

编译原理作业 (4)

姓名: 陈彦泽 学号: 181250015

评分: 10 评阅: 王

2020 年 12 月 03 日

请独立完成作业, 不得抄袭。
若得到他人帮助, 请致谢。
若参考了其它资料, 请给出引用。
鼓励讨论, 但需独立书写解题过程。

1 作业 (必做部分)

题目 1 ([10 = 1 + 2 + 2 + 3 + 2 分])

给定下述文法 G ,

$$L \rightarrow LP \quad (1)$$

$$L \rightarrow P \quad (2)$$

$$P \rightarrow (P) \quad (3)$$

$$P \rightarrow () \quad (4)$$

(1) 简述 G 所对应的语言;

(2) 为 G 构造 $LR(0)$ 自动机;

注意: 为了尽量统一状态编号, 便于批改, 当计算 CLOSURE 时, 请按照文法编号大小顺序加入新项。当计算 $GOTO(I, X)$ 时, 请按照 I 中项的出现顺序依次考虑可能的转移符号 X 。

要求: 给出初始状态 I_0 的计算方法以及 $GOTO(I_0, ())$ 的计算方法。

(3) 为该文法设计 $LR(0)$ 分析表; 该文法是 $LR(0)$ 文法吗? 请说明理由。

(4) 为该文法设计 $SLR(1)$ 分析表; 该文法是 $SLR(1)$ 文法吗? 请说明理由。

要求: 请说明归约的设置条件。

(5) 如果该文法是 $SLR(1)$ 文法, 请给出识别输入串 $((()))()$ 时自动机所经历的状态 (编号)。

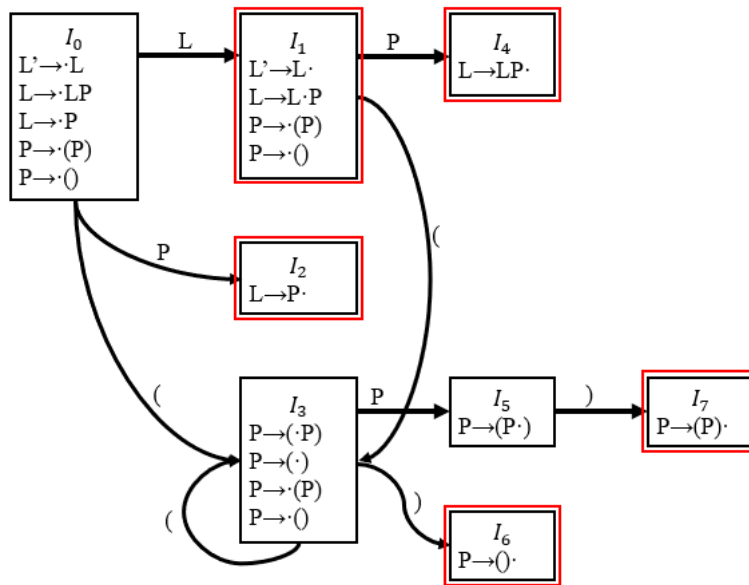
解答:

(1) 一组或多组成对括号，一组成对括号的形式为 $()$, $((()))$, $(((((...))))))$ ，但不包括一对括号内包含多个成对括号的情况，如 $((()))$

(2) $I_0 = \text{CLOSURE}(\{L' \rightarrow \cdot L\})$
 $= \text{CLOSURE}(\{L' \rightarrow \cdot L, L \rightarrow \cdot LP, L \rightarrow \cdot P\})$
 $= \{L' \rightarrow \cdot L, L \rightarrow \cdot LP, L \rightarrow \cdot P, P \rightarrow \cdot (P), P \rightarrow \cdot ()\}$

$\text{GOTO}(I_0, () = \text{CLOSURE}(\{P \rightarrow (\cdot P), P \rightarrow (\cdot)\})$
 $= \{P \rightarrow (\cdot P), P \rightarrow (\cdot), P \rightarrow \cdot (P), P \rightarrow \cdot ()\}$

如图所示:



(3) 该文法是 $LR(0)$ 文法，因为 $LR(0)$ 分析表没有冲突

状态	ACTION			GOTO	
	()	\$	L	P
0	s_3			g_1	g_2
1	s_3		acc		g_4
2	r_2	r_2	r_2		
3	s_3	s_6			g_5
4	r_1	r_1	r_1		
5		s_7			
6	r_4	r_4	r_4		
7	r_3	r_3	r_3		

- (4) $FOLLOW(L) = \{ (, \$ \}$
 $FOLLOW(P) = \{ (,), \$ \}$

状态	ACTION			GOTO	
	()	\$	L	P
0	s_3			g_1	g_2
1	s_3		acc		g_4
2	r_2		r_2		
3	s_3	s_6			g_5
4	r_1		r_1		
5		s_7			
6	r_4	r_4	r_4		
7	r_3	r_3	r_3		

- (5) 0 3 3 6 5 7 2 1 3 6 4 1

I_0 : next-token (s_3 to I_3 stack:(s_3
 I_3 : next-token (s_3 to I_3 stack:(s_3 (s_3
 I_3 : next-token) s_6 to I_6 stack:(s_3 (s_3) s_6
 I_6 : next-token) r_4 、 g_5 to I_5 stack:(s_3 (s_3) P_5
 I_5 : next-token) s_7 to I_7 stack:(s_3 (s_3) P_5) s_7
 I_7 : next-token (r_3 、 g_2 to I_2 stack: P_2
 I_2 : next-token (r_2 、 g_1 to I_1 stack: L_1
 I_1 : next-token (s_3 to I_3 stack: L_1 (s_3
 I_3 : next-token) s_6 to I_6 stack: L_1 (s_3) s_6
 I_6 : next-token \$ r_4 、 g_4 to I_4 stack: L_1 P_4
 I_4 : next-token \$ r_1 、 g_1 to I_1 stack: L_1
 I_1 : next-token \$ acc
