

第一章

1. 典型的编译程序在逻辑功能上由哪几部分组成？

答：编译程序主要由以下几个部分组成：词法分析、语法分析、语义分析、中间代码生成、中间代码优化、目标代码生成、错误处理、表格管理。

2. 实现编译程序的主要方法有哪些？

答：主要有：转换法、移植法、自展法、自动生成法。

3. 将用户使用高级语言编写的程序翻译为可直接执行的机器语言程序有哪几种主要的方式？

答：编译法、解释法。

4. 编译方式和解释方式的根本区别是什么？

答：编译方式：是将源程序经编译得到可执行文件后，就可脱离源程序和编译程序单独执行，所以编译方式的效率高，执行速度快；

解释方式：在执行时，必须源程序和解释程序同时参与才能运行，其不产生可执行程序文件，效率低，执行速度慢。

第二章

1. 乔姆斯基文法体系中将文法分为哪几类？文法的分类同程序设计语言的设计与实现关系如何？

答：1) 0 型文法、1 型文法、2 型文法、3 型文法。

2)

2. 写一个文法，使其语言是偶整数的集合，每个偶整数不以 0 为前导。

答：

$Z \rightarrow SME \mid B$

$S \rightarrow 1|2|3|4|5|6|7|8|9$

$M \rightarrow \varepsilon \mid D \mid MD$

$D \rightarrow 0 \mid S$

$B \rightarrow 2|4|6|8$

$E \rightarrow 0 \mid B$

3. 设文法 G 为：

$N \rightarrow D \mid ND$

$D \rightarrow 0|1|2|3|4|5|6|7|8|9$

请给出句子 123、301 和 75431 的最右推导和最左推导。

答： $N \Rightarrow ND \Rightarrow N3 \Rightarrow ND3 \Rightarrow N23 \Rightarrow D23 \Rightarrow 123$

$N \Rightarrow ND \Rightarrow NDD \Rightarrow DDD \Rightarrow 1DD \Rightarrow 12D \Rightarrow 123$

$N \Rightarrow ND \Rightarrow N1 \Rightarrow ND1 \Rightarrow N01 \Rightarrow D01 \Rightarrow 301$

$N \Rightarrow ND \Rightarrow NDD \Rightarrow DDD \Rightarrow 3DD \Rightarrow 30D \Rightarrow 301$

$N \Rightarrow ND \Rightarrow N1 \Rightarrow ND1 \Rightarrow N31 \Rightarrow ND31 \Rightarrow N431 \Rightarrow ND431 \Rightarrow N5431 \Rightarrow D5431 \Rightarrow 75431$

$N \Rightarrow ND \Rightarrow NDD \Rightarrow NDDD \Rightarrow NDDDD \Rightarrow DDDDD \Rightarrow 7DDDD \Rightarrow 75DDDD \Rightarrow 754DD \Rightarrow 7543D \Rightarrow 75431$

4. 证明文法 $S \rightarrow iSeS \mid iS \mid i$ 是二义性文法。

答：对于句型 $iiSeS$ 存在两个不同的最左推导：

$S \Rightarrow iSeS \Rightarrow iiSes$

$S \Rightarrow iS \Rightarrow iiSeS$

所以该文法是二义性文法。

5. 给出描述下面语言的上下文无关文法。

(1) $L1 = \{a^n b^n c^i \mid n \geq 1, i \geq 0\}$

(2) $L2 = \{a^i b^j \mid j \geq i \geq 1\}$

(3) $L3 = \{a^n b^m c^m d^n \mid m, n \geq 0\}$

答：

(1) $S \rightarrow AB$

$A \rightarrow aAb \mid ab$

$B \rightarrow cB \mid \varepsilon$

(2) $S \rightarrow ASb \mid ab$

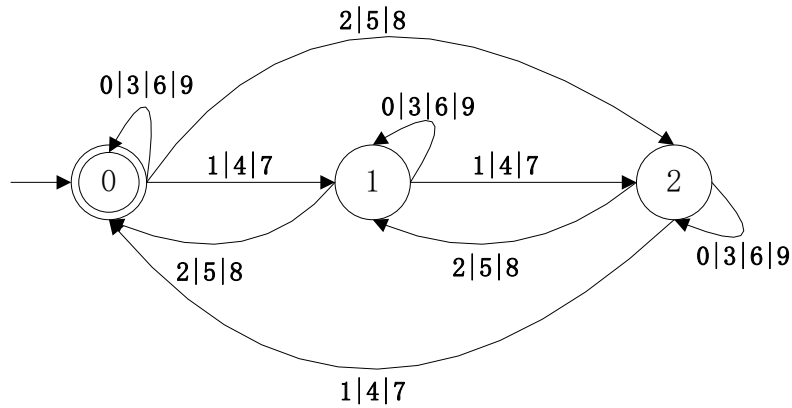
$A \rightarrow a \mid \varepsilon$

(3) $S \rightarrow aSd \mid A \mid \varepsilon$

$A \rightarrow bAc \mid \varepsilon$

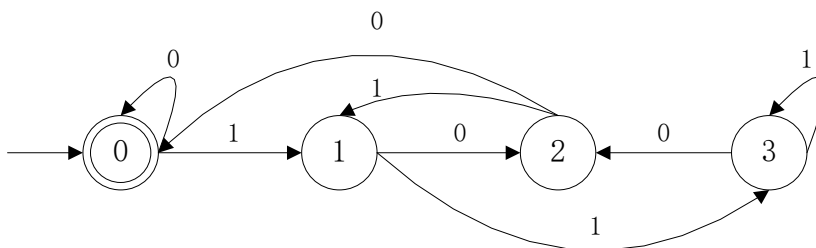
6. 设计一个最简的 DFA M,使其能够识别所有的被 3 整除的无符号十进制整数。

答:



7. 设计一个 DFA,使其能够接受被 4 整除的二进制数。

答:



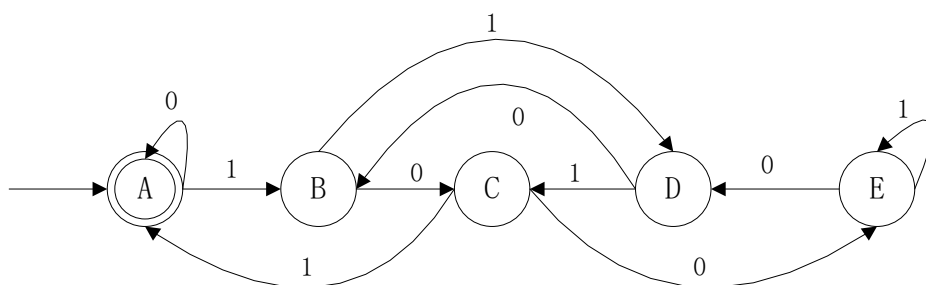
8. 写出表达下列各项的正则表达式。

- (1) 二进制数且为 5 的倍数。
- (2) $\Sigma = \{a, b, c\}$, 第一个 a 位于第一个 b 之前的字符串。
- (3) $\Sigma = \{a, b, c\}$, 包含偶数个 a 的字符串。
- (4) $\Sigma = \{0, 1\}$, 不包含 11 子串的字符串。

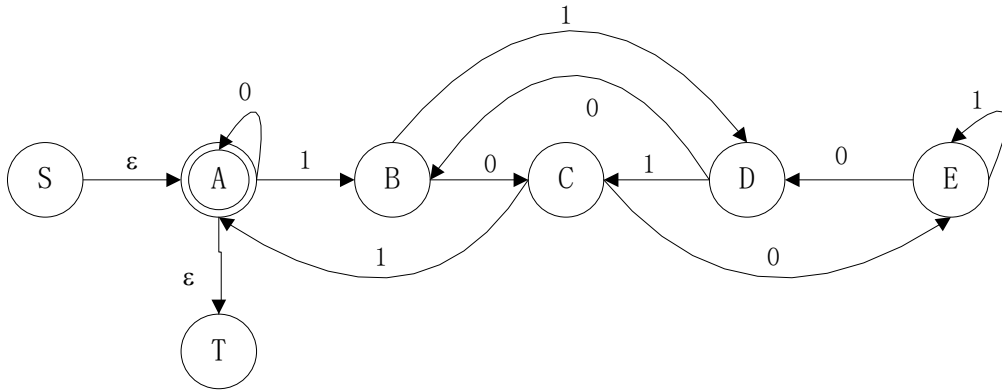
答:

(1)

可以先画出相应的有限自动机:



添加新的开始状态 S 和终止状态 T:



根据以上自动机，列出以下方程：

- ① $S=A$
- ② $A=0A+1B+T$
- ③ $B=0C+1D$
- ④ $C=0E+1A$
- ⑤ $D=0B+1C$
- ⑥ $E=0D+1E$

解以上方程组：

$$\textcircled{6} \Rightarrow E=1^*0D$$

$$\textcircled{2} \Rightarrow A=0^*1B+0^*T$$

$$\textcircled{6} \text{ 代入 } \textcircled{4} \Rightarrow C=01^*0D+1A$$

$$\textcircled{5} \text{ 代入 } \textcircled{4} \Rightarrow C=01^*00B+01^*01C+1A$$

$$\Rightarrow C=xB+yA$$

$$\text{其中 } x=(01^*01)^*01^*00 \quad y=(01^*01)^*1$$

$$\textcircled{5} \text{ 代入 } \textcircled{3} \Rightarrow B=0C+10B+11C$$

$$\Rightarrow B=(0|11)C+10B$$

$$\Rightarrow B=(10)^*(0|11)C$$

$$\text{将 } C=xB+yA \text{ 代入上式} \Rightarrow B=uB+vA$$

$$\Rightarrow B=u^*vA$$

$$\text{其中 } u=(10)^*(0|11)x \quad v=(10)^*(0|11)y$$

$$\text{将 } B=u^*vA \text{ 代入 } \textcircled{2} \Rightarrow A=0^*1u^*vA+0^*T$$

$$\Rightarrow A=(0^*1u^*v)^*0^*T$$

$$\text{将 } A \text{ 代入 } \textcircled{1} \Rightarrow S=(0^*1u^*v)^*0^*T$$

串 $(0^*1u^*v)^*0^*$ 即为所求。

(2) 该题目理解为：至少有一个 a 和一个 b ，且 a 出现在 b 的前面（可以有间隔），则答案为：

$$c^*a(a|c)^*b(a|b|c)^*$$

$$(3) ((b|c)^*a(b|c)^*a)^*(b|c)^* \quad (a(b|c)^*a | b | c)^*$$

$$(4) (0|10)^*(1|\varepsilon)$$

第三章

1. 词法分析器的功能是什么？

答：读源程序的字符序列，逐个拼出单词，并构造相应的内部表示 TOKEN；同时检查源程序中的词法错误。

2. 试分析分隔符（空格、跳格及回车等）对词法分析的影响。

答：空格、跳格、回车等分隔符号对词法分析不起作用，可以删除。但是回车符号可以用于错误定位，所以在删除回车符号前需要统计回车的个数。

3. 给出识别 C 语言全部实型常数的自动机。

答： $(+|-|\epsilon)([1-9][0-9]^*|0)(.[0-9][0-9]^*|\epsilon)(E(+|-|\epsilon)[0-9][0-9]^*)$

4. 写出识别 C 语言中所有单词的 LEX 程序。

答：

```
L=[A-Z] | [a-z]
D=[0-9]
D1=[1-9]
%%
(L|_)(L|D|_)* {return (1);}
D1D*          {return (2);}
+              {return (3);}
.....
```

第四章

1. 设有如下文法 $G[S]$:

$$S \rightarrow aABbcd \mid \varepsilon$$
$$A \rightarrow ASd \mid \varepsilon$$
$$B \rightarrow SAh \mid eC \mid \varepsilon$$
$$C \rightarrow Sf \mid Cg \mid \varepsilon$$

(1) 求每个产生式的 Predict 集。

(2) 该文法是否为 LL(1)文法? 为什么?

答: (1)

$$\text{Predict}(S \rightarrow aABbcd) = \{a\}$$
$$\text{Predict}(S \rightarrow \varepsilon) = \{\#, d, f, a, h\}$$
$$\text{Predict}(A \rightarrow ASd) = \{a, d\}$$
$$\text{Predict}(A \rightarrow \varepsilon) = \{h, a, d, b, e\}$$
$$\text{Predict}(B \rightarrow SAh) = \{a, d, h\}$$
$$\text{Predict}(B \rightarrow eC) = \{e\}$$
$$\text{Predict}(B \rightarrow \varepsilon) = \{b\}$$
$$\text{Predict}(C \rightarrow Sf) = \{a, f\}$$
$$\text{Predict}(C \rightarrow Cg) = \{a, f, g\}$$
$$\text{Predict}(C \rightarrow \varepsilon) = \{g, b\}$$

(2) 由于 $\text{Predict}(A \rightarrow ASd) \cap \text{Predict}(A \rightarrow \varepsilon) \neq \emptyset$, 所以该文法不是 LL(1)文法。

2. 下列描述括号匹配的文法中, 哪些是 LL(1)文法?

(1) $S \rightarrow (SS' \mid \varepsilon$

$$S' \rightarrow) \mid \varepsilon$$

(2) $S \rightarrow (S)S \mid \varepsilon$

(3) $S \rightarrow S(S)S \mid \varepsilon$

(4) $S \rightarrow (S \mid S'$

$$S' \rightarrow (S') \mid \varepsilon$$

答: (1)不是, (2)是, (3)不是, (4)不是

3. 已知文法 $G[E]$:

$$E \rightarrow E+T \mid T$$
$$T \rightarrow T * F \mid F$$
$$F \rightarrow i \mid (E)$$

请按递归下降法构造该文法的语法分析程序。

答: 求产生式的 predict 集合:

$$\text{predict}(E \rightarrow E+T) = \{i, (\}$$
$$\text{predict}(E \rightarrow T) = \{i, (\}$$
$$\text{predict}(T \rightarrow T * F) = \{i, (\}$$
$$\text{predict}(T \rightarrow F) = \{i, (\}$$

由于文法中非终极符号 E 和 T 对应的产生式的 **predict** 集合的交集都不为空，所以该文法不满足自顶向下分析的条件，现对文法进行等价变换得到如下文法：

$E \rightarrow TE'$

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

$T \rightarrow FT'$

$T' \rightarrow *FT' \mid \varepsilon$

$F \rightarrow i$

$F \rightarrow (E)$

求新文法的 **predict** 集合：

$\text{Predict}(E \rightarrow TE') = \{ (, i \}$

$\text{Predict}(E' \rightarrow +TE') = \{ + \}$

$\text{Predict}(E' \rightarrow \varepsilon) = \{ \#, \}$

$\text{Predict}(T \rightarrow FT') = \{ i, (\}$

$\text{Predict}(T' \rightarrow *FT') = \{ * \}$

$\text{Predict}(T' \rightarrow \varepsilon) = \{ +, \#, \}$

$\text{Predict}(F \rightarrow i) = \{ i \}$

$\text{Predict}(F \rightarrow (E)) = \{ (\}$

由于以上文法中任意非终极符号对应的产生式的 **predict** 集合的交集都为空，所以满足自顶向下分析的条件，所以可以写出如下的递归下降语法分析伪代码：

Void E()

{ if(token $\in \{ (, i \}$) {T();E'();}

else Error();}

void E'()

{ if(token $\in \{ + \}$) {Match('+');T();E'();}

else if(token $\in \{ \#, \}$) {;}

else Error();}

void T()

{ if(token $\in \{ i, (\}$) {F();T'();}

else Error();}

void T'()

{ if(token $\in \{ * \}$) {Match('*');F();T'();}

else if(token $\in \{ +, \#, \}$) {;}

else Error();}

void F()

{ if(token $\in \{ i \}$) {Match('i');}

else if(token $\in \{ (\}$) {Match('(');E();Match(')');}

else Error();}

4. 构造一个 LL(1)文法 G ，它能识别语言 L ：

$L = \{ \omega \mid \omega \text{ 为字母表 } \Sigma \text{ 上不包括两个相邻的 } 1 \text{ 的非空串} \}$ ，其中 $\Sigma = \{ 0, 1 \}$ 。

并证明你所构造的文法是 LL(1)文法。

答: $A \rightarrow 0E \mid 1F$

$B \rightarrow 0E \mid 1F$

$C \rightarrow 0E$

$E \rightarrow B \mid \varepsilon$

$F \rightarrow C \mid \varepsilon$

$\text{Predict}(A \rightarrow 0E) = \{0\}$

$\text{Predict}(A \rightarrow 1F) = \{1\}$

$\text{Predict}(B \rightarrow 0E) = \{0\}$

$\text{Predict}(B \rightarrow 1F) = \{1\}$

$\text{Predict}(E \rightarrow B) = \{0, 1\}$

$\text{Predict}(E \rightarrow \varepsilon) = \{\#\}$

$\text{Predict}(F \rightarrow C) = \{0\}$

$\text{Predict}(F \rightarrow \varepsilon) = \{\#\}$

对任意非终极符号对应的产生式的 predict 集合的交集都为空, 所以该文法是 LL(1)文法。

5. 已知文法 $G[A]$ 为:

$A \rightarrow aABe \mid a$

$B \rightarrow Bb \mid d$

(1) 试给出与 $G[A]$ 等价的 LL(1)文法 $G'[A]$ 。

(2) 构造 $G'[A]$ 的 LL(1)分析表并给出输入串 $aade\#$ 的分析过程。

答: (1) 所求 $G'[A]$ 为:

$A \rightarrow aA'$ (1)

$A' \rightarrow ABe$ (2)

$A' \rightarrow \varepsilon$ (3)

$B \rightarrow dB'$ (4)

$B' \rightarrow bB'$ (5)

$B' \rightarrow \varepsilon$ (6)

$\text{Predict}(A \rightarrow aA') = \{a\}$

$\text{Predict}(A' \rightarrow ABe) = \{a\}$

$\text{Predict}(A' \rightarrow \varepsilon) = \{\#, d\}$

$\text{Predict}(B \rightarrow dB') = \{d\}$

$\text{Predict}(B' \rightarrow bB') = \{b\}$

$\text{Predict}(B' \rightarrow \varepsilon) = \{e\}$

对任意非终极符号对应的产生式的 predict 集合的交集都为空, 所以该文法是 LL(1)文法。

(3) 分析表如下:

	a	b	d	e	#
A	(1)				
A'	(2)		(3)		(3)

B			(4)		
B'		(5)		(6)	

aade#的分析过程如下

分析栈	输入流	动作
A#	aade#	替换(1)
aA' #	aade#	匹配
A' #	ade#	替换(2)
ABe#	ade#	替换(1)
aA'Be#	ade#	匹配
A'Be#	de#	替换(3)
Be#	de#	替换(4)
dB'e#	de#	匹配
B'e#	e#	替换
e#	e#	匹配
#	#	成功

第五章（这章答案是错的）

1. 设有下列文法：

(1) $S \rightarrow aA$

$A \rightarrow Ab$

$A \rightarrow b$

(2) $S \rightarrow aSSb$

$S \rightarrow aSSS$

$S \rightarrow c$

(3) $S \rightarrow AS$

$S \rightarrow b$

$A \rightarrow SA$

$A \rightarrow a$

(4) $S \rightarrow cA$

$S \rightarrow ccB$

$B \rightarrow ccB$

$B \rightarrow b$

$A \rightarrow cA$

$A \rightarrow a$

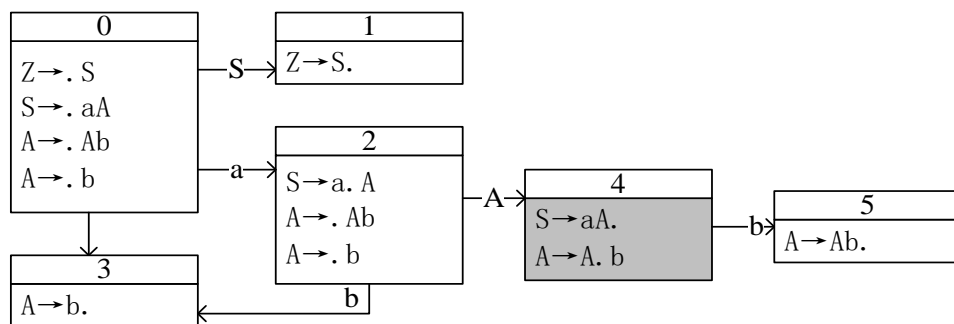
构造上述文法的 LR(0)归约活前缀状态机，并给出 LR(0)分析表。

答：

(1)

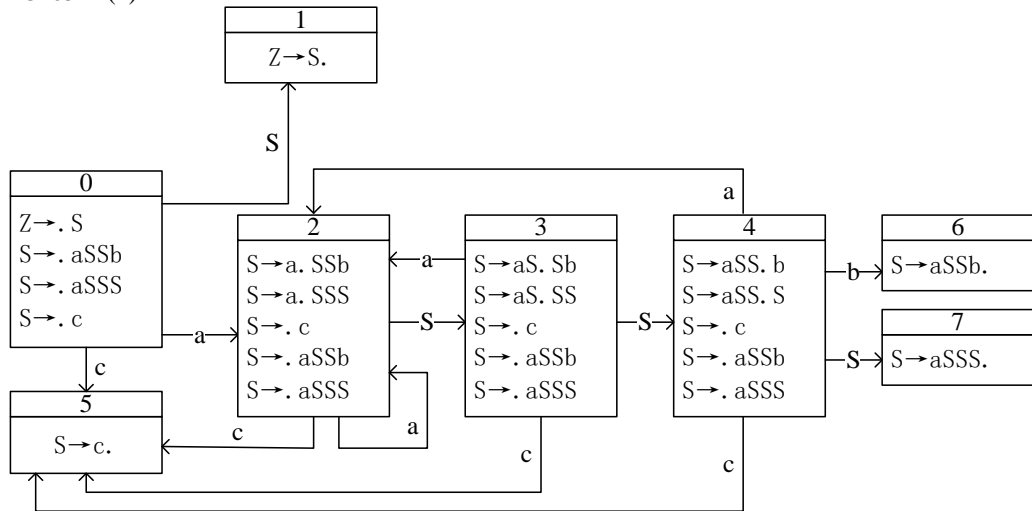
Ch05-1-(1)

有移入、规约冲突



(2)

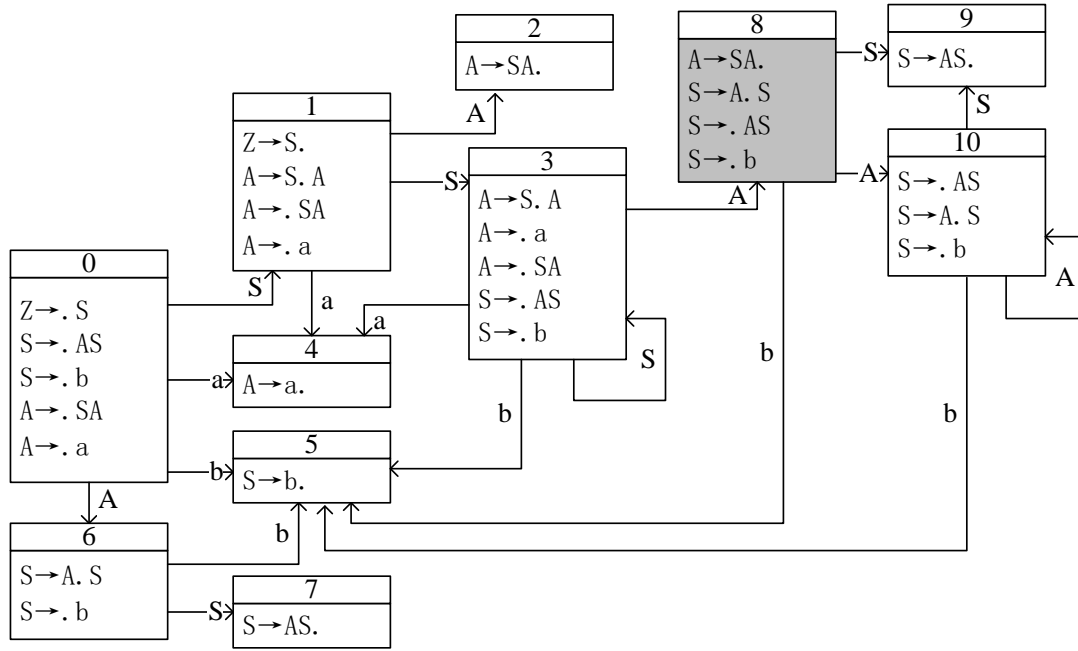
Ch05-1-(2)



$Z \rightarrow \cdot S$ (1)
 $S \rightarrow \cdot aSSb$ (2)
 $S \rightarrow \cdot aSSS$ (3)
 $S \rightarrow \cdot c$ (4)

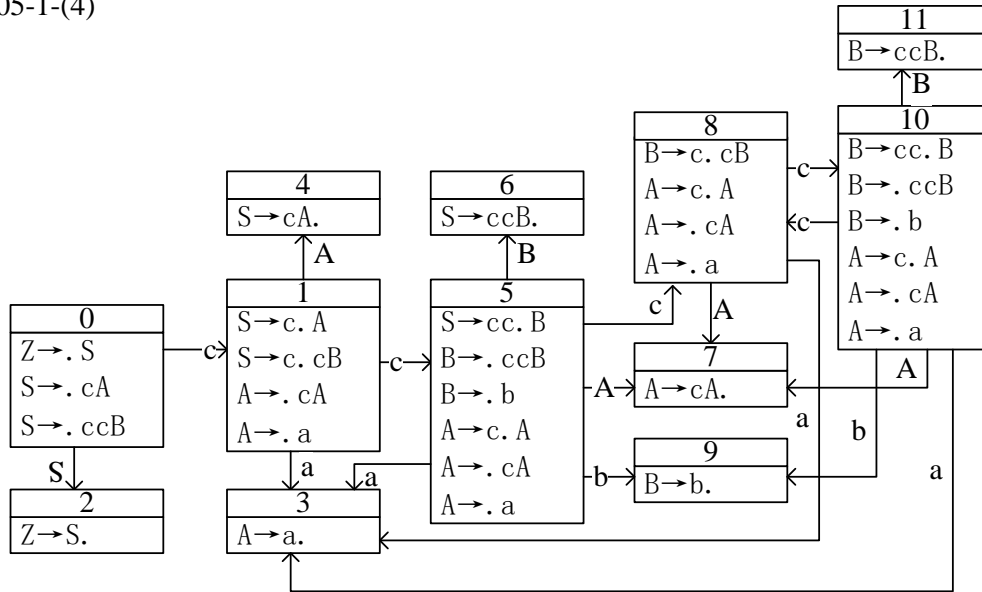
	action				goto
	a	b	c	#	S
S0	s2		s5		1
S1				Acc	
S2	s2		s5		3
S3	s2		s5		4
S4	s2	s6	s5		7
S5	R4	R4	R4	R4	
S6	R2	R2	R2	R2	
S7	R3	R3	R3	R3	

(3)



(4)

Ch05-1-(4)



$Z \rightarrow S$ (1)
 $S \rightarrow cA$ (2)
 $S \rightarrow ccB$ (3)
 $B \rightarrow ccB$ (4)
 $B \rightarrow b$ (5)
 $A \rightarrow cA$ (6)
 $A \rightarrow a$ (7)

	action				goto		
	a	b	c	#	A	B	S
S0			s1				2
S1	s3		s5		4		
S2				Acc			
S3	R7	R7	R7	R7			
S4	R6	R6	R6	R6			
S5	s3	s9	s8		7	6	
S6	R3	R3	R3	R3			
S7	R2	R2	R2	R2			
S8	s3		s10		7		
S9	R5	R5	R5	R5			
S10	s3	s9	s8		7	11	
S11	R4	R4	R4	R4			

2. 设有下列文法：

- (1) $S \rightarrow SaS \mid b$
 (2) $S \rightarrow bSb \mid cSc \mid b \mid c$
 (3) $S \rightarrow bSb \mid bSc \mid d$
 (4) $S \rightarrow aSb \mid bSa \mid ab$
 (5) $S \rightarrow Sab \mid bR$
 $R \rightarrow S \mid a$
 (6) $S \rightarrow SAB \mid BA$
 $B \rightarrow b$
 $A \rightarrow aA \mid B$
 (7) $S \rightarrow AaAb \mid BbBa$
 $B \rightarrow \varepsilon$
 $A \rightarrow \varepsilon$
 (8) $A \rightarrow aABe \mid Ba$
 $B \rightarrow dB \mid \varepsilon$

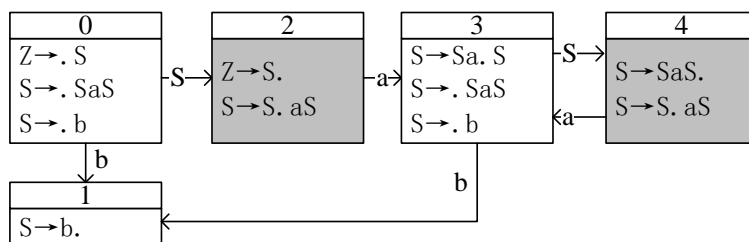
说明上述文法是否为 SLR(1) 文法。若是，请构造 SLR(1) 分析表；若不是，请说明理由。

答：

(1)

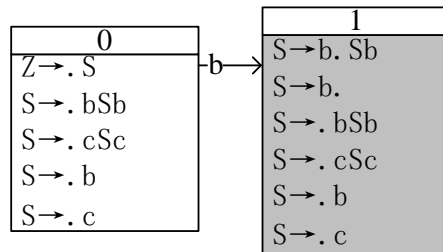
Ch05-2-(1) 不是 SLR (1)

Follow(S)={#,a}



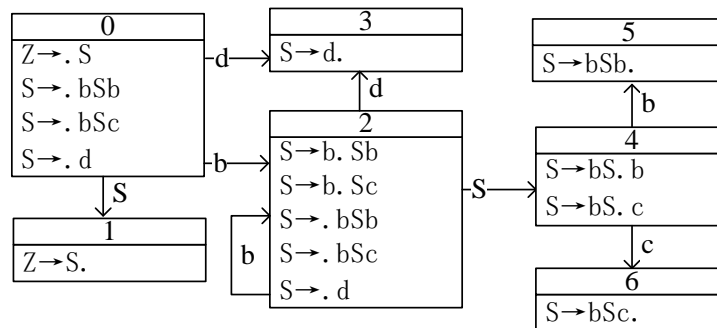
(2)

Ch05-2-(2) 不是SLR (1)
Follow(S)={#,b,c}



(3)

Ch05-2-(3) 是SLR (1)



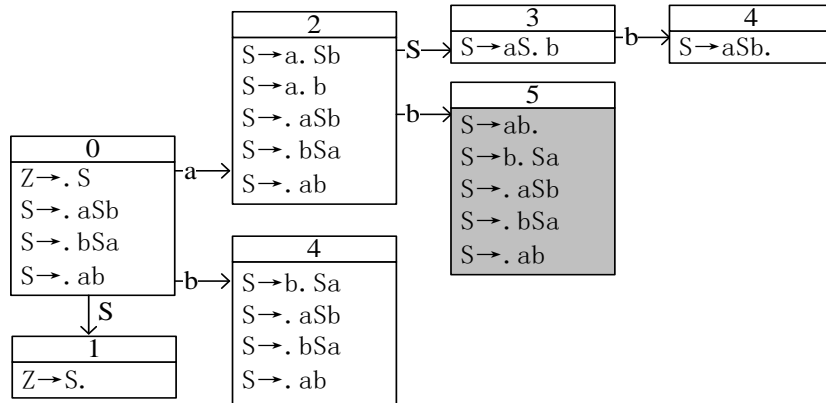
$Z \rightarrow . S$ (1)
 $S \rightarrow . b Sb$ (2)
 $S \rightarrow . b Sc$ (3)
 $S \rightarrow . d$ (4)

	action				goto
	b	c	d	#	S
S0	s2		s3		1
S1				Acc	
S2	s2		s3		4
S3	R4	R4		R4	
S4	s5	s6			
S5	R2	R2		R2	
S6	R3	R3		R3	

(4)

Ch05-2-(4) 不是SLR (1)

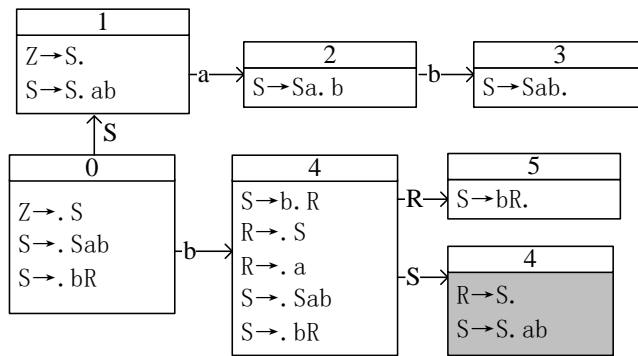
Follow(S)={#,a,b}



(5)

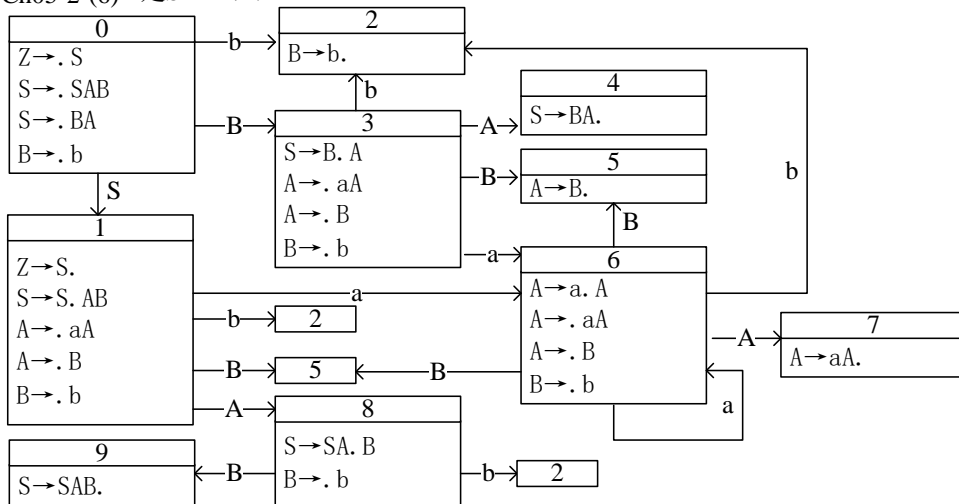
Ch05-2-(5) 不是SLR (1)

Follow(R)={#,a}



(6)

Ch05-2-(6) 是SLR (1)



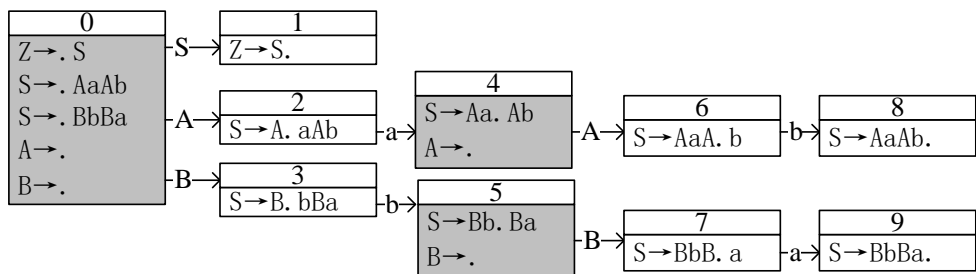
action				goto		
	a	b	#	A	B	S
S0		s2			3	1
S1	s6	s2	Acc	8	5	
S2	R6	R6	R6			
S3	s6	s2		4	5	
S4	R3	R3	R3			
S5	R5	R5	R5			
S6	s6	s2		7	5	
S7	R4	R4	R4			
S8		s2			9	
S9	R2	R2	R2			

$Z \rightarrow S$ (1)
 $S \rightarrow SAB$ (2)
 $S \rightarrow BA$ (3)
 $A \rightarrow aA$ (4)
 $A \rightarrow B$ (5)
 $B \rightarrow b$ (6)

(7)

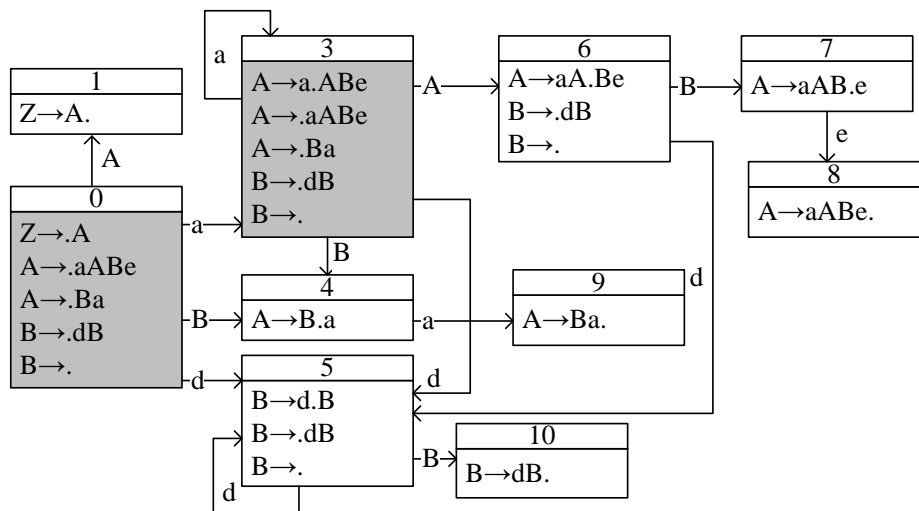
Ch05-2-(7) 不是SLR (1)

$\text{Follow}(A) = \{a, b\}$ $\text{Follow}(B) = \{a, b\}$



(8)

Ch05-2-(8) 不是SLR (1) $\text{Follow}(B) = \{a, e\}$



3. 设有下列文法:

$S \rightarrow aAd \mid bBd \mid aBe \mid bAe$

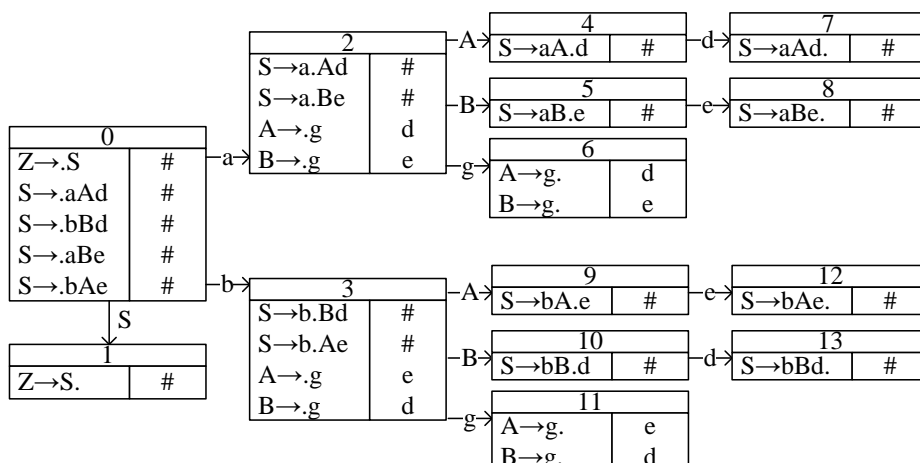
$A \rightarrow g$

$B \rightarrow g$

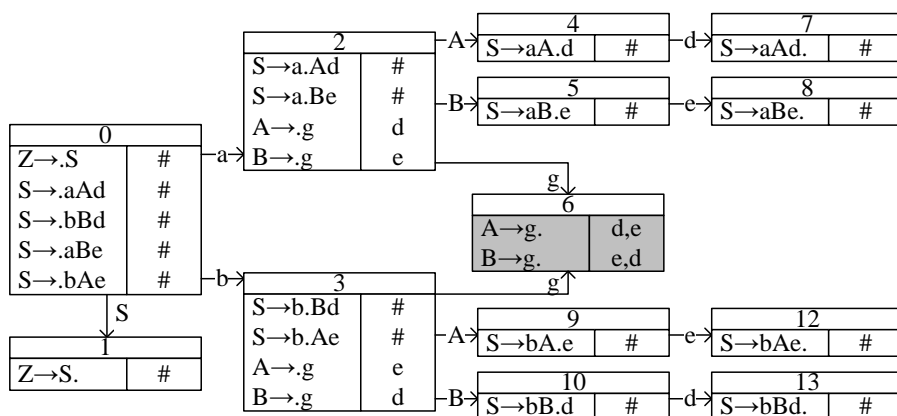
说明该文法是 LR(1)文法, 但不是 LALR(1)文法。

答:

Ch05-3 是LR(1)文法



Ch05-3 不是LALR(1)文法



4. 设有下列文法:

(1) $E \rightarrow E+T \mid T$

$T \rightarrow TF \mid T$

$F \rightarrow (E) \mid F^* \mid a \mid b$

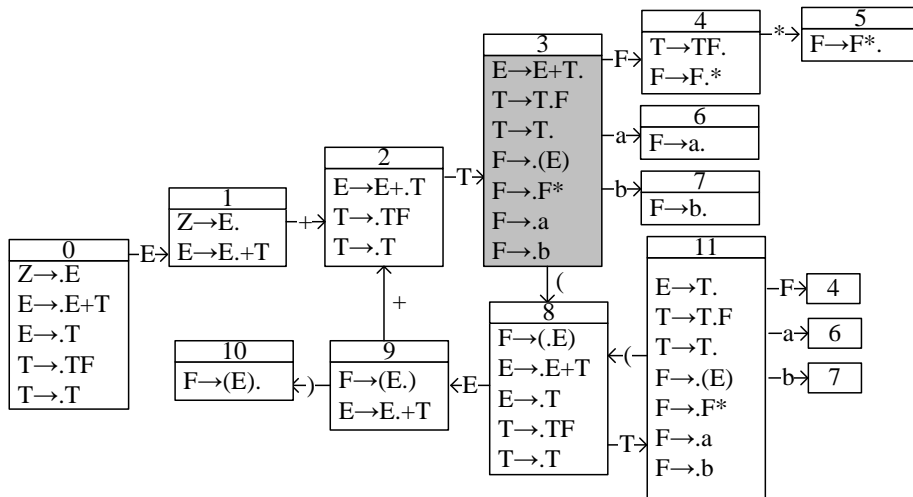
(2) $S \rightarrow Aa \mid bAc \mid dc \mid bda$

$A \rightarrow d$

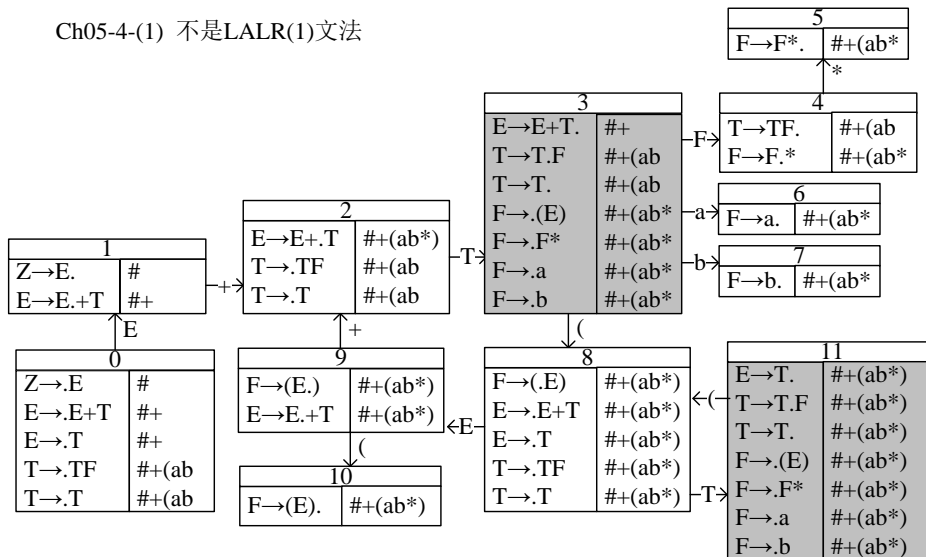
说明上述文法是否为 SLR(1)文法? 是否为 LALR(1)文法? 并构造相应的分析表。

答: (1)

Ch05-4-(1) 不是SLR(1)文法 Follow(T)={#,(,a,b,+),}

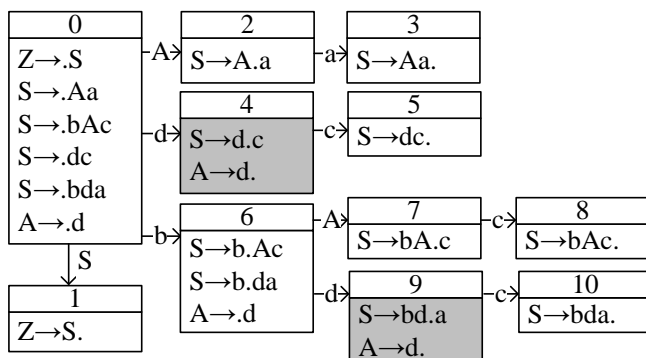


Ch05-4-(1) 不是LALR(1)文法

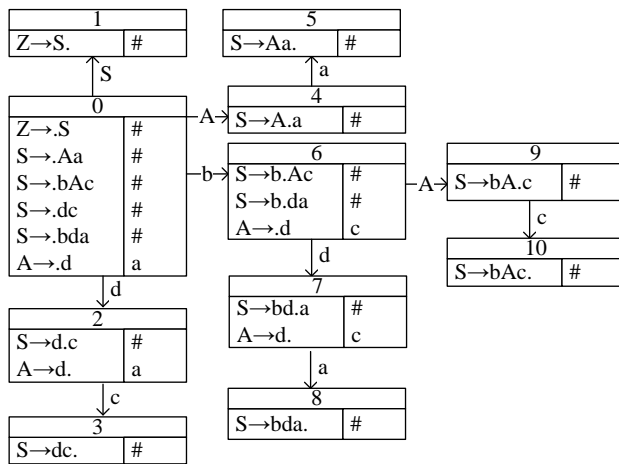


(2)

Ch05-4-(2) 不是SLR(1)文法 Follow(A)={a,c}



Ch05-4-(2) 是LALR(1)文法



	action					goto	
	a	b	c	d	#	A	S
$Z \rightarrow .S$ (1)							
$S \rightarrow .Aa$ (2)							
$S \rightarrow .bAc$ (3)							
$S \rightarrow .dc$ (4)							
$S \rightarrow .bda$ (5)							
$A \rightarrow .d$ (6)							
S0		s6		s2		4	1
S1					Acc		
S2	R6		s3				
S3					R4		
S4	s5						
S5					R2		
S6				s7		9	
S7	s8		R6				
S8					R5		
S9			s10				
S10					R3		

5. 设有下列文法:

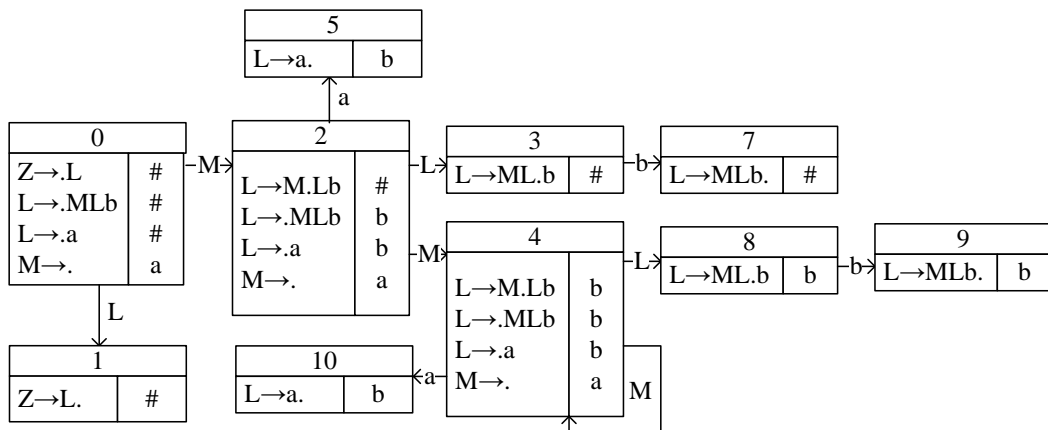
$L \rightarrow MLb \mid a$

$M \rightarrow \epsilon$

说明上述文法是否为 LR(1)文法, 若不是, 请说明理由。

答:

Ch05-5 是LR(1)文法



第六章

1. 试写出下列类型的内部表示:

a. integer

b. **ARRAY** [1..5] of **RECORD** i:integer; b:boolean **END**

c. **ARRAY** [1..5] of **RECORD** a:**ARRAY** [1..10]; r:**RECORD** i,j:integer **END** **END**

d. **RECORD** r: **RECORD** x,y:real **END**; a: **ARRAY** [1..10] of integer **END**

2. 设当前层数为 l, 可用区距为 101, 且有下列程序段:

CONST mm=333; nn=444;

TYPE atype = **ARRAY**[1..10] OF real;

 rtype = **RECORD** i,j:integer **END**;

VAR a,b:atype;

 x,y:real

试写出各标识符的内部表示。

3. 设当前层数和区距分别为 l 和 off, 且有函数说明首部:

FUNCTION f (A: atype; VAR B: atype; VAR X: real): integer

 其中 atype 的定义见题 5, 试写出 f 的内部表示。

4. 要求在下面括号中写上相应 l (层数) 和区距 (off)。

() () PROCEDURE g (A: atype; () ()

 VAR B: atype; () ()

 VAR X: real () () () () .

5. 给出下面 C 程序扫描到语句 c=a+b+x 时相应的全局符号表(采用顺序表结构)。

```
main ( )
{
    int a=0;
    float c=1.0;
    {
        float a=3.0;
        {
            float x=1.3;
            float b=0.3;
        }
        {
            int b=10;
            c=a+b+x;
        }
    }
}
```

6. 给出题 1 中程序扫描到语句 c=a+b+x 时相应的全局符号表(采用外拉链的散列表结构)。

7. 根据标识符的作用域规则，分别给出图 6.5 的程序中，过程 P、Q、R、S 中有效的标识符。

第七章

1. 将算术表达式 $(a+b)*(c+d)-(a+b+c)$ 翻译成四元式序列。
2. 将下列语句翻译成四元式序列：
 - a. IF $x > 0$ THEN $x := 0$ ELSE $x := 1$
 - b. WHILE $x > 0$ DO $x := x - 1$
 - c. IF $x > 0$ THEN $x := x + 1$ ELSE
IF $x < 0$ THEN $x := x + 1$ ELSE $x := x + 1$
 - d. WHILE $x > 0$ DO
WHILE $y > 0$ DO
BEGIN $y := y - x$; $x := x - 1$ END
3. 给出如下程序段的四元式序列。(四元式的操作符可用自身代替)。其中 A: Array of [1..10] of integer。

```
a:=1;
while a<=10 do
  begin if a<>b then
    A[a]:=A[b]+2;
  else a:=a+1;
  b:=b+1;
end
```
4. 试将 FOR 语句翻译成四元式序列。
5. 试将 UNTIL 语句翻译成四元式序列。
6. 试将 CASE 语句翻译成四元式序列。
7. 试给出 4、5、6 题中翻译过程的语法制导程序。

第八章

1. 将下面的程序段划分为基本块并画出其程序流图。

```
    read(A);
    read(B);
    F:=1;
    C:=A*A;
    D:=B*B;
    if C<D goto L1;
    E:=A*A;
    F:=F+1;
    E:=E+F;
    write(E);
    goto L3;
L1:  E:=B*B;
    F:=F+2;
    write(E);
    if E>100 goto L2;
    goto L3;
L2:  F:=F-1;
    goto L1;
L3:  write(E);
```

2. 假设有如下语句序列，写出常表达式优化前和优化后的四元式中间代码。

```
(1) i:=1;          (2) a:=20;
    j:=i*(i+1);      b:=a*(a+10);
    k:=2*(i+j);      c:=a*b;
```

3. 假设有如下语句序列或表达式，写出公共表达式优化前和优化后的四元式中间代码。

```
(1) x:=x*y+z; y:=x*y+z; z:=x*y+z;
(2) (a*b+c)/(a*b-c)+(c*b+a-d)/(a*b+c)
```

4. 写出如下循环语句不变式外提后的四元式中间代码。

```
while i<=100 do
    begin    u:=A*B;
            m:=u*u;
            S:=S+m*m;
            i:=i+1;
    end
```

5. 写出下面循环语句不变式外提后的四元式中间代码，其中数组各下标的类型为 1..10。

```
while i<=100 do
    begin    j:=1;
            while j<=100 do
```

```
begin    k:=1;
        while k<=100 do
            A[i][j][k]:=0;
        end
end
```


第九章

1.过程活动记录包含哪些信息？各信息的作用？何时填写它们？

2.下面是一个调用递归函数的 Pascal 程序

```
program PP(input,output)
  VAR k:integer;
  FUNCTION F(n:integer):integer
  begin
    if n<=0 then F:=1
    else F:=n*F(n-1);
  end;
begin
  k:=F(10);
  ...
end.
```

当第二次（递归地）进入 F 后，DISPLAY 的内容是什么？当时整个运行栈的内容是什么？

3.对于下面的程序：

```
procedure P(X,Y,Z);
begin
  Y:=Y+1;
  Z:=Z+X;
end P;
begin
  A:=2;
  B:=3;
  P(A+B,A,A);
  print A
end
```

当参数传递的办法分别为（1）传值；（2）传地址；（3）值-结果；（4）传名时，程序执行时输出的 A 分别是什么？

4.应用 Pascal 语言的作用域规则，说明下面程序中的名字 a 和 b 的每一次出现所应用的声明。

```
program a(input,output);
  procedure b(u,v,x,y:integer);
    var a:record a,b:integer end;
    b:reocrd b,a:integer end;
  begin
    with a do begin a:=u;b:=v end;
    with b do begin a:=x;b:=y end;
    writeln(a.a,a.b,b.a,b.b)
  end;
begin
  b(1,2,3,4)
end.
```

5.为下面的 C 程序构造一个可能的运行时环境。

```
int a[10];
char *a=" hello" ;
int f(int i,int b[])
{  int j=1;
  A: { int i=j;
      char c=b[l];
      ... }
}
```

```

void g(char *s)
{ char c[10];
  B: { int a[5];
      ... }
}
main()
{ int x=1;
  x=f(x,a);
  g(s);
  return 0; }

```

(1) 在进入函数 **f** 中的块 **A** 之后。

(2) 在进入函数 **g** 中的块 **B** 之后。

6. **Display** 表和静态链的作用是什么？试举一个程序例子，并考察其 **Display** 表和静态链的内容。

7. 过程参数的传递方式有几种？简述“传地址”和“传值”的实现原理。

第十章

1. 什么叫指令的执行代价？
2. 寄存器分配的原则是什么？
3. 在剥夺寄存器的时候，应考虑哪些因素？什么时候一个变量可以主动释放它所占用的寄存器？
4. 试写出程序段

```
IF x > 0 THEN y:=y+1 ELSE
```

```
IF x < 0 THEN y:=y-1
```

的目标代码，其中的变量均为非形参实型变量。

5. 试写出程序段

```
WHILE x < y DO
```

```
BEGIN y := y + 1;
```

```
IF y > 0 THEN y := y-x ELSE
```

```
WHILE y < 0 DO y := y + x
```

```
END
```

的目标代码，其中变量均为非形参实型变量。

6. 试为 FOR 循环语句设计目标代码。
7. 试为 REPEAT 循环语句设计目标代码。
8. 试为 CASE 语句设计目标代码。

