

编译原理第四章作业

1、答：(1)

将状态 S 、 T 按照 1、2 编序。

当 $i = 1, j = 0$ 时， S 不存在直接左递归；

当 $i = 2, j = 1$ 时，无需消除间接左递归，

消除 T 的直接左递归：

$$\begin{aligned} T &\rightarrow ST' \\ T' &\rightarrow , \quad ST' \mid \varepsilon \end{aligned}$$

因此消去左递归后 G_1 的文法为：

$$\begin{aligned} S &\rightarrow a \mid \wedge \mid (T) \\ T &\rightarrow ST' \\ T' &\rightarrow , \quad ST' \mid \varepsilon \end{aligned}$$

分析其 $FIRST$ 集与 $FOLLOW$ 集：

$$FIRST(S) = \{ a, \wedge, (\}$$

$$FIRST(T) = FIRST(S) = \{ a, \wedge, (\}$$

$$FIRST(T') = \{ , , \varepsilon \}$$

$$FOLLOW(S) = (FIRST(T') - \{ \varepsilon \}) \cup FOLLOW(T) \cup \{ \# \} = \{ , ,), \# \}$$

$$FOLLOW(T) = \{) \}$$

$$FOLLOW(T') = FOLLOW(T) = \{) \}$$

因此，不带回溯的递归子程序为：

```

P(S)
BEGIN
    IF ch = 'a' OR ch = '^' THEN
        read(ch);
    ELSE IF ch = '(' THEN
        BEGIN
            read(ch);
            P(T);
            IF ch = ')' THEN
                read(ch);
            ELSE ERROR;
        END
    ELSE ERROR;
END
P(T)
BEGIN
    P(S);
    P(T');
END
P(T')
BEGIN
    IF ch = ',' THEN
        read(ch);
        P(S);
        P(T');
    ELSE IF ch = ')' THEN
        return;
    ELSE ERROR;
END

```

(2) 该文法不存在左递归（包括直接左递归和间接左递归）；

文法中每个非终结符 A 的各个产生式的候选首符集两两不相交；

对于 $\varepsilon \in FIRST(A)$ 的非终结符 A (即 T') , 满足 $FIRST(A) \cap FOLLOW(A) = \emptyset$ 。

因此, 该文法是 $LL(1)$ 的。

预测分析表如下:

	()	a	\wedge	,	#
S	$S \rightarrow (T)$		$S \rightarrow a$	$S \rightarrow \wedge$		
T	$T \rightarrow ST'$		$T \rightarrow ST'$	$T \rightarrow ST'$		
T'		$T' \rightarrow \varepsilon$			$T' \rightarrow, ST'$	

2、答: (1)

$$FIRST(E) = FIRST(T) = FIRST(F) = FIRST(P) = \{ (, a, b, \wedge \}$$

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

$$FIRST(T) = FIRST(F) = FIRST(P) = \{ (, a, b, \wedge \}$$

$$FIRST(T') = FIRST(F) \cup \{\varepsilon\} = \{ (, a, b, \wedge, \varepsilon \}$$

$$FIRST(F) = FIRST(P) = \{ (, a, b, \wedge \}$$

$$FIRST(F') = \{ *, \varepsilon \}$$

$$FIRST(P) = \{ (, a, b, \wedge \}$$

$$FOLLOW(E) = \{ \#,) \}$$

$$FOLLOW(E') = FOLLOW(E) = \{ \#,) \}$$

$$FOLLOW(T) = (FIRST(E') - \{\varepsilon\}) \cup FOLLOW(E) = \{ \#,), + \}$$

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

$$FOLLOW(F) = (FIRST(T') - \{\varepsilon\}) \cup FOLLOW(T) = \{ (, a, b, \wedge, \#,), + \}$$

$$FOLLOW(F') = FOLLOW(F) = \{ (, a, b, \wedge, \#,), + \}$$

$$FOLLOW(P) = (FIRST(F') - \{ \epsilon \}) \cup FOLLOW(F) = \{ (, a, b, \wedge, \#,), +, * \}$$

(2) 该文法不存在左递归 (包括直接左递归和间接左递归);

文法中每个非终结符 A 的各个产生式的候选首符集两两不相交;

对于 $\epsilon \in FIRST(A)$ 的非终结符 A (即 E' , T' , F') , 均满足 $FIRST(A) \cap FOLLOW(A) = \emptyset$ 。

因此, 该文法是 $LL(1)$ 的。

(3)

	+	*	\wedge	a	b	()	#
E			$E \rightarrow TE'$	$E \rightarrow TE'$	$E \rightarrow TE'$	$E \rightarrow TE'$		
E'	$E' \rightarrow +E$						$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T			$T \rightarrow FT'$	$T \rightarrow FT'$	$T \rightarrow FT'$	$T \rightarrow FT'$		
T'	$T' \rightarrow \epsilon$		$T' \rightarrow T$	$T' \rightarrow T$	$T' \rightarrow T$	$T' \rightarrow T$	$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F			$F \rightarrow PF'$	$F \rightarrow PF'$	$F \rightarrow PF'$	$F \rightarrow PF'$		
F'	$F' \rightarrow \epsilon$	$F' \rightarrow *$	$F' \rightarrow \epsilon$					
P			$P \rightarrow \wedge$	$P \rightarrow a$	$P \rightarrow b$	$P \rightarrow (E)$		

(4)

P(E) BEGIN P(T) ; P(E') ; END	P(E') BEGIN IF ch = '+' THEN BEGIN read(ch) ; P(E) ; END ELSE IF ch = '#' OR ch = ')' THEN return ; ELSE ERROR; END
P(T) BEGIN P(F) ; P(T') ; END	P(T') BEGIN IF ch = '#' or ch = ')' or ch = '+' THEN return ; ELSE BEGIN P(T) ; END END
P(F) BEGIN P(P) ; P(F') ; END	P(F') BEGIN IF ch = '*' THEN read(ch) ; P(F') ; ELSE IF ch = 'a' OR ch = 'b' OR ch = '(' OR ch = ')' OR ch = '&' OR ch = '+' OR ch = '#' THEN return ; ELSE ERROR; END
P(P) BEGIN IF ch = '(' THEN read(ch) ;	
P(E) ; if ch = ')' THEN read(ch) ; ELSE ERROR; ELSE IF ch = 'a' OR ch = 'b' OR ch = '&' THEN read(ch) ; ELSE ERROR; END	

3、(1)解：(1)根据定义判断该文法是否为LL(1)。

①该文法中不含左递归，符合LL(1)文法定义的第一个条件。

② $S \rightarrow A\epsilon$ 中只有一个候选式，不需要考虑候选式FIRST相交是否为 \emptyset 的情况。

$A \rightarrow a \mid \epsilon$ 中， $\text{FIRST}(a) = \{a\}$ $\text{FIRST}(\epsilon) = \{\epsilon\}$

$\text{FIRST}(a) \cap \text{FIRST}(\epsilon) = \emptyset$

$B \rightarrow b \mid \epsilon$ 中， $\text{FIRST}(b) = \{b\}$ $\text{FIRST}(\epsilon) = \{\epsilon\}$

$\text{FIRST}(b) \cap \text{FIRST}(\epsilon) = \emptyset$ 。

该文法中任一非终结符的产生式的FIRST集两两不相交，符合LL(1)文法定义的第二个条件。

③ $\text{FIRST}(A) = \{a, \epsilon\}$ $\text{FIRST}(B) = \{b, \epsilon\}$ $\text{FIRST}(S) = \{a, b\}$

$\epsilon \notin \text{FIRST}(S)$ ，不考虑FIRST(S)与FOLLOW(S)相交情况。

$\epsilon \in \text{FIRST}(A)$ $\text{FOLLOW}(A) = \{b\}$ $\text{FIRST}(A) \cap \text{FOLLOW}(A) = \emptyset$

$\epsilon \in \text{FIRST}(B)$ $\text{FOLLOW}(B) = \emptyset$ $\text{FIRST}(B) \cap \text{FOLLOW}(B) = \emptyset$

该文法中，任一非终结符，若 ϵ 属于该非终结符的FIRST集，则非终结符的FIRST集与FOLLOW集交为 \emptyset ，符合LL(1)文法的第三个条件。

因此，该文法是LL(1)文法。

(2)解：①首先该文法无左递归，计算各非终结符的FIRST集和FOLLOW集

$\text{FIRST}(S) = \{a, b\}$

$\text{FIRST}(A) = \{a, b, \epsilon\}$

$\text{FIRST}(B) = \{b, \epsilon\}$

$\text{FOLLOW}(S) = \{\#\}$

$\text{FOLLOW}(A) = \text{FOLLOW}(B) / \{\epsilon\} = \{b\}$

$\text{FOLLOW}(B) = \text{FOLLOW}(A) = \{b\}$

②考虑 $A \rightarrow a \mid B \mid \epsilon$ 与 $B \rightarrow b \mid \epsilon$ 的话，可知

$\text{FIRST}(B) \cap \text{FOLLOW}(A) = \{b\}$

该文法不是LL(1)文法。

4、解：(1)计算各非终结符的FIRST集和FOLLOW集如下：

$$\text{FIRST}(\text{Expr}) = \{ -, (, \text{id} \} \}$$

$$\text{FIRST}(\text{ExprTail}) = \{ -, \epsilon \}$$

$$\text{FIRST}(\text{Var}) = \{ \text{id} \}$$

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

$$\text{FOLLOW}(\text{Expr}) = \{), \# \}$$

$$\text{FOLLOW}(\text{ExprTail}) = \{), \# \}$$

$$\text{FOLLOW}(\text{Var}) = \{ -,), \# \}$$

$$\text{FOLLOW}(\text{VarTail}) = \{ -, |, \# \}$$

构造其预测分析表如下：

	-	id	()	#
Expr	Expr \rightarrow -Expr	Expr \rightarrow Var ExprTail	Expr \rightarrow (Expr)		
ExprTail	ExprTail \rightarrow -Expr			ExprTail \rightarrow ϵ	ExprTail \rightarrow ϵ
Var		Var \rightarrow id VarTail			
VarTail	VarTail \rightarrow ϵ		VarTail \rightarrow (Expr)	VarTail \rightarrow ϵ	VarTail \rightarrow ϵ

(2)句子id--id((id))的分析过程如下：

步骤	符号栈	输入串	所用产生式
0	# Expr	id - id((id)) #	
1	# ExprTail Var	id - id((id)) #	Expr \rightarrow Var ExprTail
2	# ExprTail VarTail id	id - id((id)) #	Var \rightarrow id VarTail
3	# ExprTail VarTail	- id((id)) #	
4	# ExprTail	- id((id)) #	VarTail \rightarrow ϵ
5	# Expr -	- id((id)) #	ExprTail \rightarrow -Expr
6	# Expr	id((id)) #	
7	# Expr	id((id)) #	Expr \rightarrow Expr
8	# ExprTail Var	id((id)) #	Expr \rightarrow Var ExprTail
9	# ExprTail VarTail id	id((id)) #	Var \rightarrow id VarTail
10	# ExprTail VarTail	((id)) #	
11	# ExprTail) Expr (((id)) #	VarTail \rightarrow (Expr)
12	# ExprTail) Expr	(id)) #	
13	# ExprTail)) Expr ((id)) #	
14	# ExprTail)) Expr) #	
15	# ExprTail)) ExprTail Var) #	Expr \rightarrow Var ExprTail
16	# ExprTail)) ExprTail VarTail id	id)) #	Var \rightarrow id VarTail
17	# ExprTail)) ExprTail VarTail) #	

```
18 # ExprTail ) ) ExprTail )) #           VarTail → ε
19 # ExprTail ) )                 )) #           ExprTail → ε
20 # ExprTail )                 ) #           ExprTail → ε
21 # ExprTail      #
22 #               #
23                                         分析成功 (suc)
```