

A1. 给定文法 $G[E]$:

$$\begin{aligned} E &\rightarrow T \mid E + T \\ T &\rightarrow F \mid T * F \\ F &\rightarrow i \mid (E) \end{aligned}$$

试分别指出句型 $E+T*F+i$ 和 $i_1*i_2+i_3$ 的所有短语，直接短语。如果这些句型同时也是右句型，那么还要给出其句柄。 请将结果填入下表中：

句型	短语	直接短语	句柄
$E+T*F+i$			
$i_1*i_2+i_3$			

参考答案：

句型	短语	直接短语	句柄
$E+T*F+i$	$E+T*F+i$ $E+T*F$ $T*F$ i	$T*F$ i	$T*F$
$i_1*i_2+i_3$	$i_1*i_2+i_3$ i_1*i_2 i_1 i_2 i_3	i_1 i_2 i_3	i_1

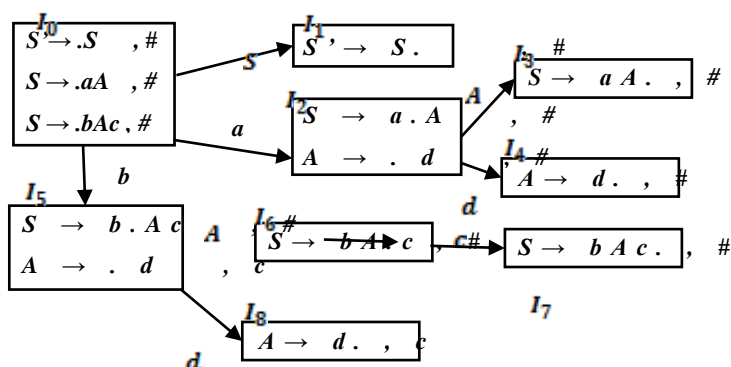
A2. 给定文法 $G[S]$:

$$\begin{aligned} (1) \quad &S \rightarrow a A \\ (2) \quad &S \rightarrow b A c \\ (3) \quad &A \rightarrow d \end{aligned}$$

- (a) 构造该文法的 LR (1) 有限状态机。
- (b) 验证该文法是 LR(1) 文法。
- (c) 该文法是否 LALR(1) 文法？
- (d) 给出对应的 LR(1) 分析表。

参考解答:

(a)



(b) 所有状态都不存在移进-归约和归约-归约冲突. 所以是 LR(1)文法。

(c) 根据 LALR(1)文法定义, 合并 LR(1)状态, 使之与 SLR(1)状态相同。根据(a)中 LR(1)有限机状态, 只需要将 **I₄** 与 **I₈** 与合并, 所得新状态无归约-归约冲突, 所以是 LALR(1)文法。

(d)

状态	ACTION				GOTO		
	a	b	c	d	#	S	A
0	s2	s5				1	
1					acc		
2				s4			3
3					r1		
4					r3		
5				s8			6
6			s7				
7					r2		
8			r3				

.....
A3. 我们用文法 $G[T]$:

$$T \rightarrow T * T \mid T \rightarrow T \mid (T) \mid \text{Int} \mid \text{Bool}$$

来定义类型表达式 T 。它可以是基本类型（这里仅考虑整数类型 **Int** 和布尔类型 **Bool**, 皆为终结符), 乘积类型 ($T * T$), 或者函数类型 ($T \rightarrow T$)。与普通的算术表达式类似, 我们通过括号 $()$ 来指定最高的优先级。此外, 运算符 $*$ 的优先级高于 \rightarrow 。运算符 $*$ 具有左结合性, 而 \rightarrow 具有右结合性。

(1) 画出下列类型表达式的抽象语法树。

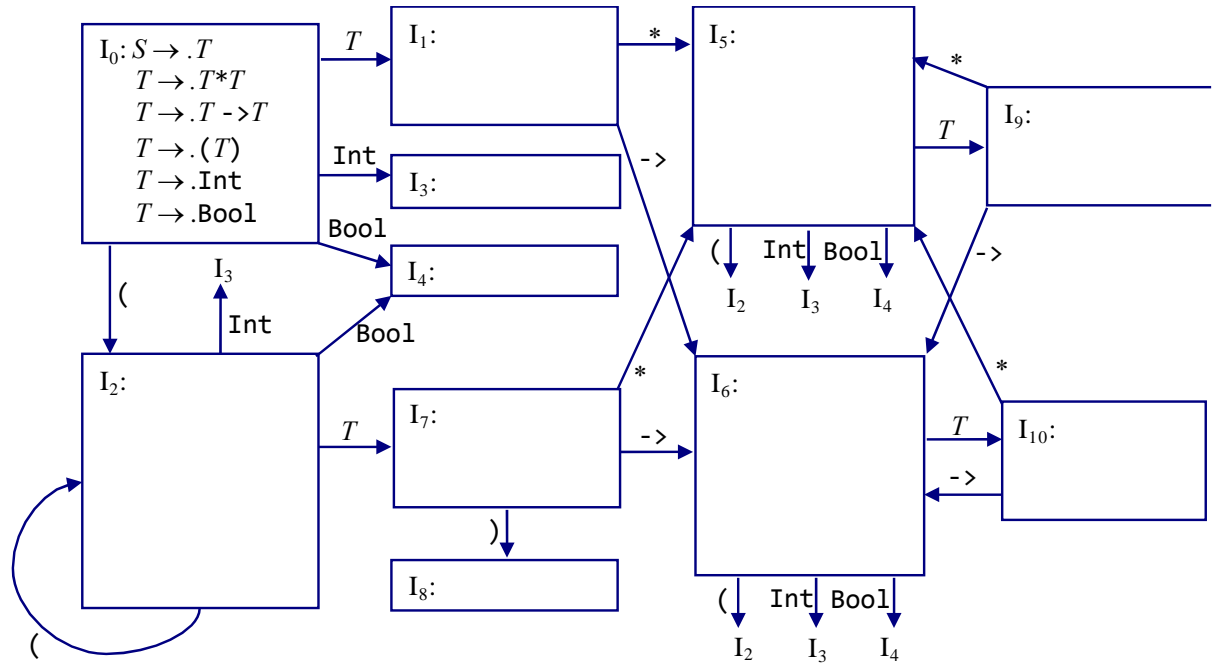
$$\text{Bool} * \text{Int} * \text{Bool} \rightarrow \text{Int} \rightarrow \text{Bool}$$

(2) 文法 $G[T]$ 的增广文法 $G'[S]$ 为

$$(0) S \rightarrow T$$

- (1) $T \rightarrow T * T$
- (2) $T \rightarrow T \rightarrow T$
- (3) $T \rightarrow (T)$
- (4) $T \rightarrow \text{Int}$
- (5) $T \rightarrow \text{Bool}$

请补全它对应的 LR(0) FSM。并说明它为什么不是 SLR(1)文法。



(3) 根据运算符的优先级和结合性，我们可以消除文法 $G[T]$ 的二义性。请补全对应的 LR 分析表。

栈顶 状态	ACTION							GOTO
	*	->	()	Int	Bool	#	
0			s2		s3	s4		1
1	s5	s6						
2			s2		s3	s4		7
3								
4								
5			s2		s3	s4		9
6			s2		s3	s4		10
7	s5	s6		s8				
8								
9								
10								

(4) 根据第 (3) 问构造的 LR 分析表，模拟第 (1) 问中输入串

Bool * Int * Bool -> Int -> Bool

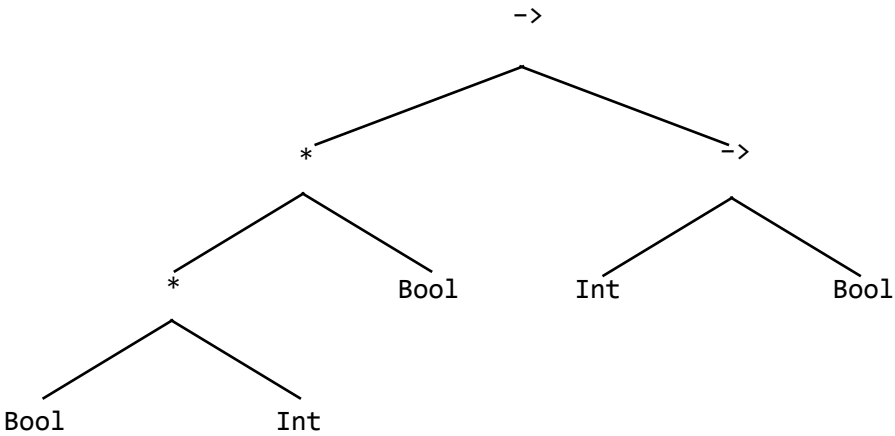
的分析过程，完成下表。（分析栈带有符号，分析动作栏只用写动作编号）

分析栈	余留输入串	分析动作
0#	Bool * Int * Bool -> Int -> Bool #	s4

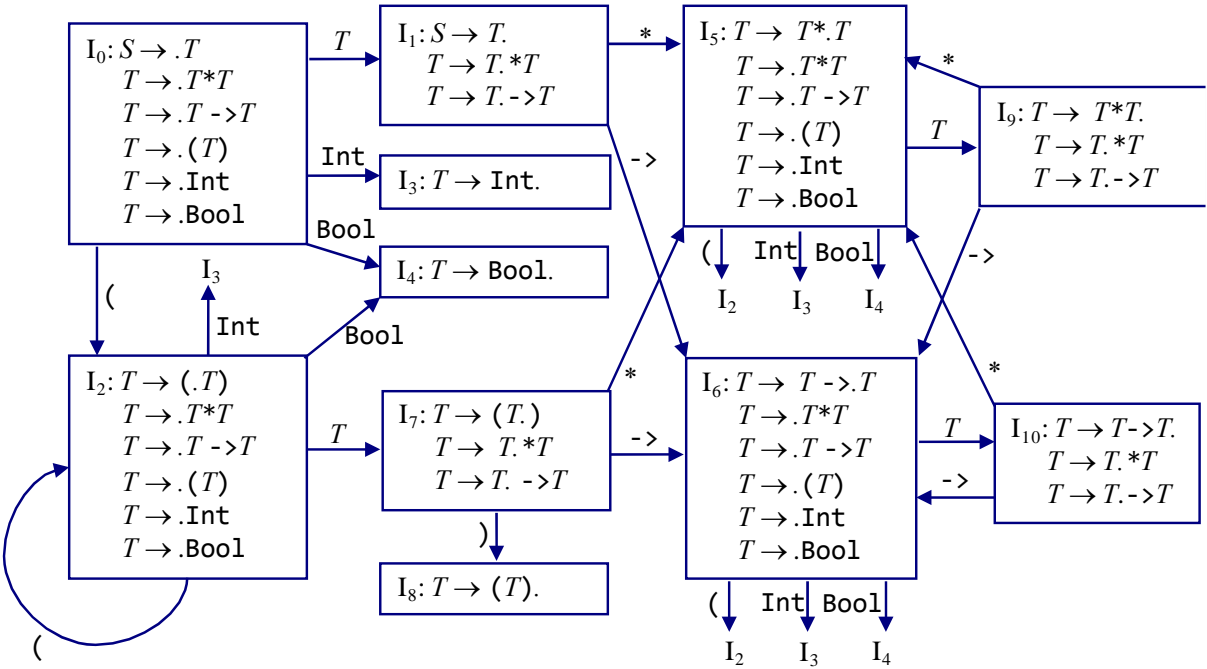
.....

答案：

(1)



(2)



该文法不是 SLR(1)文法，因为*和->均属于 Follow(T)，从而状态 I₉ 和 I₁₀ 均存在移进——归约冲突。

(3) LR 分析表如下

栈顶 状态	ACTION							GOTO
	*	->	()	Int	Bool	#	<i>T</i>
0			s2		s3	s4		1
1	s5	s6					acc	
2			s2		s3	s4		7
3	r4	r4		r4			r4	
4	r5	r5		r5			r5	
5			s2		s3	s4		9
6			s2		s3	s4		10
7	s5	s6		s8				
8	r3	r3		r3			r3	
9	r1	r1		r1			r1	
10	s5	s6		r2			r2	

(4) 分析过程如下

分析栈	余留输入串	动作
0#	Bool * Int * Bool -> Int -> Bool #	s4
0# 4Bool	* Int * Bool -> Int -> Bool #	r5
0# 1 <i>T</i>	* Int * Bool -> Int -> Bool #	s5
0# 1 <i>T</i> 5*	Int * Bool -> Int -> Bool #	s3
0# 1 <i>T</i> 5* 3Int	* Bool -> Int -> Bool #	r4
0# 1 <i>T</i> 5* 9 <i>T</i>	* Bool -> Int -> Bool #	r1
0# 1 <i>T</i>	* Bool -> Int -> Bool #	s5
0# 1 <i>T</i> 5*	Bool -> Int -> Bool #	s4
0# 1 <i>T</i> 5* 4Bool	-> Int -> Bool #	r5
0# 1 <i>T</i> 5* 9 <i>T</i>	-> Int -> Bool #	r1
0# 1 <i>T</i>	-> Int -> Bool #	s6
0# 1 <i>T</i> 6->	Int -> Bool #	s3
0# 1 <i>T</i> 6-> 3Int	-> Bool #	r4
0# 1 <i>T</i> 6-> 10 <i>T</i>	-> Bool #	s6
0# 1 <i>T</i> 6-> 10 <i>T</i> 6->	Bool #	s4
0# 1 <i>T</i> 6-> 10 <i>T</i> 6-> 4Bool	#	r5
0# 1 <i>T</i> 6-> 10 <i>T</i> 6-> 10 <i>T</i>	#	r2
0# 1 <i>T</i> 6-> 10 <i>T</i>	#	r2
0# 1 <i>T</i>	#	acc

A4. 设语言 $L = \{ a^n b^m \mid m > n \geq 0 \}$ ，下列 $G_1[S]$ 和 $G_2[S]$ 是该语言的两个无二义文法（对应两种不同的配对方法）：

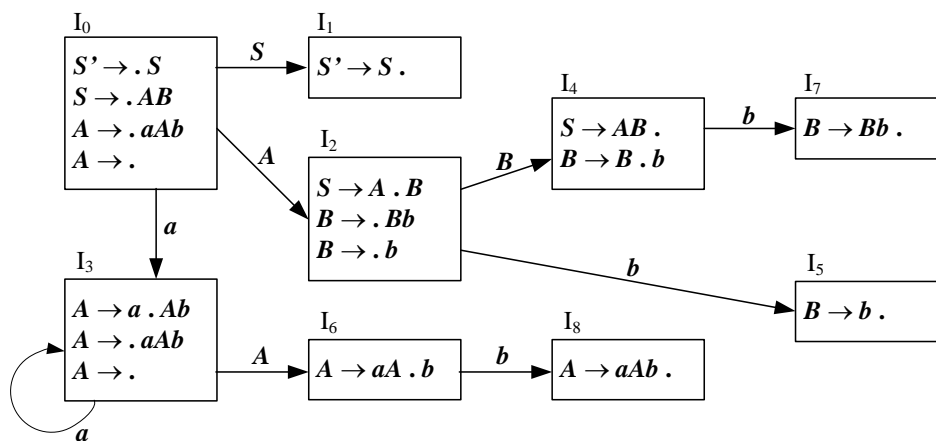
$G_1[S]$:

- (1) $S \rightarrow AB$
- (2) $A \rightarrow aAb$
- (3) $A \rightarrow \varepsilon$
- (4) $B \rightarrow Bb$
- (5) $B \rightarrow b$

$G_2[S]$:

- (1) $S \rightarrow aSb$
- (2) $S \rightarrow B$
- (3) $B \rightarrow Bb$
- (4) $B \rightarrow b$

(1) 下图是相应于 $G_1[S]$ 的 LR(0) 自动机:



文法 $G_1[S]$ 不是 LR(0) 文法。试指出 $G_1[S]$ 的 LR(0) 自动机中存在哪些冲突的状态？并指出这些状态的冲突类别，即是移进-归约冲突还是归约-归约冲突？

(2) 文法 $G_1[S]$ 是 SLR(1) 文法。试解释为什么？

(3) 下图表示 $G_1[S]$ 的 LR(0) 分析表和 SLR(1) 分析表中状态 I_3 和 I_4 两行所对应的内容，上半部分是 LR(0) 分析表，下半部分是 SLR(1) 分析表，但表中的 ACTION 部分没有给出，试补齐之。

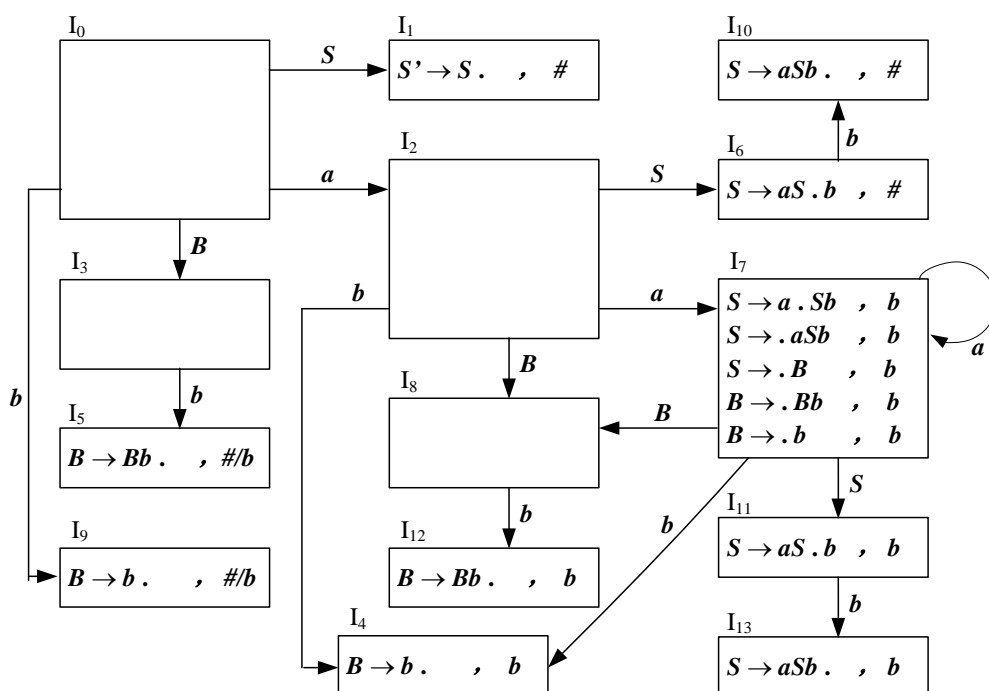
状态		ACTION			GOTO		
		a	b	#	S	A	B
LR(0) 分析表	3					6	
	4						
SLR(1) 分析表	3					6	
	4						

(4) 在针对 $G_1[S]$ 的 SLR(1) 分析过程的某个时刻，符号栈的栈顶是 A ，且栈中包含 a ，则此时所期待的句柄有哪些？而在另一个时刻，符号栈的栈顶是 A ，但栈中不包含 a ，则此刻所期待的句柄有哪些？

(5) 在针对 $G_1[S]$ 的 SLR(1) 分析过程中，若输入消耗掉相同数目的 a 和 b 时发

生错误，你认为合适的报错信息有哪些？

(6) 下图是相应于 $G_2[S]$ 的 LR(1) 自动机，但部分状态所对应的项目集未给出，试补齐之（即分别给出状态 I_0 , I_2 , I_3 和 I_8 对应的项目集。



(7) $G_2[S]$ 是否LR(1)文法？若不是，则指出 $G_2[S]$ 的 LR(1) 自动机中哪些状态有冲突？

参考解答/评分方案：

(1) $G_1[S]$ 的 LR(0) 自动机中存在3个冲突的状态： I_0 , I_3 和 I_4 ；均为移进-归约冲。

(2) $\text{FOLLOW}(S)=\{\#\}$, $\text{FOLLOW}(A)=\{b\}$ 。

因为 $\text{FOLLOW}(A)$ 中不含 a ，所以 I_0 和 I_3 的移进-归约冲突可解决；又因为， $\text{FOLLOW}(S)$ 中不含 b ，所以 I_4 的移进-归约冲突可解决。

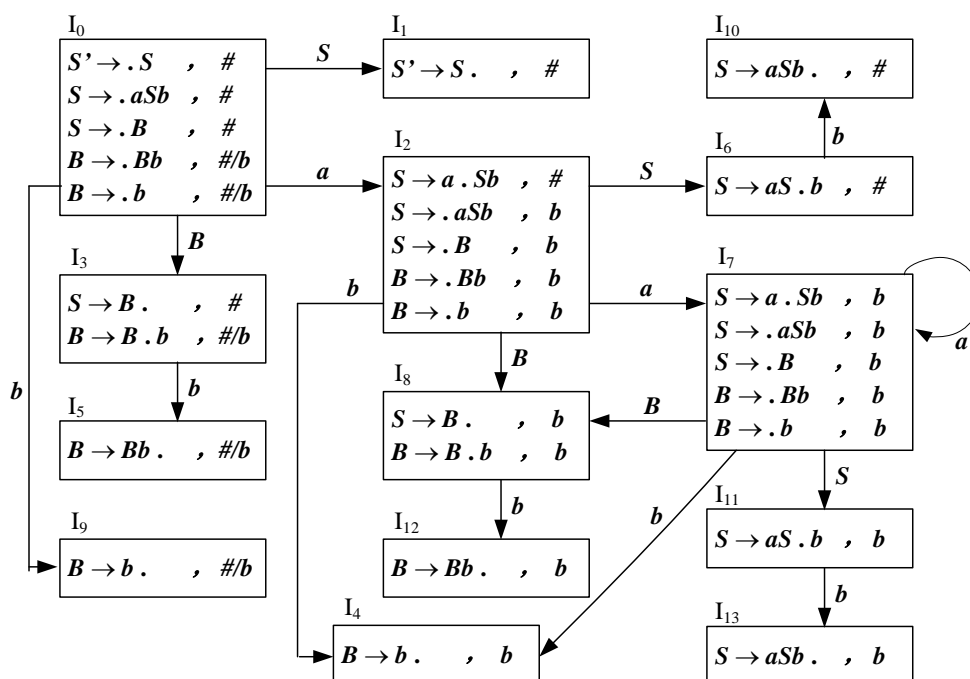
(3) 完整的分析表如下：

状态		ACTION			GOTO		
		<i>a</i>	<i>b</i>	#	<i>S</i>	<i>A</i>	<i>B</i>
LR(0) 分析表	3	s3,r3	r3	r3		6	
	4	r1	s7,r1	r1			
SLR(1) 分析表	3	s3	r3			6	
	4		s7	r1			

(4) 在针对 $G_1[S]$ 的SLR(1)分析过程的某个时刻, 符号栈的栈顶是 A , 且栈中包含 a , 则此时所期待的句柄只有 aAb 。而在另一个时刻, 符号栈的栈顶是 A , 但栈中不包含 a , 则此刻所期待的句柄有3个: Bb 、 b 和 AB 。

(5) 在针对 $G_1[S]$ 的SLR(1)分析过程中, 若输入消耗掉相同数目的 a 和 b 时发生错误, 你认为合适的报错信息有哪些? **缺 a , 缺 b** (批阅过程中再确认一下)

(6)



(7)

不是, 冲突的状态仅一个: I_8 。

.....

以下是 Lecture05 文档中的题目

2. (1)

(a) 构造该文法 $G[S]$ 的 LR(0) 有限状态机。

(b) 说明该文法不是 LR(0) 文法。

(c) 该文法是否 SLR(1) 文法？为什么？

(2) 给定文法 $G[S]$:

$$S \rightarrow SS \mid (S) \mid a$$

完成同 (1) 一样三个问题 (a), (b), (c)。

参考解答:

(2) 首先变换文法为增广文法。增加如下产生式

$$S' \rightarrow S$$

得到增广文法如下

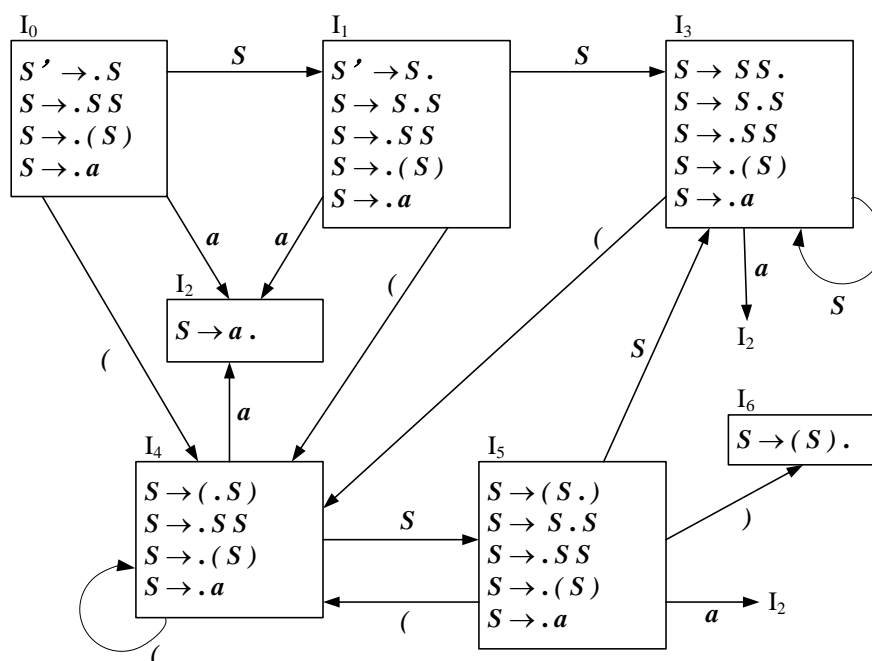
$$S' \rightarrow S$$

$$S \rightarrow SS$$

$$S \rightarrow (S)$$

$$S \rightarrow a$$

(a) 该文法的 LR(0) 有限状态机状态转换图如下:



(b) 状态 I₃ 中包含移进-归约冲突，所以 $G(S)$ 不是 LR(0) 文法。

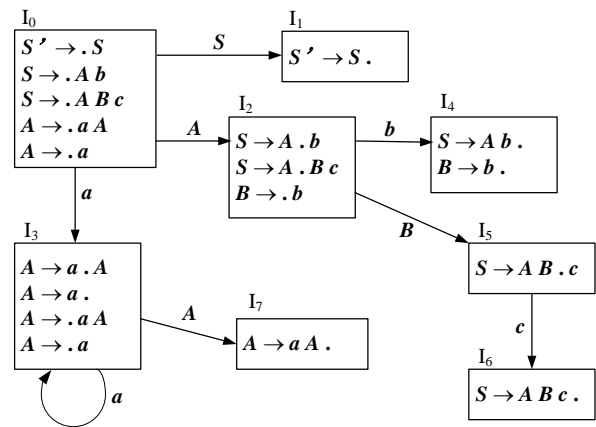
(c) 对于状态 I₃，由于 $FOLLOW(S) = \{a, (,), \#\}$ ，在面临第一个符号是 a 时，不能选

择是归约成S还是移进a，所以，不是SLR(1)文法。

4. 给定 SLR(1) 文法 G[S]:

- (1) $S \rightarrow A b$
- (2) $S \rightarrow A B c$
- (3) $A \rightarrow a A$
- (4) $A \rightarrow a$
- (5) $B \rightarrow b$

其 LR(0) 有限状态机如下图所示：



- (a) 构造该文法的 SLR(1) 分析表。
- (b) 若采用 SLR(1) 方法对于 $L(G)$ 中的某一输入串进行分析，当栈顶出现句柄 a 时，余留输入符号串中的第一个符号是什么？

参考解答:

(a) SLR(1) 分析表:

状态	ACTION				GOTO		
	a	b	c	#	S	A	B
0	s ₃				1	2	
1				acc			
2		s ₄					5
3	s ₃	r ₄				7	
4			r ₅	r ₁			
5			s ₆				
6				r ₂			
7		r ₃					

(b) b

8. (1) 构造下列增广文法 $G[P']$ 的 LR(1) FSM, 验证原文法是 LR(1) 文法:

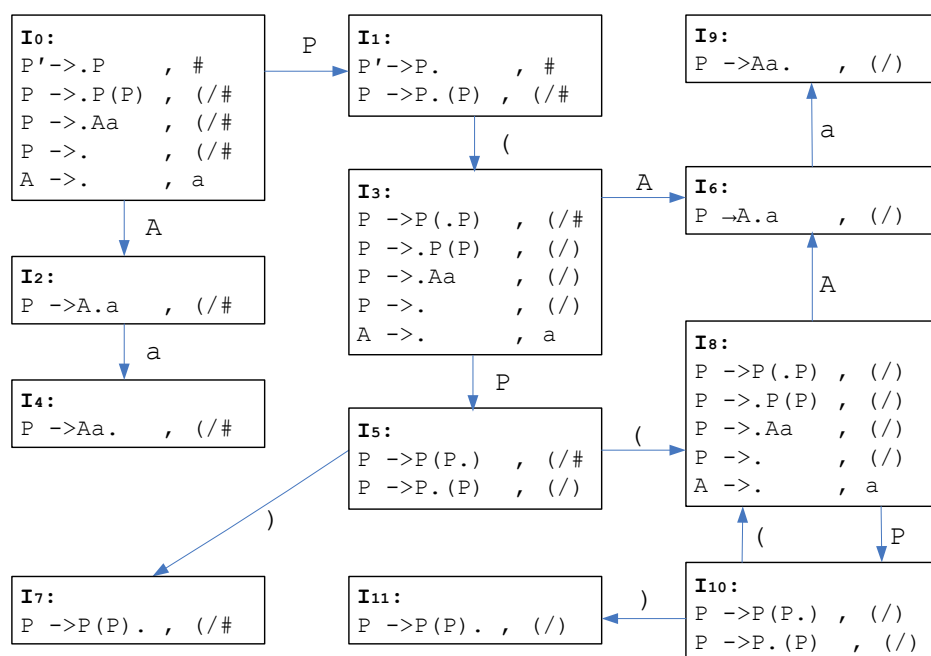
- (0) $P' \rightarrow P$
- (1) $P \rightarrow P(P)$
- (2) $P \rightarrow Aa$
- (3) $P \rightarrow \varepsilon$
- (4) $A \rightarrow \varepsilon$

其中 P' , P , A 为非终结符

(2) 通过合并同芯集(状态)的方法构造相应于上述 LR(1) FSM 的 LALR(1) FSM, 并判断原文法是否 LALR(1) 文法?

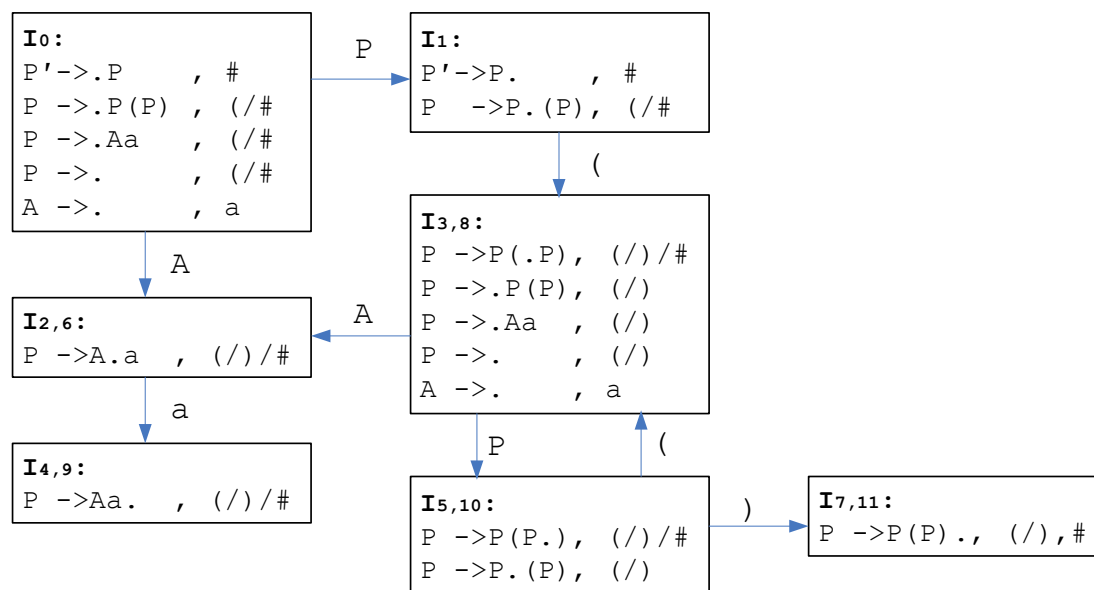
参考解答:

构造增广文法 $G(P')$ 的 LR(1) 项目集族和转换函数如下:



其中的每个状态均无冲突, 所以原文法是 LR(1) 文法。

可以看出: $I2$ 和 $I6$, $I3$ 和 $I8$, $I4$ 和 $I9$, $I5$ 和 $I10$, $I7$ 和 $I11$ 是同芯状态. 通过合并同芯状态的方法构造相应于上述 LR(1) 转换图的 LALR(1) 转换图 如下:



可以看出：所有状态都不存在移进-归约和归约-归约冲突. 所以，原文法是 **LALR(1)** 文法。

10. 给定增广文法 $G[S']$:

- (0) $S' \rightarrow S$
- (1) $S \rightarrow \underline{\text{do}} S \text{ or } S$
- (2) $S \rightarrow \underline{\text{do}} S$
- (3) $S \rightarrow S;S$
- (4) $S \rightarrow \underline{\text{act}}$

其中 S' , S 为非终结符, 其余符号为终结符。

- a) 构造 LR(0)FSM;
- b) 判别原文法是否 LR(0)或 SLR(1) 文法;
- c) 若对一些终结符的优先级或结合性作如下规定:

or 优先性大与 do;
 ; 优先性大与 do
 ; 优先性大与 or
 ; 服从左结合性

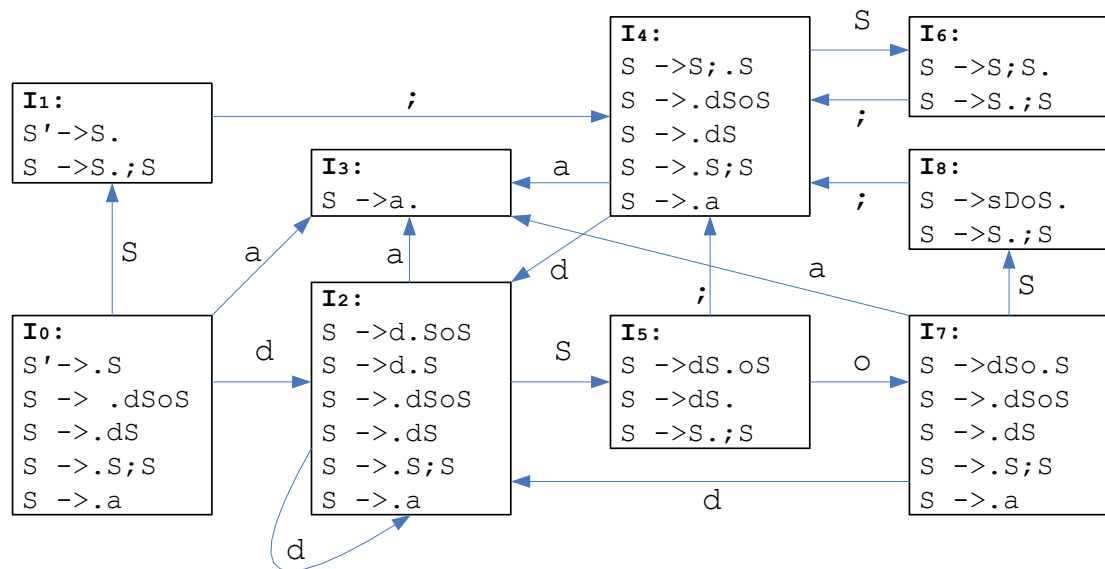
试构造原文法的一个 LR 分析表。

参考解答:

由产生式知:

$\text{Follow}(S) = \{\text{or}, ;, \#\}$

a) 原文法的 LR(0)FSM (其中, d、o 和 a 分别代表 do、or 和 act, 下同):



b) 原文法不是 LR(0) 的也不是 SLR(1) 的, 因为:

在 I5、I6、和 I8 中存在移进-归约冲突, 因此所给文法不是 LR(0) 文法。

又由于 $\text{Follow}(S) = \{o, ;, \#\}$

在 I6、和 I8 中:

$\text{Follow}(S) \cap \{ ; \} = \{o, ;, \#\} \cap \{ ; \} = \{ ; \} \neq \emptyset$

在 I5 中:

$\text{Follow}(S) \cap \{ ;, o \} = \{o, ;, \#\} \cap \{ ;, o \} = \{ ;, o \} \neq \emptyset$

所以该文法也不是 SLR(1) 文法。

实际上很容易证明所给文法是二义性的, 因此该文法不可能是 LR 文法,

c) 若对一些终结符的优先级以及算符的结合规则规定如下:

or 优先性大与 do

; 优先性大与 do

; 优先性大与 or

; 服从左结合性

则在 I5 中: “o 和 ; 优先性都大于 d”, 所以遇输入符号 o 和 ; 时移进; 遇 # 号归约。

在 I6 中: “; 服从左结合” 所以遇输入符属于 $\text{Follow}(S)$ 的都应归约。

在 I8 中: “; 优先性大于 o” 所以遇输入符为 ; 号移进; 遇 o 和 # 归约。

此外, 在 I1 中: 接受和移进可以不看成冲突, 因为接受只有遇 #。

由以上分析, 所有存在的移进-归约冲突可用规定的终结符优先级以及算符的结合规则解决, 所构造的 LR 分析表如下表所示:

STATE	ACTION					GOTO
	<u>do</u>	<u>or</u>	;	<u>act</u>	#	S
0	<i>S2</i>			<i>S3</i>		1
1			<i>S4</i>		<i>accept</i>	
2	<i>S2</i>			<i>S3</i>		5
3		<i>r4</i>	<i>r4</i>		<i>r4</i>	
4	<i>S2</i>			<i>S3</i>		6
5		<i>S7</i>	<i>S4</i>		<i>r2</i>	
6		<i>r3</i>	<i>r3</i>		<i>r3</i>	
7	<i>S2</i>			<i>S3</i>		8
8		<i>r1</i>	<i>S4</i>		<i>r1</i>	

12. 已知某文法 $G[S]$ 的 LALR(1)分析表如下：

状态	ACTION					GOTO
	<i>a</i>	<i>t</i>	<i>g</i>	<i>c</i>	#	<i>S</i>
0	s11	s8		s4		1
1				s2	acc	
2			s3			
3	s11	s8		s4		16
4	s5					
5	s6					
6				s7		
7			r1	r1	R1	
8			s9			
9				s10		
10	s11	s8		s4		14
11	s11	s8		s4		12
12			s13	s2		
13	s11	s8		s4		15
14			r4	s2	R4	
15			r2	s2	R2	
16			r3	s2	R3	

并且已知各规则右边语法符号的个数以及左边的非终结符如下：

规则编号	1	2	3	4
右部长	4	4	4	4
左部符号	S	S	S	S

(a) 写出使用上述 LALR(1)分析器分析下面串的过程（只需写出前 10 步，列出所有可能的 ri ， sj 序列，注意先后次序）：

acaaccgtgccaacgatgccaa ...

(b) 试指出该串相对于上述文法的句柄。

参考解答:

(a) s11, s4, s5, s6, s7, r1, s2, s3, s8, s9, ...

(b) 原始文法（不在题目中公开）是

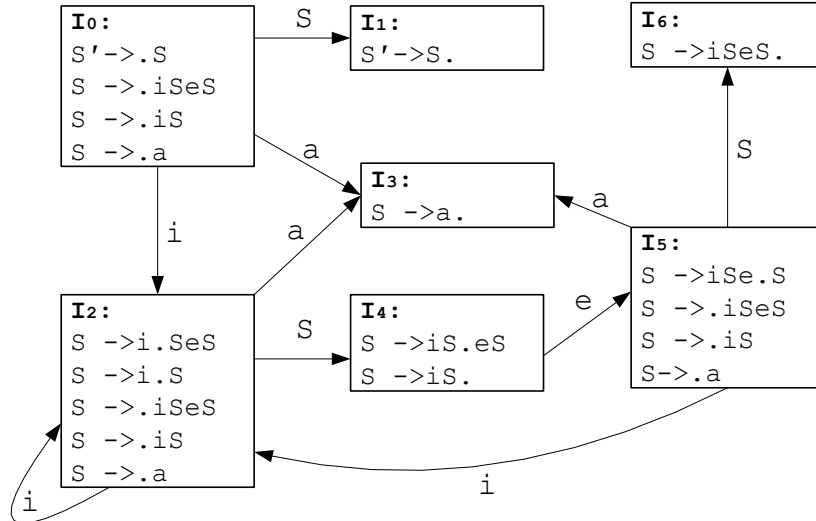
$S \rightarrow caac \mid aSgS \mid ScgS \mid tgcS$

该串相对于上述文法的句柄是：caac

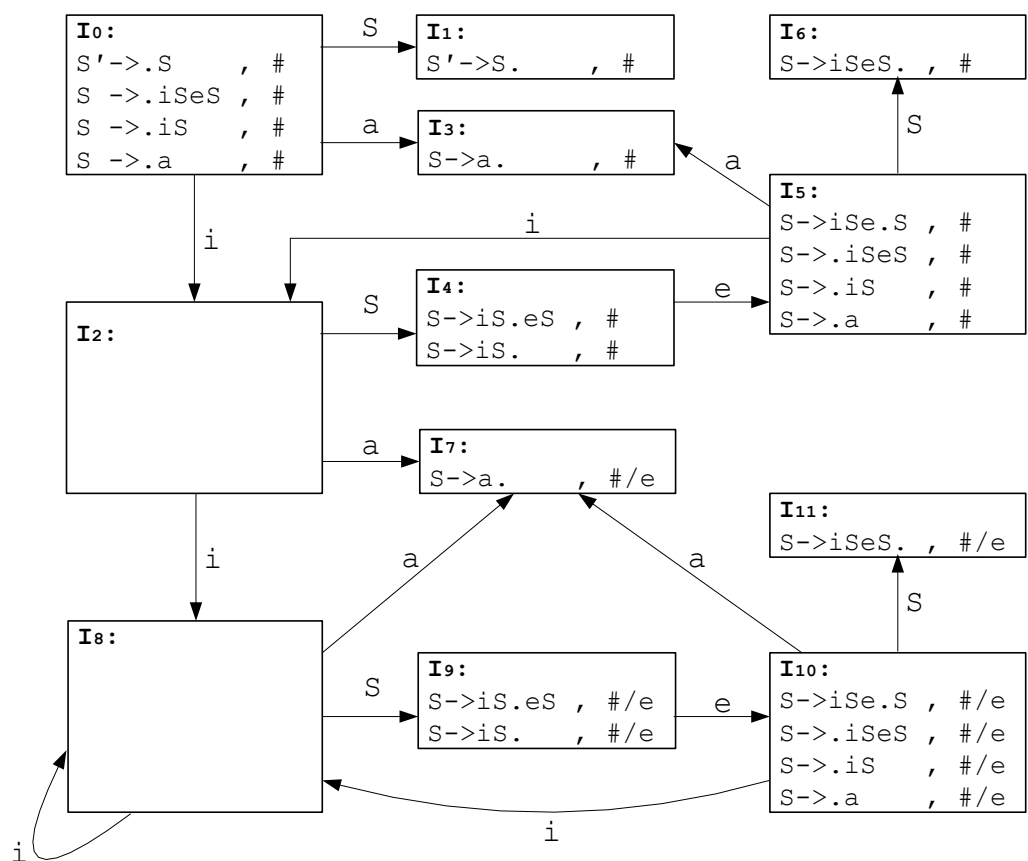
13. 给定如下文法 $G[S]$:

- (1) $S \rightarrow \underline{\text{if}} S \underline{\text{else}} S$
- (2) $S \rightarrow \underline{\text{if}} S$
- (3) $S \rightarrow a$

为文法 $G[S]$ 增加产生式 $S' \rightarrow S$, 得到增广文法 $G'[S']$, 下图是相应的LR(0)自动机 (i 表示 if, e 表示 else):



- (1) 指出LR(0)自动机中的全部冲突状态及其冲突类型，以说明文法 $G[S]$ 不是LR(0)文法。
- (2) 文法 $G[S]$ 也不是SLR(1)文法。为什么？
- (3) 下图表示文法 $G[S]$ 的LR(1)自动机，部分状态所对应的的项目集未给出，试补齐之（即分别给出状态 I_2 , I_8 , 和 I_{10} 对应的项目集。



- (4) 指出LR(1)自动机中的全部冲突状态，这说明文法 $G[S]$ 也不是 LR(1) 文法。
- (5) 若规定最近匹配原则，即 `else` 优先匹配左边靠近它的未匹配的if，则可以解决上述2个自动机中的状态冲突。下图表示文法 $G[S]$ 在规规定这一规则情况下的SLR(1)分析表，状态 4~6 对应的行未给出，试补齐之。

状态	ACTION				GOTO
	i	e	a	#	S
0	s2		s3		1
1				acc	
2	s2		s3		4
3		r3		r3	
4					
5					
6					

下图表示文法 $G[S]$ 在规规定这一规则情况下的LR(1)分析表，状态 4, 7 和 9 对

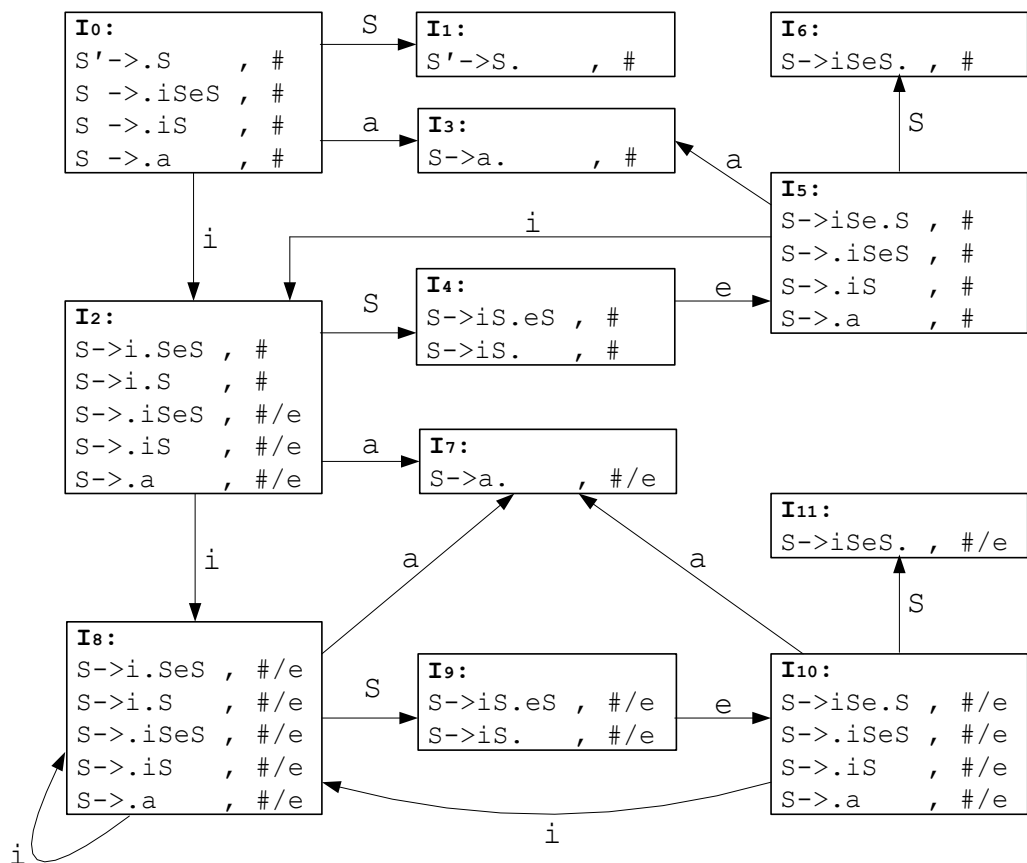
应的行未给出，试补齐之。

状态	ACTION				GOTO
	i	e	a	#	S
0	s2		s3		1
1				acc	
2	s8		s7		4
3				r3	
4					
5	s2		s3		6
6				r1	
7					
8	s8		s7		9
9					
10	s8		s7		11
11		r1		r1	

- (6) 对于文法G[S]中正确的句子，基于上述两个分析表均可以成功进行LR分析。然而，对于不属于文法G[S]中的句子，两种分析过程发现错误的速度不同，即发现错误时所经过的移进/归约总步数有差异。试给出一个长度不超过10的句子（即所包含的终结符个数不超过10），使得两种分析过程发现错误的速度不同。哪一个更快？对于你给的例子，两种分析过程分别到达哪个状态会发现错误？

参考解答:

- (1) 状态 I_4 有冲突，为移进-归约冲突。
- (2) 因 $\text{Follow}(S) = \{ e, \# \}$ ， $e \in \text{Follow}(S)$ ，所以状态 I_4 的移进-归约冲突不可解决，所以该文法不是 SLR(1) 文法。
- (3) 完整的 LR(1) 自动机如下：



(4) 状态 I9 有冲突，同样为移进-归约冲突。

(5) 完整的 SLR(1)分析表

状态	ACTION				GOTO
	i	e	a	#	S
0	s2		s3		1
1				acc	
2	s2		s3		4
3		r3		r3	
4		s5		r2	
5	s2		s3		6
6		r1		r1	

完整的 LR(1)分析表

状态	ACTION				GOTO
	i	e	a	#	S
0	s2		s3		1
1				acc	
2	s8		s7		4
3				r3	
4		s5		r2	
5	s2		s3		6
6				r1	
7		r3		r3	
8	s8		s7		9
9		s10		r2	
10	s8		s7		11
11		r1		r1	

(6) 如对于句子 a else，用LR(1)分析1步后到达状态3发现错误，用SLR(1)分析2步后到达状态1发现错误，所以LR(1) 分析更快。

另解：如对于句子 if a else a else，用LR(1)分析5步后到达状态3发现错误，用SLR(1)分析8步后到达状态1发现错误，所以LR(1) 分析更快。