(z, 1) = \{y\}$, 构造相应的 DFA.

解: NFA:



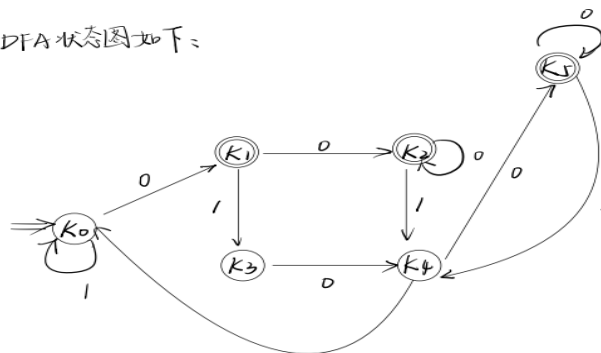
NFA \rightarrow DFA:

$K_0 = \{x\}$
 $move(K_0, 0) = \{z\}$
 $move(K_0, 1) = \{x\}$
 $K_1 = \{z\}$
 $move(K_1, 0) = \{x, z\}$
 $move(K_1, 1) = \{y\}$
 $K_2 = \{x, z\}$
 $move(K_2, 0) = \{x, z\}$
 $move(K_2, 1) = \{x, y\}$
 $K_3 = \{y\}$
 $move(K_3, 0) = \{x, y\}$
 $move(K_3, 1) = \emptyset$
 $K_4 = \{x, y\}$
 $move(K_4, 0) = \{x, y, z\}$
 $move(K_4, 1) = \{x\}$
 $K_5 = \{x, y, z\}$
 $move(K_5, 0) = \{x, y, z\}$
 $move(K_5, 1) = \{x, y\}$

DFA矩阵表示如下:

K	move(K, 0)	move(K, 1)
$K_0 = \{x\}$	K_1	K_0
$K_1 = \{z\}$	K_2	K_3
$K_2 = \{x, z\}$	K_2	K_4
$K_3 = \{y\}$	K_4	
$K_4 = \{x, y\}$	K_5	K_0
$K_5 = \{x, y, z\}$	K_5	K_4

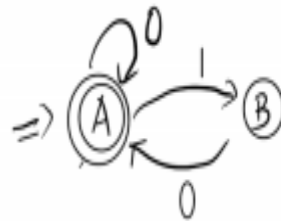
DFA状态图如下:



第四章作业

- P72 : 1 (1)
 , 2, 5
- P73: 8, 9

5. 构造一个 DFA, 它接收 $\Sigma = \{0, 1\}$ 上所有满足如下条件的字符串:
每一个 1 都有 0 直接跟在右边。然后再构造该语言的正规文法。



正规式: $(0|10)^*$

正规文法: A 即为起始状态, 也为终止状态.

$A \rightarrow 0$ $A \rightarrow 1B$ $A \rightarrow 0A$
 $B \rightarrow 0$ $B \rightarrow 0A$

第四章作业

- P72 : 1 (1)
 , 2, 5
- P73: 8, 9

8. 给出下述文法所对应的正规式

$$S \rightarrow 0A \mid 1B$$

$$A \rightarrow 1S \mid 1$$

$$B \rightarrow 0S \mid 0$$

$$S \rightarrow 0A$$

$$\rightarrow 01S \rightarrow 01$$

$$S \rightarrow 1B$$

$$\rightarrow 10S \rightarrow 10$$

$$S = (01 \mid 10)^*$$

9. 将图 4.22 的 DFA 最小化, 并用正规式描述它所识别的语言。

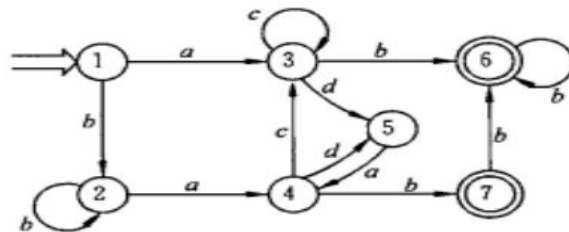
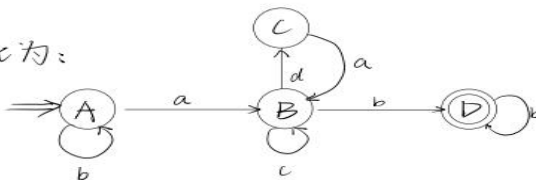


图 4.22

解: $\{1, 2, 3, 4, 5\}$ 为不可接受状态, $\{6, 7\}$ 为可接受状态
 用 b 分割 $\{1, 2, 3, 4, 5\}$, 得到 $\{1, 2, 5\}$ 和 $\{3, 4\}$
 用 a 分割 $\{1, 2, 5\}$ 得到 $\{1, 2\}$ 和 $\{5\}$
 令 $A = \{1, 2\}$ $B = \{3, 4\}$ $C = \{5\}$ $D = \{6, 7\}$

最小化为:



正规式为:

$$b^* a (c \mid da)^* b^+$$