

！练习4.4.12：图4-24中给出了对应于某些语句的文法。你可以将  $e$  和  $s$  当作分别代表条件表达式和“其他语句”的终结符号。如果我们按照下列方法来解决因为展开可选“else”（非终结符号  $stmtTail$ ）而引起的冲突：当从输入中看到一个 **else** 时就选择消耗掉这个 **else**。使用4.4.5节中描述的同步符号的思想：

1) 为这个文法构造一个带有错误纠正信息的预测分析表。

2) 给出你的语法分析器在处理下列输入时的行为：

- Ⓐ **if  $e$  then  $s$  ; if  $e$  then  $s$  end**
- Ⓑ **while  $e$  do begin  $s$  ; if  $e$  then  $s$  ; end**

$stmt$	$\rightarrow$	if $e$ then $stmt$ $stmtTail$
	$ $	while $e$ do $stmt$
	$ $	begin $list$ end
	$ $	$s$
$stmtTail$	$\rightarrow$	else $stmt$
	$ $	$\epsilon$
$list$	$\rightarrow$	$stmt$ $listTail$
$listTail$	$\rightarrow$	; $list$
	$ $	$\epsilon$

图4-24 某种类型语句的文法

(1) 构造一个带有错误纠正信息的预测分析表

使用 FIRST 和 FOLLOW 符号作为同步集合，使用 synch 来表示根据相应非终结符号的 FOLLOW 集合得到的同步词法单元。

FIRST 函数：

计算各个文法符号  $X$  的  $FIRST(X)$  时，不断应用下列规则，直到再没有新的终结符号或  $\epsilon$  可以被加入到任何 FIRST 集合中为止。

1) 如果  $X$  是一个终结符号，那么  $FIRST(X) = X$ 。

2) 如果  $X$  是一个非终结符号，且  $X \rightarrow Y_1 Y_2 \dots Y_k$  是一个产生式，其中  $k \geq 1$ ，那么如果对于某个  $i$ ， $a$  在  $FIRST(Y_i)$  中且  $\epsilon$  在所有的  $FIRST(Y_1)$ 、 $FIRST(Y_2)$ 、 $\dots$ 、 $FIRST(Y_{i-1})$  中，就把  $a$  加入到  $FIRST(X)$  中。也就是说， $Y_1 \dots Y_{i-1} \Rightarrow \epsilon$ 。如果对于所有的  $j = 1, 2, \dots, k$ ， $\epsilon$  在  $FIRST(Y_j)$  中，那么将  $\epsilon$  加入到  $FIRST(X)$  中。比如， $FIRST(Y_1)$  中的所有符号一定在  $FIRST(X)$  中。如果  $Y_1$  不能推导出  $\epsilon$ ，那么我们就不会再向  $FIRST(X)$  中加入任何符号，但是如果  $Y_1 \Rightarrow \epsilon$ ，那么我们就加上  $FIRST(Y_2)$ ，依此类推。

3) 如果  $X \rightarrow \epsilon$  是一个产生式，那么将  $\epsilon$  加入到  $FIRST(X)$  中。

FOLLOW 函数：

1) 将  $\$$  放到  $FOLLOW(S)$  中，其中  $S$  是开始符号，而  $\$$  是输入右端的结束标记。

2) 如果存在一个产生式  $A \rightarrow \alpha B \beta$ ，那么  $FIRST(\beta)$  中除  $\epsilon$  之外的所有符号都在  $FOLLOW(B)$  中。

3) 如果存在一个产生式  $A \rightarrow \alpha B$ ，或存在产生式  $A \rightarrow \alpha B \beta$  且  $FIRST(\beta)$  包含  $\epsilon$ ，那么  $FOLLOW(A)$  中的所有符号都在  $FOLLOW(B)$  中。

由文法  $G$  构造预测分析表  $M$  的方法：

方法：对于文法  $G$  的每个产生式  $A \rightarrow \alpha$ ，进行如下处理：

1) 对于  $FIRST(\alpha)$  中的每个终结符号  $a$ ，将  $A \rightarrow \alpha$  加入到  $M[A, a]$  中。

2) 如果  $\epsilon$  在  $FIRST(\alpha)$  中，那么对于  $FOLLOW(A)$  中的每个终结符号  $b$ ，将  $A \rightarrow \alpha$  加入到  $M[A, b]$  中。如果  $\epsilon$  在  $FIRST(\alpha)$  中，且  $\$$  在  $FOLLOW(A)$  中，也将  $A \rightarrow \alpha$  加入到  $M[A, \$]$  中。

在完成上面的操作之后，如果  $M[A, a]$  中没有产生式，那么将  $M[A, a]$  设置为 **error**（我们通常在表中用一个空条目表示）。 □

预测分析表												
非终端符号	输入符号											
	if	e	then	while	do	begin	end	s	else	;	\$	
stmt	①			②		③	synch	④	synch	synch	synch	
stmtTail							⑥		⑤	⑥	⑥	
list	⑦			⑦		⑦	synch	⑦				
listTail							⑨			⑧		

文法对应 9 个产生式如右图，依次分析如下

①  $stmt \rightarrow \text{if } e \text{ then } stmt \text{ stmtTail}$

$\text{FIRST}(\text{if } e \text{ then } stmt \text{ stmtTail}) = \text{FIRST}(\text{if}) = \{\text{if}\}$

$M[stmt, \text{if}] = ①$

②  $stmt \rightarrow \text{while } e \text{ do } stmt$

$\text{FIRST}(\text{while } e \text{ do } stmt) = \text{FIRST}(\text{while}) = \{\text{while}\}$

$M[stmt, \text{while}] = ②$

③  $stmt \rightarrow \text{begin } list \text{ end}$

$\text{FIRST}(\text{begin } list \text{ end}) = \text{FIRST}(\text{begin}) = \{\text{begin}\}$

$M[stmt, \text{begin}] = ③$

④  $stmt \rightarrow s$

$\text{FIRST}(s) = \{s\}$

$M[stmt, s] = ④$

⑤  $stmtTail \rightarrow \text{else } stmt$

$\text{FIRST}(\text{else } stmt) = \text{FIRST}(\text{else}) = \{\text{else}\}$

$M[stmtTail, \text{else}] = ⑤$

⑥  $stmtTail \rightarrow \epsilon$

$\text{FIRST}(\epsilon) = \{\epsilon\}$

需求得  $\text{FOLLOW}(stmtTail)$ ，对其中每个终结符号  $b$ ，将⑥加入到  $M[stmtTail, b]$

由  $stmt \rightarrow \text{if } e \text{ then } stmt \text{ stmtTail}$ ，在所有产生式中， $stmtTail$  只出现在  $stmt$  产生式的尾部，可知：

$\text{FOLLOW}(stmt) = \text{FOLLOW}(stmtTail)$

由产生式可知：

a)  $\text{FIRST}(stmtTail)$  除  $\epsilon$  之外的所有符号都在  $\text{FOLLOW}(stmt)$  中

$\text{FIRST}(stmtTail) = \{\text{else}, \epsilon\}$

$\text{FOLLOW}(stmt) = \{\text{else}\}$

同时， $stmt$  是开始符号， $\text{FOLLOW}(stmt) = \{\text{else}, \$\}$

b)  $\text{FIRST}(listTail)$  除  $\epsilon$  之外的所有符号都在  $\text{FOLLOW}(stmt)$  中

$\text{FIRST}(listTail) = \{;, \epsilon\}$

$\epsilon$  在  $\text{FIRST}(listTail)$  中， $\text{FOLLOW}(listTail)$  中所有符号都在  $\text{FOLLOW}(stmt)$  中

由⑦⑧  $\text{FOLLOW}(list) = \text{FOLLOW}(listTail) = \{\text{end}\}$

$\text{FOLLOW}(stmt) = \{\text{else}, \$, ;, \text{end}\}$  (最终完整结果)

综上可得： $\text{FOLLOW}(stmtTail) = \{\text{else}, \$, ;, \text{end}\}$

表中  $M[stmtTail, \text{else}] = ⑤$  位置不变，其余对应  $\text{FOLLOW}(stmtTail)$  中终结符号位置的均插入⑥

⑦  $list \rightarrow stmt \text{ listTail}$

$\text{FIRST}(stmt \text{ listTail}) = \text{FIRST}(stmt) = \{\text{if}, \text{while}, \text{begin}, s\}$

- ①  $stmt \rightarrow \text{if } e \text{ then } stmt \text{ stmtTail}$
- ②  $stmt \rightarrow \text{while } e \text{ do } stmt$
- ③  $stmt \rightarrow \text{begin } list \text{ end}$
- ④  $stmt \rightarrow s$
- ⑤  $stmtTail \rightarrow \text{else } stmt$
- ⑥  $stmtTail \rightarrow \epsilon$
- ⑦  $list \rightarrow stmt \text{ listTail}$
- ⑧  $listTail \rightarrow ; \text{ list}$
- ⑨  $listTail \rightarrow \epsilon$

表中相应四处插入⑦

⑧  $listTail \rightarrow ; list$

$FIRST(; list) = \{;\}$

$M[listTail, ;] = \textcircled{8}$

⑨  $listTail \rightarrow \epsilon$

$FIRST(\epsilon) = \{\epsilon\}$

需求得  $FOLLOW(listTail)$ ，对其中每个终结符号  $b$ ，将⑨加入到  $M[listTail, b]$

由⑥中过程得到：

$FOLLOW(list) = FOLLOW(listTail) = \{end\}$

$M[listTail, end] = \textcircled{9}$

由恐慌模式的错误恢复过程，要根据 FOLLOW 函数结果把 synch 插入表中相应位置。

$FOLLOW(stmt) = FOLLOW(stmtTail) = \{else, \$, ,, end\}$

$FOLLOW(list) = FOLLOW(listTail) = \{end\}$

之前已经正确插入产生式的位置，不再插入 synch。

(2) 给出你的语法分析器在处理下列输入时的行为

预测分析表处理输入的预测分析过程：

```

设置 ip 使它指向 w 的第一个符号，其中 ip 是输入指针；
令 X = 栈顶符号；
while ( X ≠ $ ) { /* 栈非空 */
    if ( X 等于 ip 所指向的符号 a ) 执行栈的弹出操作，将 ip 向前移动一个位置；
    else if ( X 是一个终结符号 ) error();
    else if ( M[X, a] 是一个报错条目 ) error();
    else if ( M[X, a] = X → Y1Y2...Yk ) {
        输出产生式 X → Y1Y2...Yk；
        弹出栈顶符号；
        将 Yk, Yk-1, ..., Y1 压入栈中，其中 Y1 位于栈顶。
    }
    令 X = 栈顶符号；
}

```

注：如果条目为 synch，那么在试图继续分析时，栈顶的非终结符号被弹出。

(a) **if e then s; if e then s end**

已匹配	栈	输入	动作
	$stmt\$$	<b>if e then s; if e then s end\$</b>	
	<b>if e then</b> $stmt stmtTail\$$	<b>if e then s; if e then s end\$</b>	输出①
<b>if</b>	<b>e then</b> $stmt stmtTail\$$	<b>e then s; if e then s end\$</b>	匹配 <b>if</b>
<b>if e</b>	<b>then</b> $stmt stmtTail\$$	<b>then s; if e then s end\$</b>	匹配 <b>e</b>
<b>if e then</b>	$stmt stmtTail\$$	<b>s; if e then s end\$</b>	匹配 <b>then</b>
	$s stmtTail\$$	<b>s; if e then s end\$</b>	输出④
<b>if e then s</b>	$stmtTail\$$	<b>; if e then s end\$</b>	匹配 <b>s</b>
	$\$$	<b>; if e then s end\$</b>	输出⑥

(b) **while  $e$  do begin  $s$ ; if  $e$  then  $s$ ; end**

已匹配	栈	输入	动作
	$stmt\$$	<b>while <math>e</math> do begin <math>s</math>; if <math>e</math> then <math>s</math>; end</b>	
	<b>while <math>e</math> do <math>stmt\\$</math></b>	<b>while <math>e</math> do begin <math>s</math>; if <math>e</math> then <math>s</math>; end</b>	输出②
<b>while</b>	$e$ <b>do <math>stmt\\$</math></b>	$e$ <b>do begin <math>s</math>; if <math>e</math> then <math>s</math>; end</b>	匹配 <b>while</b>
<b>while <math>e</math></b>	<b>do <math>stmt\\$</math></b>	<b>do begin <math>s</math>; if <math>e</math> then <math>s</math>; end</b>	匹配 $e$
<b>while <math>e</math> do</b>	$stmt\$$	<b>begin <math>s</math>; if <math>e</math> then <math>s</math>; end</b>	匹配 <b>do</b>
	<b>begin list end</b>	<b>begin <math>s</math>; if <math>e</math> then <math>s</math>; end</b>	输出③
<b>while <math>e</math> do begin</b>	$list$ <b>end</b>	$s$ ; <b>if <math>e</math> then <math>s</math>; end</b>	匹配 <b>begin</b>
	$stmt$ $listTail$ <b>end</b>	$s$ ; <b>if <math>e</math> then <math>s</math>; end</b>	输出⑦
	$s$ $listTail$ <b>end</b>	$s$ ; <b>if <math>e</math> then <math>s</math>; end</b>	输出④
<b>while <math>e</math> do begin <math>s</math></b>	$listTail$ <b>end</b>	$s$ ; <b>if <math>e</math> then <math>s</math>; end</b>	匹配 $s$
	$s$ ; $list$ <b>end</b>	$s$ ; <b>if <math>e</math> then <math>s</math>; end</b>	输出⑧
<b>while <math>e</math> do begin <math>s</math>;</b>	$list$ <b>end</b>	<b>if <math>e</math> then <math>s</math>; end</b>	匹配;
	$stmt$ $listTail$ <b>end</b>	<b>if <math>e</math> then <math>s</math>; end</b>	输出⑦
	<b>if <math>e</math> then <math>stmt</math> <math>stmtTail</math></b> $listTail$ <b>end</b>	<b>if <math>e</math> then <math>s</math>; end</b>	输出①
<b>while <math>e</math> do begin <math>s</math>; if</b>	$e$ <b>then <math>stmt</math> <math>stmtTail</math></b> $listTail$ <b>end</b>	$e$ <b>then <math>s</math>; end</b>	匹配 <b>if</b>
<b>while <math>e</math> do begin <math>s</math>;</b> <b>if <math>e</math></b>	<b>then <math>stmt</math> <math>stmtTail</math></b> $listTail$ <b>end</b>	<b>then <math>s</math>; end</b>	匹配 $e$
<b>while <math>e</math> do begin <math>s</math>;</b> <b>if <math>e</math> then</b>	$stmt$ $stmtTail$ $listTail$ <b>end</b>	$s$ ; <b>end</b>	匹配 <b>then</b>
	$s$ $stmtTail$ $listTail$ <b>end</b>	$s$ ; <b>end</b>	输出④
<b>while <math>e</math> do begin <math>s</math>;</b> <b>if <math>e</math> then <math>s</math></b>	$stmtTail$ $listTail$ <b>end</b>	$s$ ; <b>end</b>	匹配 $s$
	$listTail$ <b>end</b>	$s$ ; <b>end</b>	输出⑥
	$s$ ; $list$ $listTail$ <b>end</b>	$s$ ; <b>end</b>	输出⑧
<b>while <math>e</math> do begin <math>s</math>;</b> <b>if <math>e</math> then <math>s</math>;</b>	$list$ $listTail$ <b>end</b>	<b>end</b>	匹配;
	$listTail$ <b>end</b>	<b>end</b>	synch!弹出
	<b>end</b>	<b>end</b>	输出⑨
<b>while <math>e</math> do begin <math>s</math>;</b> <b>if <math>e</math> then <math>s</math>; end</b>	$s$	$s$	匹配 <b>end</b>