第二章

P36-6

(1)

$L(G_1)$是 0~9 组成的数字串

(2)

最左推导:

$N \Rightarrow ND \Rightarrow NDD \Rightarrow NDDD \Rightarrow DDDD \Rightarrow 0DDD \Rightarrow 01DD \Rightarrow 012D \Rightarrow 0127$

$N \Rightarrow ND \Rightarrow DD \Rightarrow 3D \Rightarrow 34$

$N \Rightarrow ND \Rightarrow NDD \Rightarrow DDD \Rightarrow 5DD \Rightarrow 56D \Rightarrow 568$

最右推导:

$N \Rightarrow ND \Rightarrow N7 \Rightarrow ND7 \Rightarrow N27 \Rightarrow ND27 \Rightarrow N127 \Rightarrow D127 \Rightarrow 0127$

$N \Rightarrow ND \Rightarrow N4 \Rightarrow D4 \Rightarrow 34$

$N \Rightarrow ND \Rightarrow N8 \Rightarrow ND8 \Rightarrow N68 \Rightarrow D68 \Rightarrow 568$

P36-7

G(S)

$O \rightarrow 1|3|5|7|9$

$N \rightarrow 2|4|6|8|O$

$D \rightarrow 0|N$

$S \rightarrow O|AO$

$A \rightarrow AD|N$

P36-8

文法:

$E \rightarrow T|E+T|E-T$
$T \rightarrow F|T*F|T/F$
$F \rightarrow (E)|i$

最左推导:

$E \Rightarrow E+T \Rightarrow T+T \Rightarrow F+T \Rightarrow i+T \Rightarrow i+T*F \Rightarrow i+F*F \Rightarrow i+i*F \Rightarrow i+i*i$

$E \Rightarrow T \Rightarrow T*F \Rightarrow F*F \Rightarrow i*F \Rightarrow i*(E) \Rightarrow i*(E+T) \Rightarrow i*(T+T) \Rightarrow i*(F+T)$

$\Rightarrow i*(i+T) \Rightarrow i*(i+F) \Rightarrow i*(i+i)$

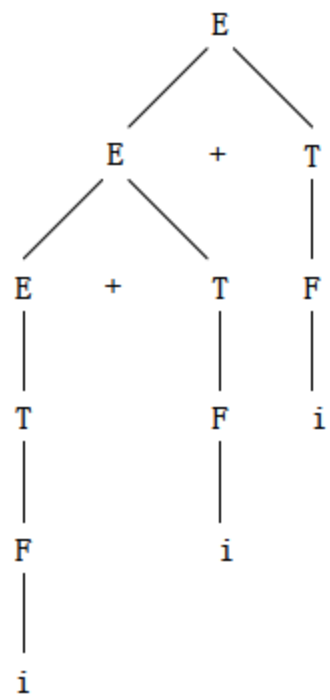
最右推导:

$E \Rightarrow E+T \Rightarrow E+T*F \Rightarrow E+T*i \Rightarrow E+F*i \Rightarrow E+i*i \Rightarrow T+i*i \Rightarrow F+i*i \Rightarrow i+i*i$

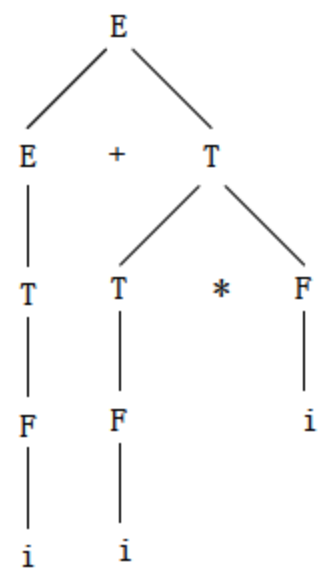
$E \Rightarrow T \Rightarrow F*T \Rightarrow F*F \Rightarrow F*(E) \Rightarrow F*(E+T) \Rightarrow F*(E+F) \Rightarrow F*(E+i)$

$\Rightarrow F*(T+i) \Rightarrow F*(F+i) \Rightarrow F*(i+i) \Rightarrow i*(i+i)$

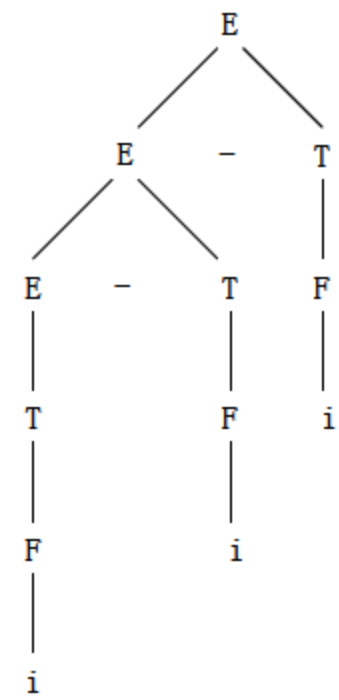
语法树：/**********



i+i+i



i-i-i



i+i*i

**********/

P36-9

句子 iiiei 有两个语法树：

$S \Rightarrow iSeS \Rightarrow iSei \Rightarrow iiSei \Rightarrow iiiei$
$S \Rightarrow iS \Rightarrow iiSeS \Rightarrow iiSei \Rightarrow iiiei$

P36-10

/**********

$S \rightarrow TS \mid T$

$T \rightarrow (S) \mid ( )$

**********/

P36-11

/**********

L1：

$S \rightarrow AC$

$A \rightarrow aAb \mid ab$

$C \rightarrow cC \mid \varepsilon$

L2：

$S \rightarrow AB$

$A \rightarrow aA \mid \varepsilon$

$B \rightarrow bBc \mid bc$

L3：

$S \to AB$

$A \to aAb \mid \varepsilon$

$B \to aBb \mid \varepsilon$

L4:

$S \to A \mid B$

$A \to 0A1 \mid \varepsilon$

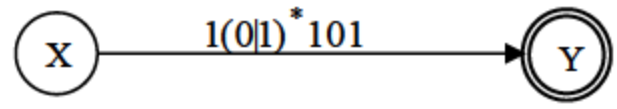
$B \to 1B0 \mid A$

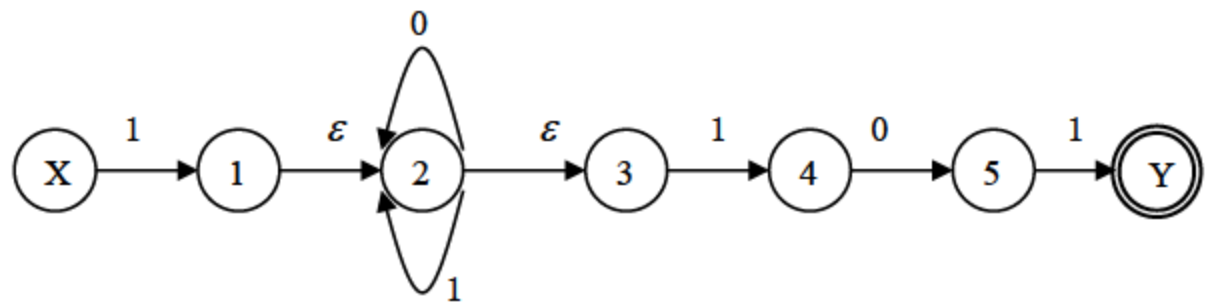
**********/

第三章习题参考答案

P64 - 7

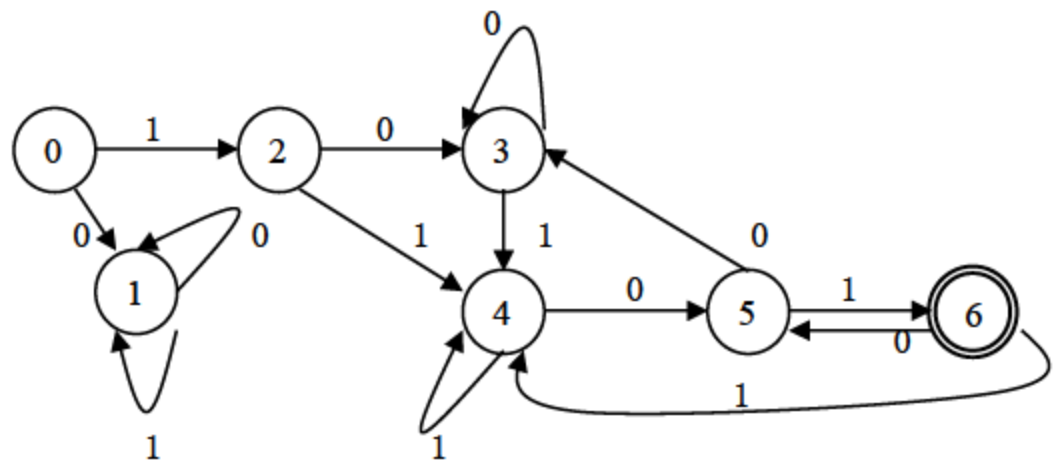
(1)





确定化：

	0	1
{X}	Φ	{1, 2, 3}
Φ	Φ	Φ
{1, 2, 3}	{2, 3}	{2, 3, 4}
{2, 3}	{2, 3}	{2, 3, 4}
{2, 3, 4}	{2, 3, 5}	{2, 3, 4}
{2, 3, 5}	{2, 3}	{2, 3, 4, Y}
{2, 3, 4, Y}	{2, 3, 5}	{2, 3, 4, }



最小化：

$\{0,1,2,3,4,5\},\{6\}$

$\{0,1,2,3,4,5\}_0 = \{1,3,5\}$ $\{0,1,2,3,4,5\}_1 = \{1,2,4,6\}$

$\{0,1,2,3,4\},\{5\},\{6\}$

$\{0,1,2,3,4\}_0 = \{1,3,5\}$

$\{0,1,2,3\},\{4\},\{5\},\{6\}$

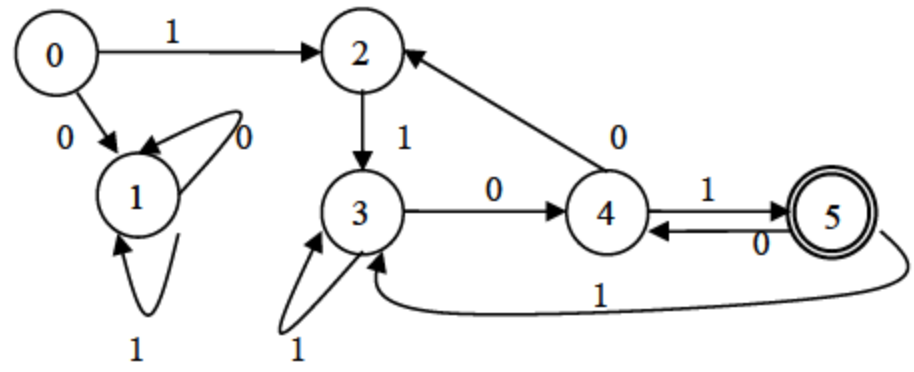
$\{0,1,2,3\}_0 = \{1,3\}$ $\{0,1,2,3\}_1 = \{1,2,4\}$

$\{0,1\},\{2,3\}\{4\},\{5\},\{6\}$

$\{0,1\}_0 = \{1\}$ $\{0,1\}_1 = \{1,2\}$

$\{2,3\}_0 = \{3\}$ $\{2,3\}_1 = \{4\}$

$\{0\},\{1\},\{2,3\},\{4\},\{5\},\{6\}$



P64 - 8

(1)

$(1|0)^*01$

(2)

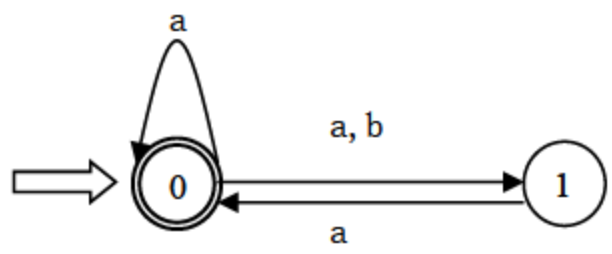
$(1|2|3|4|5|6|7|8|9)(0|1|2|3|4|5|6|7|8|9)^*(0|5)|(0|5)$

(3)

$0^*1(0|10^*1)^*|1^*0(0|10^*1)^*$

P64 - 12

(a)

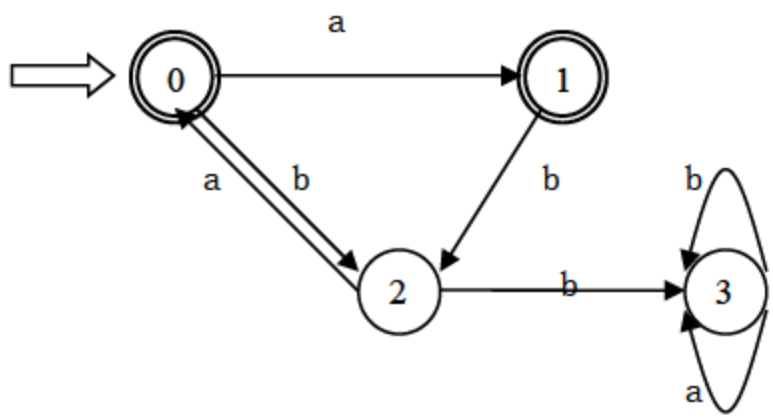


确定化：

	a	b
{0}	{0, 1}	{1}
{0, 1}	{0, 1}	{1}
{1}	{0}	ϕ
ϕ	ϕ	ϕ

给状态编号：

	a	b
0	1	2
1	1	2
2	0	3
3	3	3



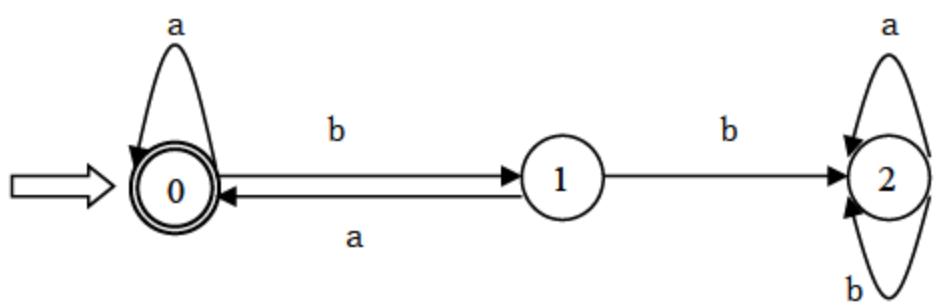
最小化：

$\{0,1\}, \{2,3\}$

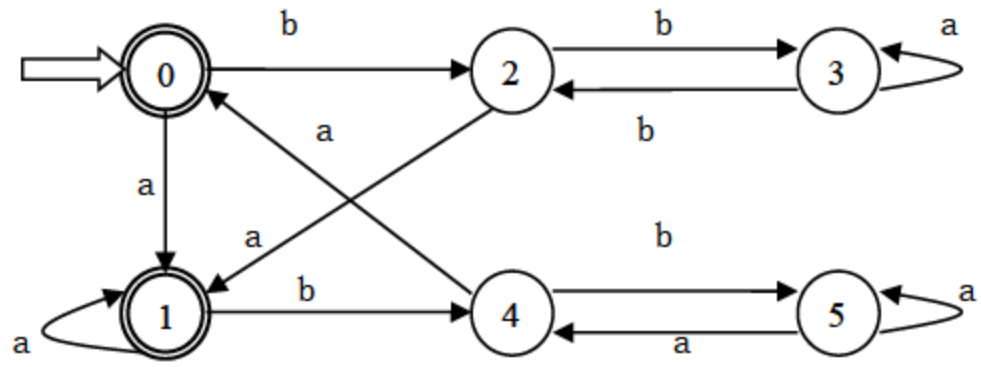
$\{0,1\}_a = \{1\}$ $\{0,1\}_b = \{2\}$

$\{2,3\}_a = \{0,3\}$ $\{2,3\}_b = \{3\}$

$\{0,1\}, \{2\}, \{3\}$



(b)



已经确定化了，进行最小化

最小化：

$\{\{0,1\},\ \{2,3,4,5\}\}$

$\{0,1\}_a = \{1\}$ $\{0,1\}_b = \{2,4\}$

$\{2,3,4,5\}_a = \{1,3,0,5\}$ $\{2,3,4,5\}_b = \{2,3,4,5\}$

$\{2,4\}_a = \{1,0\}$ $\{2,4\}_b = \{3,5\}$

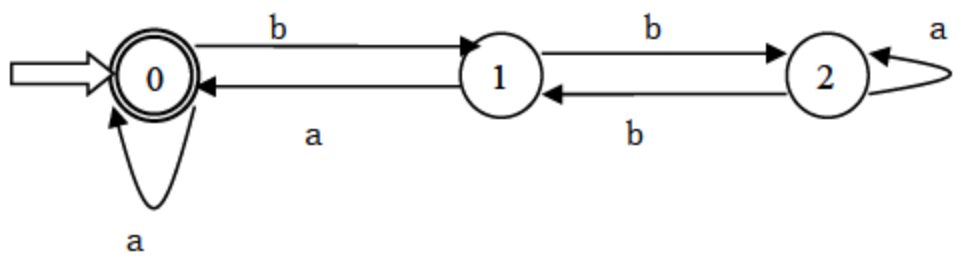
$\{3,5\}_a = \{3,5\}$ $\{3,5\}_b = \{2,4\}$

$\{\{0,1\},\{2,4\},\{3,5\}\}$

$\{0,1\}_a = \{1\}$ $\{0,1\}_b = \{2,4\}$

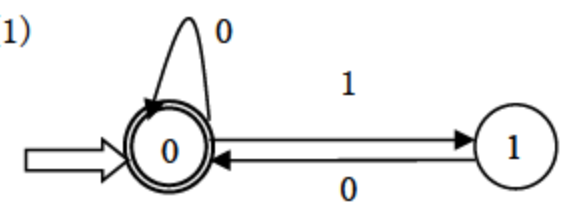
$\{2,4\}_a = \{1,0\}$ $\{2,4\}_b = \{3,5\}$

$\{3,5\}_a = \{3,5\}$ $\{3,5\}_b = \{2,4\}$

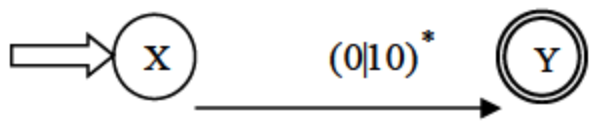


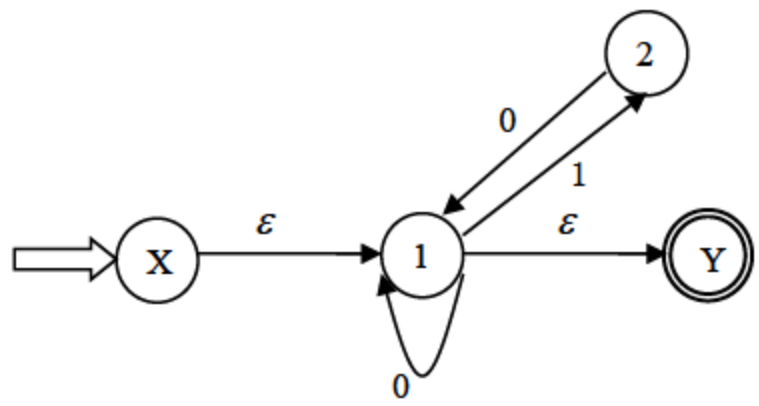
P64 - 14

(1)



(2):





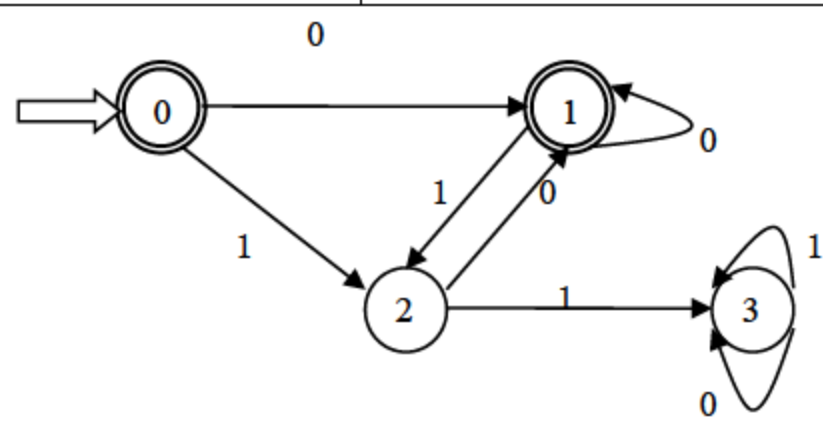
确定化:

	0	1
{X, 1, Y}	{1, Y}	{2}
{1, Y}	{1, Y}	{2}

{2}	{1, Y}	ϕ
ϕ	ϕ	ϕ

给状态编号：

	0	1
0	1	2
1	1	2
2	1	3
3	3	3



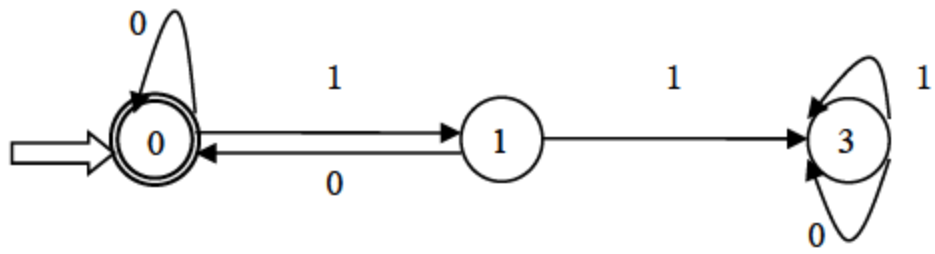
最小化：

$\{0,1\},\{2,3\}$

$\{0,1\}_0 = \{1\}$　　$\{0,1\}_1 = \{2\}$

$\{2,3\}_0 = \{1,3\}$　　$\{2,3\}_1 = \{3\}$

$\{0,1\},\{2\},\{3\}$



第四章

P81 - 1

(1) 按照 T,S 的顺序消除左递归

$G'(S)$

$S \to a \mid \hat{} \mid (T)$

$T \to ST'$

$T' \to ,ST' \mid \varepsilon$

递归子程序：

```
procedure S;
begin
    if sym='a' or sym='^'
        then abvance
        else if sym='('
            then begin
                advance;T;
                if sym=')' then advance;
                    else error;
                end
            else  error
end;
procedure T;
begin
    S;T'
end;
procedure T';
begin
    if sym=','
        then begin
            advance;
            S;T'
        end
end;
```

其中：

sym:是输入串指针 IP 所指的符号

advance:是把 IP 调至下一个输入符号

error:是出错诊察程序

(2)

FIRST(S)={a,^,(}

FIRST(T)={a,^,(}

FIRST(T')={,,ε}

FOLLOW(S)={),,,#}

FOLLOW(T)={)}

FOLLOW(T')={)}

预测分析表

	a	^	(	)	,	#
S	$S \to a$	$S \to ^$	$S \to (T)$			
T	$T \to ST'$	$T \to ST'$	$T \to ST'$			
T'				$T' \to \varepsilon$	$T' \to ,ST'$	

是 LL(1) 文法

P81 - 2

文法：

$E \to TE'$

$E' \to +E \mid \varepsilon$

$T \to FT'$

$T' \to T \mid \varepsilon$

$F \to PF'$

$F' \to *F' \mid \varepsilon$

$P \to (E) \mid a \mid b \mid ^$

(1)

FIRST(E)={(, a, b, ^}

FIRST(E')={+, ε}

FIRST(T)={(, a, b, ^}

FIRST(T')={(, a, b, ^, ε}

FIRST(F)={(, a, b, ^}

FIRST(F')={*, ε}

FIRST(P)={(, a, b, ^}

FOLLOW(E)={#,)}

FOLLOW(E')={#,)}

FOLLOW(T)={+,), #}

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

FOLLOW(F)={(, a, b, ^, +,), #}

FOLLOW(F')={(, a, b, ^, +,), #}

FOLLOW(P)={*, (, a, b, ^, +,), #}

(2)

考虑下列产生式：

$E' \to +E \mid \varepsilon$

$T' \to T \mid \varepsilon$

$F' \to *F' \mid \varepsilon$

$P \to (E) \mid ^ \mid a \mid b$

FIRST(+E) ∩ FIRST(ε)={+} ∩ { ε }= ϕ

FIRST(+E) ∩ FOLLOW(E')={+} ∩ {#,)}= ϕ

FIRST(T) ∩ FIRST(ε)={(, a, b, ^} ∩ { ε }= ϕ

FIRST(T)∩FOLLOW(T')={(, a, b, ^}∩{+,), #}=ϕ

FIRST(*F')∩FIRST(ε)={*}∩{ε}=ϕ

FIRST(*F')∩FOLLOW(F')={*}∩{(, a, b, ^, +,), #}=ϕ

FIRST((E))∩FIRST(a) ∩FIRST(b) ∩FIRST(^)=ϕ

所以,该文法式LL(1)文法.

(3)

	+	*	(	)	a	b	^	#
E			$E \to TE'$		$E \to TE'$	$E \to TE'$	$E \to TE'$	
E'	$E' \to +E$			$E' \to \varepsilon$				$E' \to \varepsilon$
T			$T \to FT'$		$T \to FT'$	$T \to FT'$	$T \to FT'$	
T'	$T' \to \varepsilon$		$T' \to T$	$T' \to \varepsilon$	$T' \to T$	$T' \to T$	$T' \to T$	$T' \to \varepsilon$
F			$F \to PF'$		$F \to PF'$	$F \to PF'$	$F \to PF'$	
F'	$F' \to \varepsilon$	$F' \to *F'$	$F' \to \varepsilon$	$F' \to \varepsilon$	$F' \to \varepsilon$	$F' \to \varepsilon$	$F' \to \varepsilon$	$F' \to \varepsilon$
P			$P \to (E)$		$P \to a$	$P \to b$	$P \to ^$	

(4)

```
procedure E;
begin
    if sym='(' or sym='a' or sym='b' or sym='^'
        then begin T; E' end
        else error
end
procedure E';
begin
    if sym='+'
        then begin advance; E end
        else if sym<>')' and sym<>'#' then error
end
procedure T;
begin
    if sym='(' or sym='a' or sym='b' or sym='^'
        then begin F; T' end
        else error
end
procedure T';
begin
    if sym='(' or sym='a' or sym='b' or sym='^'
        then T
        else if sym='*' then error
end
procedure F;
begin
    if sym='(' or sym='a' or sym='b' or sym='^'
        then begin P; F' end
        else error
```

```
end
procedure F';
begin
    if sym='*'
        then begin advance; F' end
end
procedure P;
begin
    if sym='a' or sym='b' or sym='^'
        then advance
        else if sym='(' then
        begin
            advance; E;
            if sym=')' then advance
                 else error
        end
        else error
end;
```

P81 - 3

/***************

(1) 是，满足三个条件。

(2) 不是，对于 A 不满足条件 3。

(3) 不是，A、B 均不满足条件 3。

(4) 是，满足三个条件。

***************/

第五章

P133 - 1

$E \Rightarrow E+T \Rightarrow E+T*F$

短语： E+T*F， T*F，

直接短语：T*F

句柄：T*F

P133 - 2

文法：

$S \rightarrow a|\hat{}|(T)$
$T \rightarrow T,S|S$

(1)

最左推导：

$S \Rightarrow (T) \Rightarrow (T,S) \Rightarrow (S,S) \Rightarrow (a,S) \Rightarrow (a,(T)) \Rightarrow (a,(T,S)) \Rightarrow (a,(S,S)) \Rightarrow (a,(a,S)) \Rightarrow (a,(a,a))$

$S \Rightarrow (T,S) \Rightarrow (S,S) \Rightarrow ((T),S) \Rightarrow ((T,S),S) \Rightarrow ((T,S,S),S) \Rightarrow ((S,S,S),S) \Rightarrow (((T),S,S),S)$
$\Rightarrow (((T,S),S,S)),S) \Rightarrow (((S,S),S,S),S) \Rightarrow (((a,S),S,S),S) \Rightarrow (((a,a),S,S),S)$
$\Rightarrow (((a,a),\hat{},S),S) \Rightarrow (((a,a),\hat{},(T)),S) \Rightarrow (((a,a),\hat{},(S)),S) \Rightarrow (((a,a),\hat{},(a)),S)$
$\Rightarrow (((a,a),\hat{},(a)),a)$

最右推导：

$S \Rightarrow (T) \Rightarrow (T,S) \Rightarrow (T,(T)) \Rightarrow (T,(T,S)) \Rightarrow (T,(T,a)) \Rightarrow (T,(S,a)) \Rightarrow (T,(a,a))$
$\Rightarrow (S,(a,a)) \Rightarrow (a,(a,a))$

$S \Rightarrow (T,S) \Rightarrow (T,a) \Rightarrow (S,a) \Rightarrow ((T),a) \Rightarrow ((T,S),a) \Rightarrow ((T,(T)),a) \Rightarrow ((T,(S)),a)$
$\Rightarrow ((T,(a)),a) \Rightarrow ((T,S,(a)),a) \Rightarrow ((T,\hat{},(a)),a) \Rightarrow ((S,\hat{},(a)),a) \Rightarrow (((T),\hat{},(a)),a)$
$\Rightarrow (((T,S),\hat{},(a)),a) \Rightarrow (((T,a),\hat{},(a)),a) \Rightarrow (((S,a),\hat{},(a)),a) \Rightarrow (((a,a),\hat{},(a)),a)$

(2)

((**a**, a), ^, (a)), a)
((**S**, a), ^, (a)), a)
((T, **a**), ^, (a)), a)
((**T, S**), ^, (a)), a)
((**(T)**, ^, (a)), a)
(**S**, ^, (a)), a)
(T, ^, (a)), a)
(**T, S**, (a)), a)
(T, (**a**)), a)
(T, (**S**)), a)
(T, (**T**)), a)
(**T, S**), a)
(T), a)
S, a)
T, S)
(T)

S

“移进-归约”过程：

步骤	栈	输入串	动作
0	#	(((a, a), ^, (a)), a)#	预备
1	#(	((a, a), ^, (a)), a)#	进
2	#((	(a, a), ^, (a)), a)#	进
3	#(((	a, a), ^, (a)), a)#	进
4	#(((a	, a), ^, (a)), a)#	进
5	#(((S	, a), ^, (a)), a)#	归
6	#(((T	, a), ^, (a)), a)#	归
7	#(((T,	a), ^, (a)), a)#	进
8	#(((T, a	), ^, (a)), a)#	进
9	#(((T, S	), ^, (a)), a)#	归
10	#(((T	), ^, (a)), a)#	归
11	#(((T)	, ^, (a)), a)#	进
12	#((S	, ^, (a)), a)#	归
13	#((T	, ^, (a)), a)#	归
14	#((T,	^, (a)), a)#	进
15	#((T, ^	, (a)), a)#	进
16	#((T, S	, (a)), a)#	归
17	#((T	, (a)), a)#	归
18	#((T,	(a)), a)#	进
19	#((T, (	a)), a)#	进
20	#((T, (a	)), a)#	进
21	#((T, (S	)), a)#	归
22	#((T, (T	)), a)#	归
23	#((T, (T)	), a)#	进
24	#((T, S	), a)#	归
25	#((T	), a)#	归
26	#((T)	, a)#	进
27	#(S	, a)#	归
28	#(T	, a)#	归
29	#(T,	a)#	进
30	#(T, a	)#	进
31	#(T, S	)#	归
32	#(T	)#	归
33	#(T)	#	进
34	#S	#	归

P133 - 3

(1)

FIRSTVT(S)={a, ^, (}

FIRSTVT(T)={, , a, ^, (}

LASTVT(S)={a, ^,)}

LASTVT(T)={,,a,^,)}

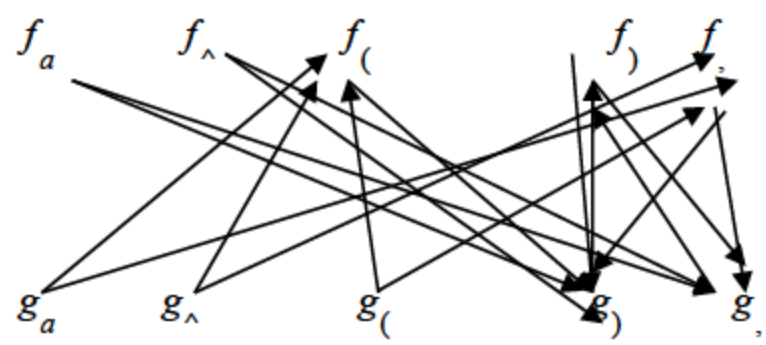
(2)

	a	^	(	)	,
a				>	>
^				>	>
(	<	<	<	=	<
)				>	>
,	<	<	<	>	>

G_6是算符文法，并且是算符优先文法

(3)优先函数

	a	^	(	)	,
f	4	4	2	4	4
g	5	5	5	2	3



（4）

栈	输入字符串	动作
#	（a, (a, a)）#	预备
#(	a, (a, a))#	进
#(a	, (a, a))#	进
#(t	, (a, a))#	归
#（t,	(a, a)）#	进
#（t, （	a, a)）#	进
#（t, （a	, a)）#	进
#（t, （t	, a)）#	归
#（t, （t,	a)）#	进
#（t, （t, a	))#	进
#（t, （t, s	))#	归
#（t, （t	))#	归
#（t, （t)	)#	进
#（t, s	)#	归
#（t	)#	归
#（t)	#	进
# s	#	归
success		

P134 - 5

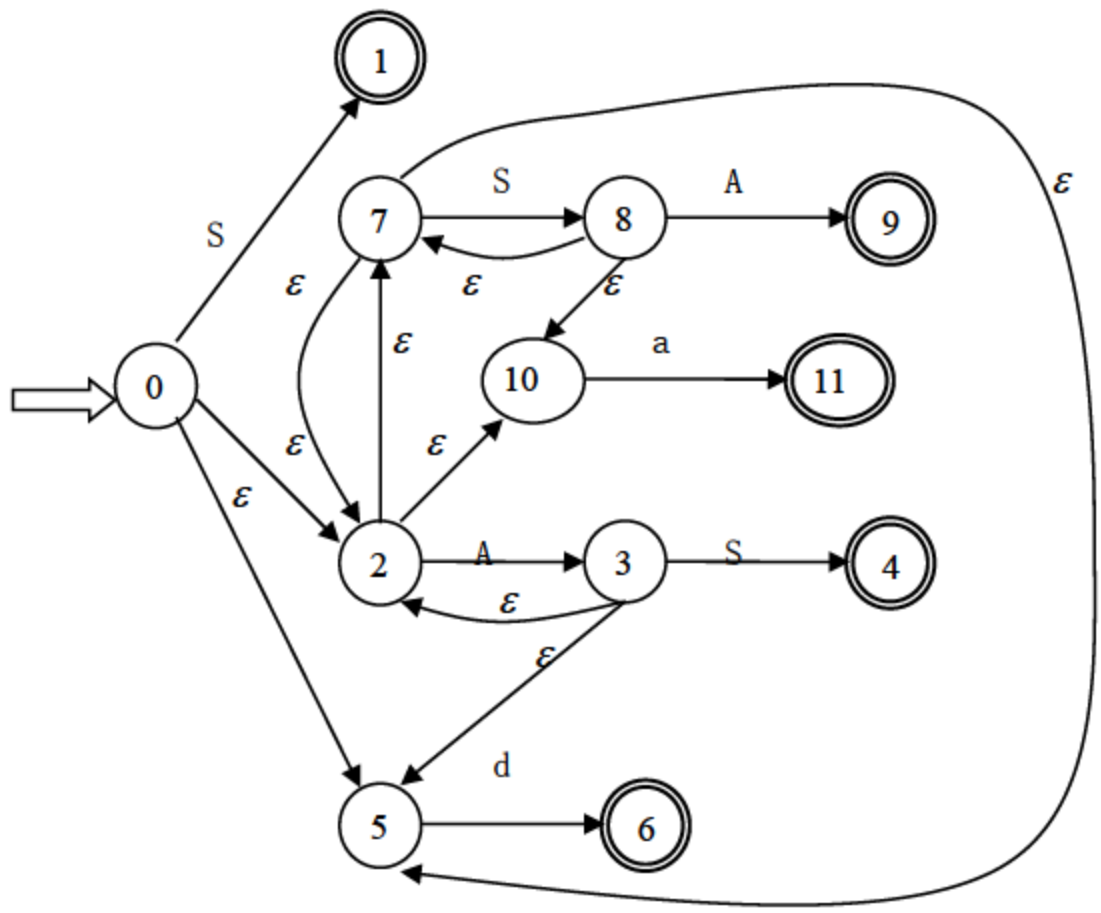
(1)

0. $S' \to \cdot S$ 1. $S' \to S \cdot$ 2. $S \to \cdot AS$ 3. $S \to A \cdot S$

4. $S \to AS \cdot$ 5. $S \to \cdot b$ 6. $S \to b \cdot$ 7. $A \to \cdot SA$

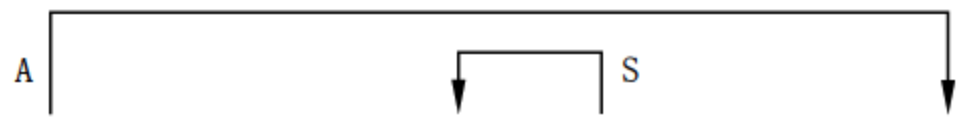
8. $A \to S \cdot A$ 9. $A \to SA \cdot$ 10. $A \to \cdot a$ 11. $A \to a \cdot$

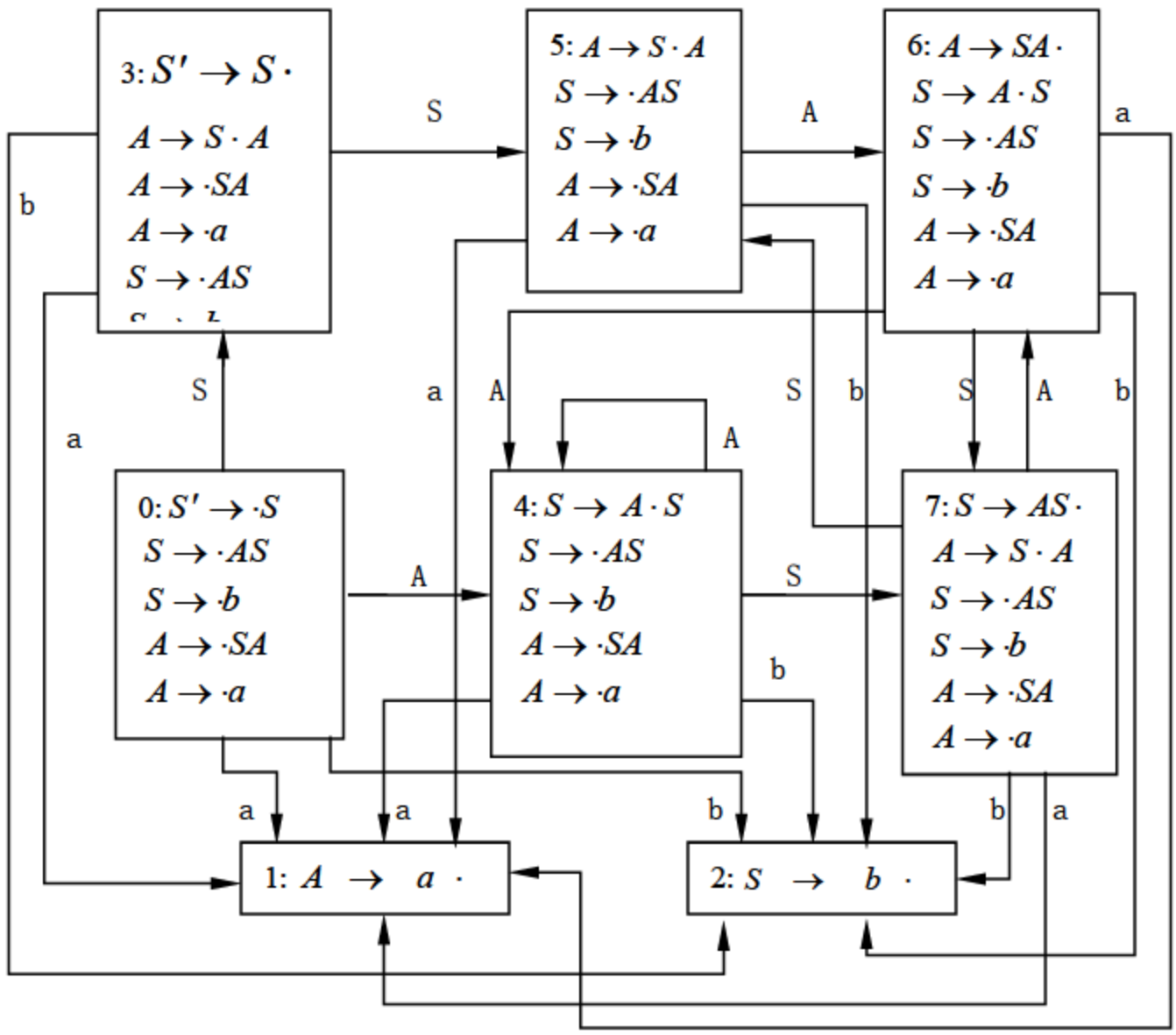
(2)



确定化：

	S	A	a	b
{0, 2, 5, 7, 10}	{1, 2, 5, 7, 8, 10 }	{2, 3, 5, 7, 10}	{11}	{6}
{1, 2, 5, 7, 8, 10 }	{2, 5, 7, 8, 10}	{2, 3, 5, 7, 9, 10 }	{11}	{6}
{2, 3, 5, 7, 10}	{2, 4, 5, 7, 8, 10 }	{2, 3, 5, 7, 10}	{11}	{6}
{2, 5, 7, 8, 10}	{2, 5, 7, 8, 10}	{2, 3, 5, 7, 9, 10 }	{11}	{6}
{2, 3, 5, 7, 9, 10 }	{2, 4, 5, 7, 8, 10 }	{2, 3, 5, 7, 10}	{11}	{6}
{2, 4, 5, 7, 8, 10 }	{2, 5, 7, 8, 10}	{2, 3, 5, 7, 9, 10 }	{11}	{6}
{11}	Φ	Φ	Φ	Φ
{6}	Φ	Φ	Φ	Φ





DFA

构造 LR(0)项目集规范族也可以用 GO 函数来计算得到。所得到的项目集规范族与上图中的项目集一样:

I_0={ $S' \to \cdot S$, $S \to \cdot AS$, $S \to \cdot b$, $A \to \cdot SA$, $A \to \cdot a$ }

GO(I_0, a)={ $A \to a \cdot$ }=I_1

GO(I_0, b)={ $S \to b \cdot$ }=I_2

GO(I_0, S)={ $S' \to S \cdot$, $A \to S \cdot A$, $A \to \cdot SA$, $A \to \cdot a$, $S \to \cdot AS$, $S \to \cdot b$ }=I_3

GO(I_0, A)={ $S \to A \cdot S$, $S \to \cdot AS$, $S \to \cdot b$, $A \to \cdot SA$, $A \to \cdot a$ }=I_4

GO(I_3, a)={ $A \to a \cdot$ }=I_1

GO(I_3, b)={ $S \to b \cdot$ }=I_2

GO(I_3, S)={ $A \to S \cdot A$, $S \to \cdot AS$, $S \to \cdot b$, $A \to \cdot SA$, $A \to \cdot a$ }=I_5

GO(I_3, A)={ $A \to SA \cdot$, $S \to A \cdot S$, $S \to \cdot AS$, $S \to \cdot b$, $A \to \cdot SA$, $A \to \cdot a$ }=I_6

GO(I_4, a)={ $A \to a \cdot$ }=I_1

GO(I_4, b)={ $S \to b \cdot$ }=I_2

GO(I_4, S)={ $S \to AS \cdot$, $A \to S \cdot A$, $S \to \cdot AS$, $S \to \cdot b$, $A \to \cdot SA$, $A \to \cdot a$ }=I_7

GO(I_4, A)={ $S \to A \cdot S$, $S \to \cdot AS$, $S \to \cdot b$, $A \to \cdot SA$, $A \to \cdot a$ }=I_4

GO(I_5, a)={ $A \to a \cdot$ }=I_1

GO(I_5, b)={ $S \to b \cdot$ }=I_2

GO(I_5, S)={ $A \to S \cdot A$, $S \to \cdot AS$, $S \to \cdot b$, $A \to \cdot SA$, $A \to \cdot a$ }=I_5

GO(I_5, A)={ $A \to SA \cdot$, $S \to A \cdot S$, $S \to \cdot AS$, $S \to \cdot b$, $A \to \cdot SA$, $A \to \cdot a$ }=I_6

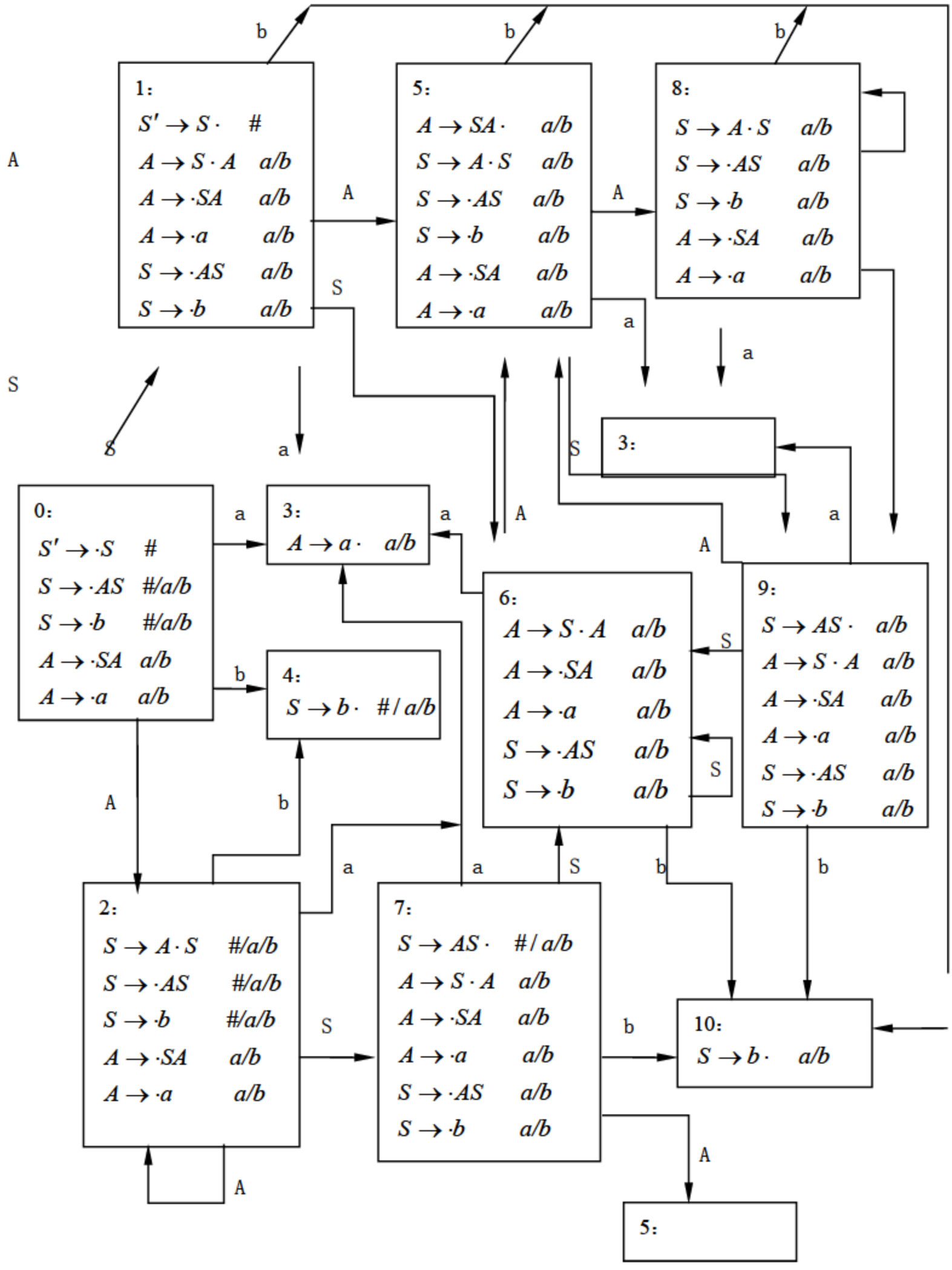
GO(I_6, a)={ $A \to a \cdot$ }=I_1

GO(I_6, b)={ $S \to b \cdot$ }=I_2

GO(I_6，S)={ $S \to AS\cdot$， $A \to S\cdot A$， $S \to \cdot AS$， $S \to \cdot b$， $A \to \cdot SA$， $A \to \cdot a$ }=I_7
GO(I_6，A)={ $S \to A\cdot S$， $S \to \cdot AS$， $S \to \cdot b$， $A \to \cdot SA$， $A \to \cdot a$ }=I_4
GO(I_7，a)={ $A \to a \cdot$ }=I_1
GO(I_7，b)={ $S \to b \cdot$ }=I_2
GO(I_7，S)={ $A \to S\cdot A$， $S \to \cdot AS$， $S \to \cdot b$， $A \to \cdot SA$， $A \to \cdot a$ }=I_5
GO(I_7，A)={ $A \to SA\cdot$， $S \to A\cdot S$， $S \to \cdot AS$， $S \to \cdot b$， $A \to \cdot SA$， $A \to \cdot a$ }=I_6

项目集规范族为C={I_1，I_2，I_3，I_4，I_5，I_6，I_7}

(3)不是SLR文法
状态3，6，7有移进归约冲突
状态3：FOLLOW(S')={#}不包含a,b
状态6：FOLLOW(S)={#,a,b}包含a,b,；移进归约冲突无法消解
状态7：FOLLOW(A)={a,b}包含a,b；移进归约冲突消解
所以不是SLR文法。
(4) 构造例如LR(1)项目集规范族
见下图：
对于状态5,因为包含项目[$A \to AS\cdot \quad a/b$],所以遇到搜索符号a或b时,应该用 $A \to AS$ 归约。又因为状态5包含项目[$A \to \cdot a \quad a/b$]，所以遇到搜索符号a时，应该移进。因此存在“移进-归约”矛盾，所以这个文法不是LR(1)文法。


1:
S′ → S· #
A → S·A a/b
A → ·SA a/b
A → ·a a/b
S → ·AS a/b
S → ·b a/b
5:
A → SA· a/b
S → A·S a/b
S → ·AS a/b
S → ·b a/b
A → ·SA a/b
A → ·a a/b
8:
S → A·S a/b
S → ·AS a/b
S → ·b a/b
A → ·SA a/b
A → ·a a/b
0:
S′ → ·S #
S → ·AS #/a/b
S → ·b #/a/b
A → ·SA a/b
A → ·a a/b
3:
A → a· a/b
3:
4:
S → b· #/a/b
6:
A → S·A a/b
A → ·SA a/b
A → ·a a/b
S → ·AS a/b
S → ·b a/b
9:
S → AS· a/b
A → S·A a/b
A → ·SA a/b
A → ·a a/b
S → ·AS a/b
S → ·b a/b
2:
S → A·S #/a/b
S → ·AS #/a/b
S → ·b #/a/b
A → ·SA a/b
A → ·a a/b
7:
S → AS· #/a/b
A → S·A a/b
A → ·SA a/b
A → ·a a/b
S → ·AS a/b
S → ·b a/b
10:
S → b· a/b
5:
A
S
a
b

第六章

/**********第六章会有点难

P164 - 5

(1)

E→E1＋T {if (E1.type = int) and (T.type = int)
then E.type := int
else E.type := real}

E→T {E.type := T.type}

T→num.num {T.type := real}

T→num {T.type := int}

(2)

P164 - 7

S→L1|L2 {S.val:=L1.val+(L2.val/$2^{L2.length}$)}

S→L {S.val:=L.val}

L→L1B {L.val:=2*L1.val + B.val;
L.length:=L1.length+1}

L→B {L.val:=B.c;
L.length :=1}

B→0 {B.c:=0}

B→1 {B.c:=1}

**********/

第七章

P217 - 1

a*(-b+c)　　ab@c+*

a+b*(c+d/e)　　abcde/+*+

-a+b*(-c+d)　　a@bc@d+*+

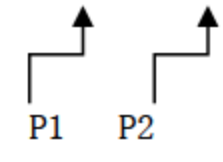
$\neg A \vee \neg(C \vee \neg D)$　　$A\neg CD\neg \vee \neg \vee$

$(A \wedge B) \vee (\neg C \vee D)$　　$AB \wedge C@D \vee \vee$

$(A \vee B) \wedge (C \vee \neg D \wedge E)$　　$AB \vee CD@E \wedge \vee \wedge$

if　(x+y)*z =0　then　(a+b)↑c　else　a↑b↑c　　xy+z*0= ab+c↑abc↑↑　￥

或 xy+z*0= P1 jez ab+c↑ P2 jump abc↑↑



P217 - 3

-(a+b)*(c+d)-(a+b+c)的

三元式序列:

(1) +, a, b

(2) @, (1), -

(3) +, c, d

(4) *, (2), (3)

(5) +, a, b

(6) +, (5), c

(7) -, (4), (6)

间接三元式序列:

三元式表:

(1) +, a, b

(2) @, (1), -

(3) +, c, d

(4) *, (2), (3)

(5) +, (1), c

(6) -, (4), (5)

间接码表:

(1)

(2)

(3)

(4)

(1)

(5)

(6)

四元式序列:

(1) +, a, b, T_1

(2) @, T_1, -, T_2

(3) +, c, d, T_3

(4) *, T_2, T_3, T_4

(5) +, a, b, T_5

(6) +, T_5, c, T_6

(7) -, T_4, T_6, T_7

P218 - 4

自下而上分析过程中把赋值句翻译成四元式的步骤:A:=B*(-C+D)

步骤	输入串	栈	PLACE	四元式
(1)	A:=B*(-C+D)			
(2)	:=B*(-C+D)	i	A	
(3)	B*(-C+D)	i:=	A-	
(4)	*(-C+D)	i:=i	A-B	
(5)	*(-C+D)	i:=E	A-B	
(6)	*(-C+D)	i:=E	A-B	
(7)	(-C+D)	i:=E*	A-B-	
(8)	-C+D)	i:=E*(	A-B--	
(9)	C+D)	i:=E*(-	A-B---	
(10)	+D)	i:=E*(-i	A-B---C	
(11)	+D)	i:=E*(-E	A-B---C	(@,C,-, T_1)
(12)	+D)	i:=E*(E	A-B-- T_1	
(13)	D)	i:=E*(E+	A-B-- T_1 -	
(14)	)	i:=E*(E+i	A-B-- T_1 -D	
(15)	)	i:=E*(E+E	A-B-- T_1 -D	(+, T_1, D, T_2)
(16)	)	i:=E(E	A-B-- T_2	
(17)		i:=E*(E)	A-B-- T_2 -	
(18)		i:=E+E	A-B- T_2	(*, B, T_2, T_3)
(19)		i:=E	A- T_3	(:=, T_3, -, A)
			(20) A	

产生的四元式:

(@,C,-, T_1)

(+, T_1, D, T_2)

(*, B, T_2, T_3)

(:=, T_3, -, A)

P218 - 5

/*****************

设 A ： 10*20，B、C、D：20，宽度为 w=4 则

T1:= i * 20

T1:=T1+j
T2:=A–84
T3:=4*T1
Tn:=T2[T3]　//这一步是多余的
T4:= i + j
T5:=B–4
T6:=4*T4
T7:=T5[T6]
T8:= i * 20
T8:=T8+j
T9:=A–84
T10:=4*T8
T11:=T9[T10]
T12:= i + j
T13:=D–4
T14:=4*T12
T15:= T13[T14]
T16:=T11+T15
T17:=C–4
T18:=4*T16
T19:=T17[T18]
T20:=T7+T19
Tn:=T20
**********/

P218 - 6

100.（jnz, A, -, 0）
101.（j, -, -, 102）
102.（jnz, B, -, 104）
103.（j, -, -, 0）
104.（jnz, C, -, 103）
105.（j, -, -, 106）
106.（jnz, D, -, 104）　--假链链首
107.（j, -, -, 100）　　--真链链首
假链：{106, 104，103}
真链：{107, 100}

P218 - 7

100.（j<, A, C, 102）
101.（j, -, -, 0）
102.（j<, B, D, 104）
103.（j, -, -, 101）
104.（j=, A, ‘1’, 106）
105.（j, -, -, 109）

106. (+, C, ‘1’, T1)
107. (:=, T1, -, C)
108. (j, -, -, 100)
109. (j≤, A, D, 111)
110. (j, -, -, 100)
111. (+, A, ‘2’, T2)
112. (:=, T2, -, A)
113. (j, -, -, 109)
114. (j, -, - 100)

P219 - 12

/*********************

(1)

MAXINT - 5

MAXINT - 4

MAXINT - 3

MAXINT - 2

MAXINT - 1

MAXINT

(2)翻译模式

方法 1：

for E1 := E2 to E3 do S

$S \rightarrow F$ do MS_1

$F \rightarrow$ For $I := E_1$ to E_2

$I \rightarrow id$

$M \rightarrow \varepsilon$

$S \rightarrow F$ do MS_1 {backpatch(S1.nextlist,nextquad);

backpatch(F.truelist,M.quad);
emit(F.place ‘:=’ F.place ‘+’ 1);
emit(‘j≤,’ F.place ‘,’ F.end ‘,’ M.quad);
S.nextlist := F.falselist;
}

$F \rightarrow$ For $I := E_1$ to E_2 {F.falselist:= makelist(nextquad);

emit(‘j>,’ E1.place ‘,’ E2.place ‘,0’);
emit(I.Place ‘:=’ E1.place);
F.truelist := makelist(nextquad);
emit(‘j,-,-,-’);
F.place := I.place;
F.end := E2.place;
}

$I \rightarrow id$ {p:=lookup(id.name);

if p <> nil then
I.place := p
else error}

$M \to \varepsilon$ {M.quad := nextquad}

**********/

方法 2:

S→ for id:=E1 to E2 do S1

S→ F S1

F→ for id:=E1 to E2 do

$F \to forid := E1toE2$ do

{

INITIAL=NEWTEMP;

emit(‘:=,’ E1.PLACE’, -,’ INITIAL);

FINAL=NEWTEMP;

emit(‘:=,’ E2.PLACE’, -,’ FINAL);

p:= nextquad+2;

emit(‘j ,’ INITIAL ‘,’ FINAL ’ ,’ p);

F.nextlist:=makelist(nextquad);

emit(‘j,—,—,—’);

F.place:=lookup(id.name);

if F.place nil then

emit(F.place ‘:=’ INITIAL)

F.quad:=nextquad;

F.final:=FINAL;

}

$S \to FS1$

{

backpatch(S1.nextlist, nextquad)

p:=nextquad+2;

emit(‘j ,’ F.place ‘,’ F.final ’ ,’ p);

S.nextlist := merge(F.nextlist, makelist(nextquad));

emit(‘j,—,—,—’);

emit(‘succ,’ F.place ’, -,’ F.place);

emit(‘j,—,—,’ F.quad);

}

第九章

P270－9

(1) 传名

即当过程调用时，其作用相当于把被调用段的过程体抄到调用出现处，但必须将其中出现的任一形式参数都代之以相应的实在参数。

A:=2;

B:=3;

A:=A+1;

A:=A+(A+B);

print A;

∴A=9

(2) 传地址

即当程序控制转入被调用段后，被调用段首先把实在参数抄进相应的形式参数的形式单元中，过程体对形参的任何引用或赋值都被处理成对形式单元的间接访问。当被调用段工作完毕返回时，形式单元（都是指示器）所指的实参单元就持有所希望的值。

①A:=2;B:=3;T:=A+B

②把 T,A,A 的地址抄进已知单元 J1,J2,J3

③x:＝J1;y:=J2;z:=J3　　//把实参地址抄进形式单元，且 J2=J3

④Y↑:=y↑+1

  Z↑:=z↑+x↑　　// Y↑：对 y 的间接访问

　　Z↑：对 z 的间接访问

⑤print A

A=8

(3) 得结果

每个形参均对应两个单元，第一个存放实参地址，第二个存放实参值，在过程体中对形参的任何引用或赋值都看成是对它的第二个单元的直接访问，但在过程工作完毕返回前必须把第二个单元的内容放到第一个单元所指的那个实参单元中

①A:=2;B:=3;T:=A+B

②把 T,A,A 的地址抄进已知单元 J1,J2,J3

③x1:=J1;x2:=T;

  y1:=J2;y2:=A;

  z1:=J3;z2:=A;　　//将实参的地址和值分别放进两个形式单元中

④y2:=y2+1; z2:=z2+x2;　　//对形参第二个单元的直接访问

⑤x1↑：＝x2; y1↑:=y2; z1↑:=z2　　//返回前把第二个单元的内容存放到第一个单元所指的实参地址中

⑥print A

A=7

(4) 传值

即被调用段开始工作时，首先把实参的值写进相应的形参单元中，然后就好像使用局部变量一样使用这些形式单元

A:=2;

B:=3;

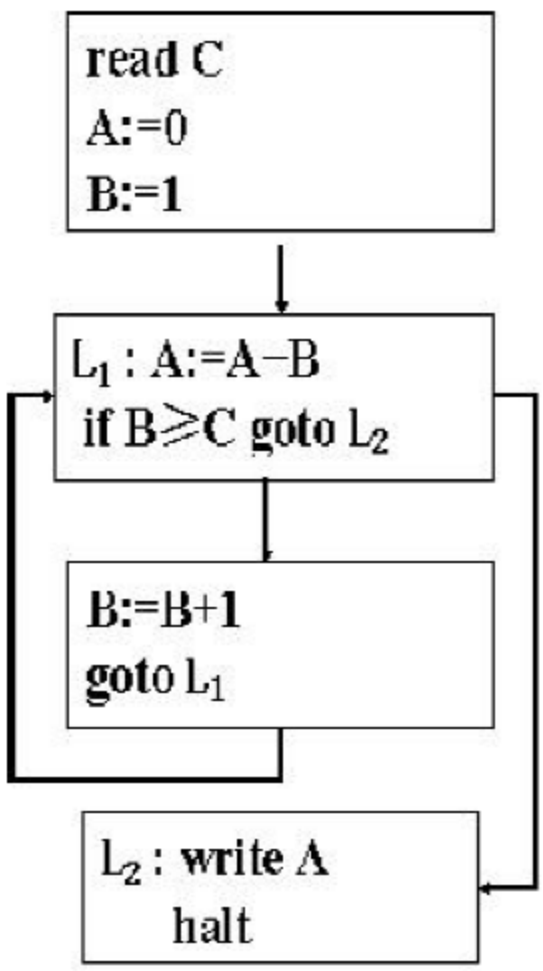
```
x:=A+B
y:=A
z:=A
y:=y+1
z:=z+x
print  A
```

A=2

过程调用不改变 A 的值

第十章

P306-1



P306-2

read A,B
F:=1

C:=A*A B_1

D:=B*B

if C<D goto L_1

E:=A*A
F:=F+1

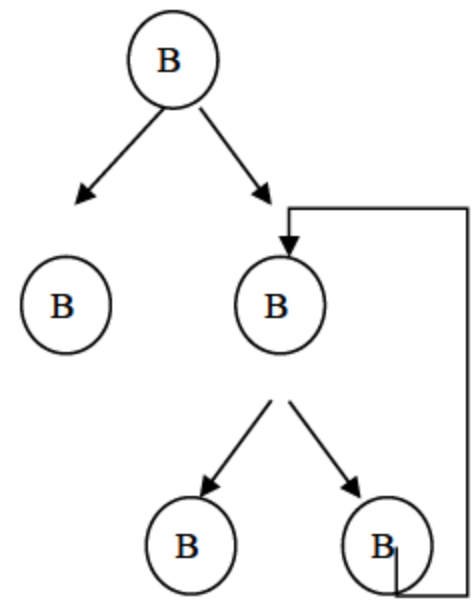
E:=E+F B_2

write E
halt

L_1: E:=B*B

F:=F+2

E:=E+F B_3



write E

if E>100 goto L_2

---

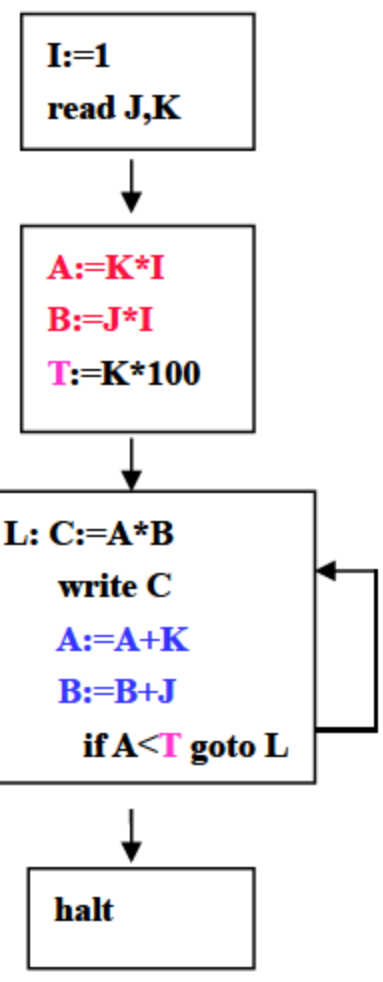
halt B_4

L_2: F:=F-1

goto L_1 B_5

基本块为B_1、B_2、B_3、B_4、B_5

P307-4



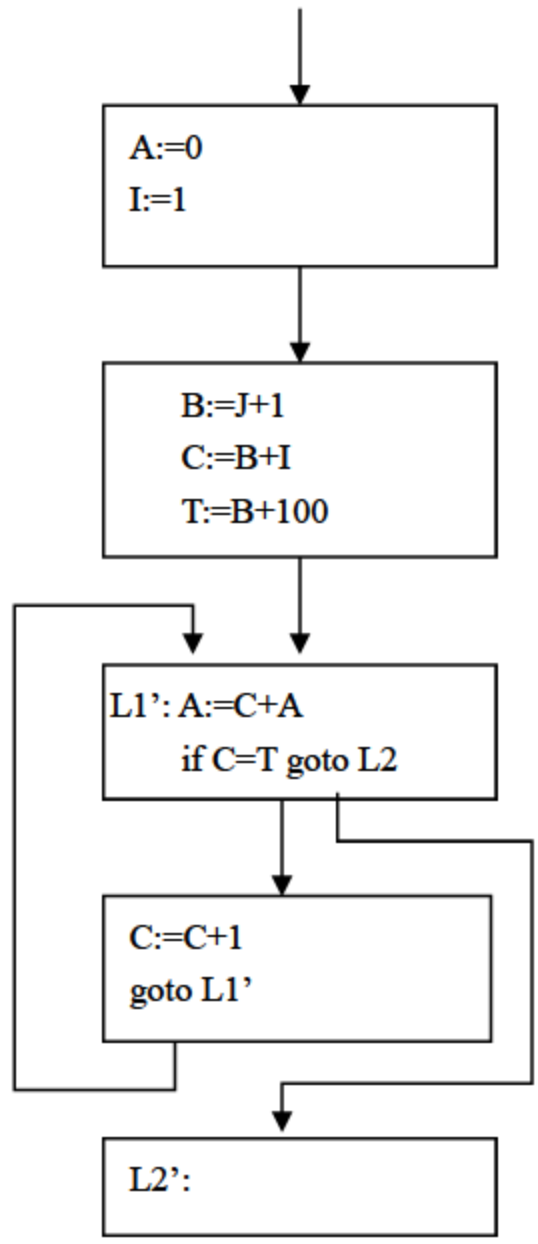
B2 有回路，所以{B2}是循环，B2 既是入口节点，又是出口节点

（1） 代码外提：不存在不变运算，故无代码外提

（2） 强度削弱：A:=K*I B:=J*I *→+

（3） 删除基本归纳变量：I<100 可以用 A<100*K 或 B<100*J 代替

P307-5



{B2,B3}是循环，B2 是入口节点，也是出口节点

（1） 代码外提：B:=J+1

（2） 删除归纳变量