

# 编译原理

## 作业 4: 编译原理 (第 5, 6 章)

中国人民大学 信息学院 崔冠宇 2018202147

### 一、书中练习题

P135, 7. 证明下面文法是 SLR(1) 但不是 LR(0) 的。

- $S \rightarrow A$
- $A \rightarrow Ab \mid bBa$
- $B \rightarrow aAc \mid a \mid aAb$

解: 尝试构造 LR(0) 分析表:

1. 构造 LR(0) 项目:

- (0)  $S' \rightarrow \cdot S$     (1)  $S' \rightarrow S \cdot$     (2)  $S \rightarrow \cdot A$     (3)  $S \rightarrow A \cdot$     (4)  $A \rightarrow \cdot Ab$
- (5)  $A \rightarrow A \cdot b$     (6)  $A \rightarrow Ab \cdot$     (7)  $A \rightarrow \cdot bBa$     (8)  $A \rightarrow b \cdot Ba$     (9)  $A \rightarrow bB \cdot a$
- (10)  $A \rightarrow bBa \cdot$     (11)  $B \rightarrow \cdot aAc$     (12)  $B \rightarrow a \cdot Ac$     (13)  $B \rightarrow aA \cdot c$     (14)  $B \rightarrow aAc \cdot$
- (15)  $B \rightarrow \cdot a$     (16)  $B \rightarrow a \cdot$     (17)  $B \rightarrow \cdot aAb$     (18)  $B \rightarrow a \cdot Ab$     (19)  $B \rightarrow aA \cdot b$
- (20)  $B \rightarrow aAb \cdot$

2. 懒惰求值方式, 按广度优先构造 LR(0) 项目集和计算转换函数  $GO(I, X)$ :

- $I_0 = \text{CLOSURE}(\{(0)\})$ :

- (0)  $S' \rightarrow \cdot S$
- (2)  $S \rightarrow \cdot A$
- (4)  $A \rightarrow \cdot Ab$

– (7)  $A \rightarrow \cdot bBa$

$GO(I_0, S) = CLOSURE(\{(1)\}) = I_1$

$GO(I_0, A) = CLOSURE(\{(3), (5)\}) = I_2$

$GO(I_0, b) = CLOSURE(\{(8)\}) = I_3$

•  $I_1 = CLOSURE(\{(1)\})$ :

– (1)  $S' \rightarrow S \cdot$

•  $I_2 = CLOSURE(\{(3), (5)\})$ :

– (3)  $S \rightarrow A \cdot$

– (5)  $A \rightarrow A \cdot b$

(此节点有移进-归约冲突，因此该文法不是 LR(o) 的)

•  $I_3 = CLOSURE(\{(8)\})$ :

– (8)  $A \rightarrow b \cdot Ba$

– (11)  $B \rightarrow \cdot aAc$

– (15)  $B \rightarrow \cdot a$

– (17)  $B \rightarrow \cdot aAb$

$GO(I_2, b) = CLOSURE(\{(6)\}) = I_4$

$GO(I_3, B) = CLOSURE(\{(9)\}) = I_5$

$GO(I_3, a) = CLOSURE(\{(12), (16), (18)\}) = I_6$

•  $I_4 = CLOSURE(\{(6)\})$ :

– (6)  $A \rightarrow Ab \cdot$

•  $I_5 = CLOSURE(\{(9)\})$ :

– (9)  $A \rightarrow bB \cdot a$

•  $I_6 = CLOSURE(\{(12), (16), (18)\})$ :

- (12)  $B \rightarrow a \cdot Ac$
- (16)  $B \rightarrow a \cdot$
- (18)  $B \rightarrow a \cdot Ab$
- (4)  $A \rightarrow \cdot Ab$
- (7)  $A \rightarrow \cdot bBa$

(此节点有移进-归约冲突，因此该文法不是 LR(o) 的)

$$GO(I_5, a) = \text{CLOSURE}(\{(10)\}) = I_7$$

$$GO(I_6, A) = \text{CLOSURE}(\{(13), (19), (5)\}) = I_8$$

$$GO(I_6, b) = \text{CLOSURE}(\{(8)\}) = I_3$$

$$\bullet I_7 = \text{CLOSURE}(\{(10)\}):$$

- (10)  $A \rightarrow bBa \cdot$

$$\bullet I_8 = \text{CLOSURE}(\{(13), (19), (5)\}):$$

- (13)  $B \rightarrow aA \cdot c$
- (19)  $B \rightarrow aA \cdot b$
- (5)  $A \rightarrow A \cdot b$

$$GO(I_8, c) = \text{CLOSURE}(\{(14)\}) = I_9$$

$$GO(I_8, b) = \text{CLOSURE}(\{(20), (6)\}) = I_{10}$$

$$\bullet I_9 = \text{CLOSURE}(\{(14)\}):$$

- (14)  $B \rightarrow aAc \cdot$

$$\bullet I_{10} = \text{CLOSURE}(\{(20), (6)\}):$$

- (20)  $B \rightarrow aAb \cdot$
- (6)  $A \rightarrow Ab \cdot$

可见项目集  $I_2, I_6$  有移进-归约冲突，因此该文法不是  $LR(o)$  的。对这两个节点考虑它们是否满足  $SLR(i)$ ：

1. 考虑项目集  $I_2$ ：

- 移进项目的字符集合为  $\{b\}$
- 各归约项目的 FOLLOW 集合分别是  $FOLLOW(S) = \{\#\}$

可见它们没有冲突，可以用  $SLR(i)$  解决冲突。

2. 考虑项目集  $I_6$ ：

- 移进项目的字符集合为  $\{A, b\}$
- 各归约项目的 FOLLOW 集合分别是  $FOLLOW(B) = \{a\}$

可见它们没有冲突，可以用  $SLR(i)$  解决冲突。

因此该文法是  $SLR(i)$  的，但不是  $LR(o)$  的。

**P135, 8.** 证明下面的文法

- $S \rightarrow AaAb \mid BbBa$
- $A \rightarrow \varepsilon$
- $B \rightarrow \varepsilon$

是  $LL(i)$  的但不是  $SLR(i)$  的。

**解：**要证明这个文法是  $LL(i)$  的，只需要考虑每个表达式，尝试构造  $LL(i)$  分析表：

1.  $S \rightarrow AaAb$ ,  $FIRST(AaAb) = \{a\}$ , 于是  $M[S, a] = S \rightarrow AaAb$ ;
2.  $S \rightarrow BbBa$ ,  $FIRST(BbBa) = \{b\}$ , 于是  $M[S, b] = S \rightarrow BbBa$ ;

3.  $A \rightarrow \varepsilon$ ,  $\text{FIRST}(\varepsilon)=\{\varepsilon\}$ ,  $\text{FOLLOW}(A)=\{a, b\}$ , 于是  $M[A, a] = A \rightarrow \varepsilon$ ,  $M[A, b] = A \rightarrow \varepsilon$ ;

4.  $B \rightarrow \varepsilon$ ,  $\text{FIRST}(\varepsilon)=\{\varepsilon\}$ ,  $\text{FOLLOW}(A)=\{a, b\}$ , 于是  $M[B, a] = B \rightarrow \varepsilon$ ,  $M[B, b] = B \rightarrow \varepsilon$ ;

没有冲突, 因此该文法是 LL(1) 的。

为了证明该文法不是 SLR(1) 的, 尝试构造 LR 分析表。

1. 先构造 LR(o) 项目:

(o)  $S' \rightarrow \cdot S$       (1)  $S' \rightarrow S \cdot$       (2)  $S \rightarrow \cdot AaAb$       (3)  $S \rightarrow A \cdot aAb$       (4)  $S \rightarrow Aa \cdot Ab$   
 (5)  $S \rightarrow AaA \cdot b$       (6)  $S \rightarrow AaAb \cdot$       (7)  $S \rightarrow \cdot BbBa$       (8)  $S \rightarrow B \cdot bBa$       (9)  $S \rightarrow Bb \cdot Ba$   
 (10)  $S \rightarrow BbB \cdot a$       (11)  $S \rightarrow BbBa \cdot$       (12)  $A \rightarrow \cdot$       (13)  $B \rightarrow \cdot$

2. 懒惰求值方式, 按广度优先构造 LR(o) 项目集和计算转换函数  $GO(I, X)$ :

$I_0 = \text{CLOSURE}(\{(o)\})$ :

- (o)  $S' \rightarrow \cdot S$
- (2)  $S \rightarrow \cdot AaAb$
- (7)  $S \rightarrow \cdot BbBa$
- (12)  $A \rightarrow \cdot$
- (13)  $B \rightarrow \cdot$

(此节点有移进-归约冲突, 因此该文法不是 LR(o) 的)

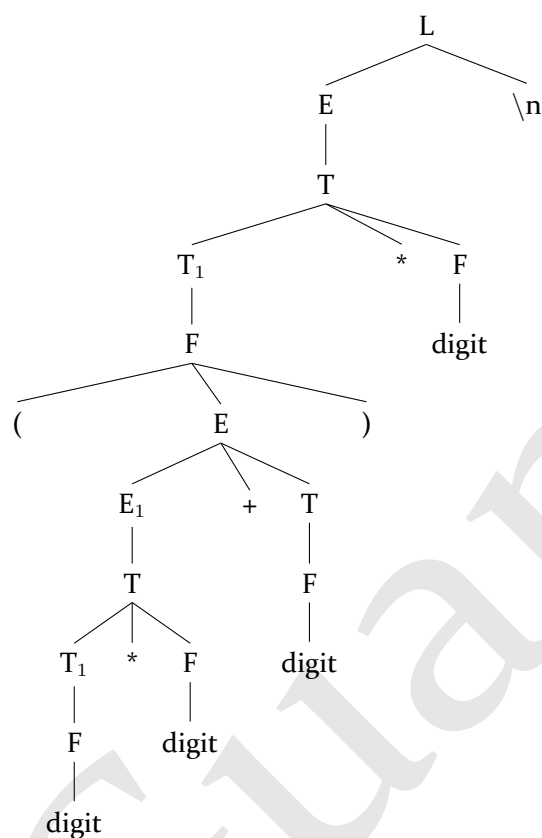
3. 看看该节点能否用 SLR(1) 处理冲突:

- 移进项目的字符集合为  $\{S, A, B\}$ ;
- 各归约项目的 FOLLOW 集合分别是  $\text{FOLLOW}(A) = \{a, b\}$ ,  $\text{FOLLOW}(B) = \{a, b\}$ , 冲突无法解决。

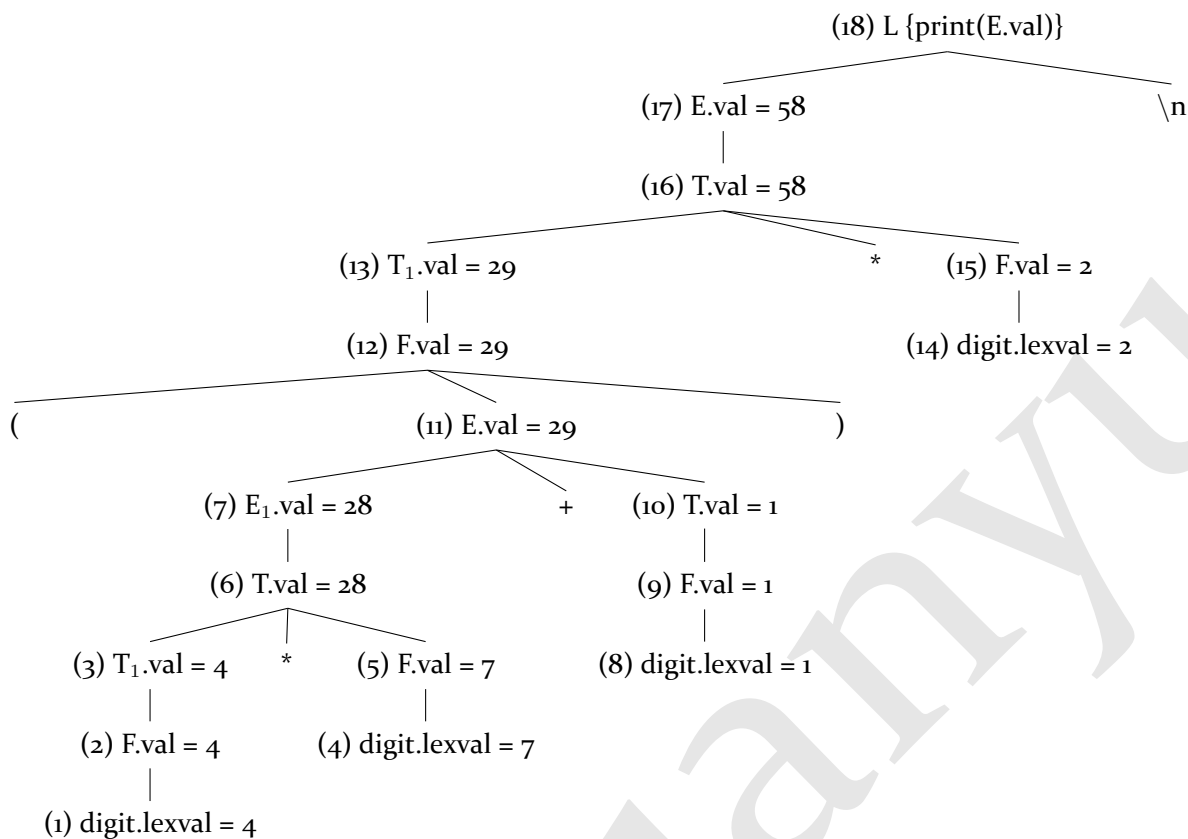
由上面的讨论, 该文法是 LL(1) 的, 但不是 SLR(1) 的。

**P164, 1.** 按照表 6.1 所示的属性文法，构造表达式  $(4 * 7 + 1) * 2$  的附注语法树。

**解:** 先给出语法树，如下：



然后自下而上计算综合属性，得到附注语法树：



**P164, 2.** 对表达式  $((a) + (b))$ :

- (1) 按照表 6.4 所示的属性文法构造该表达式的抽象语法树;
- (2) 按照图 6.17 所示的翻译模式, 构造该表达式的抽象语法树。

**解:**

- (1) 先给出文法对应的语法树, 如下所示:

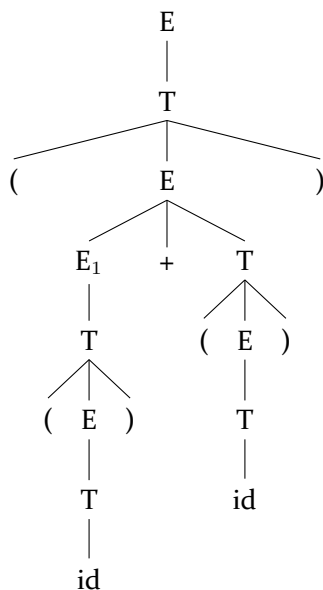
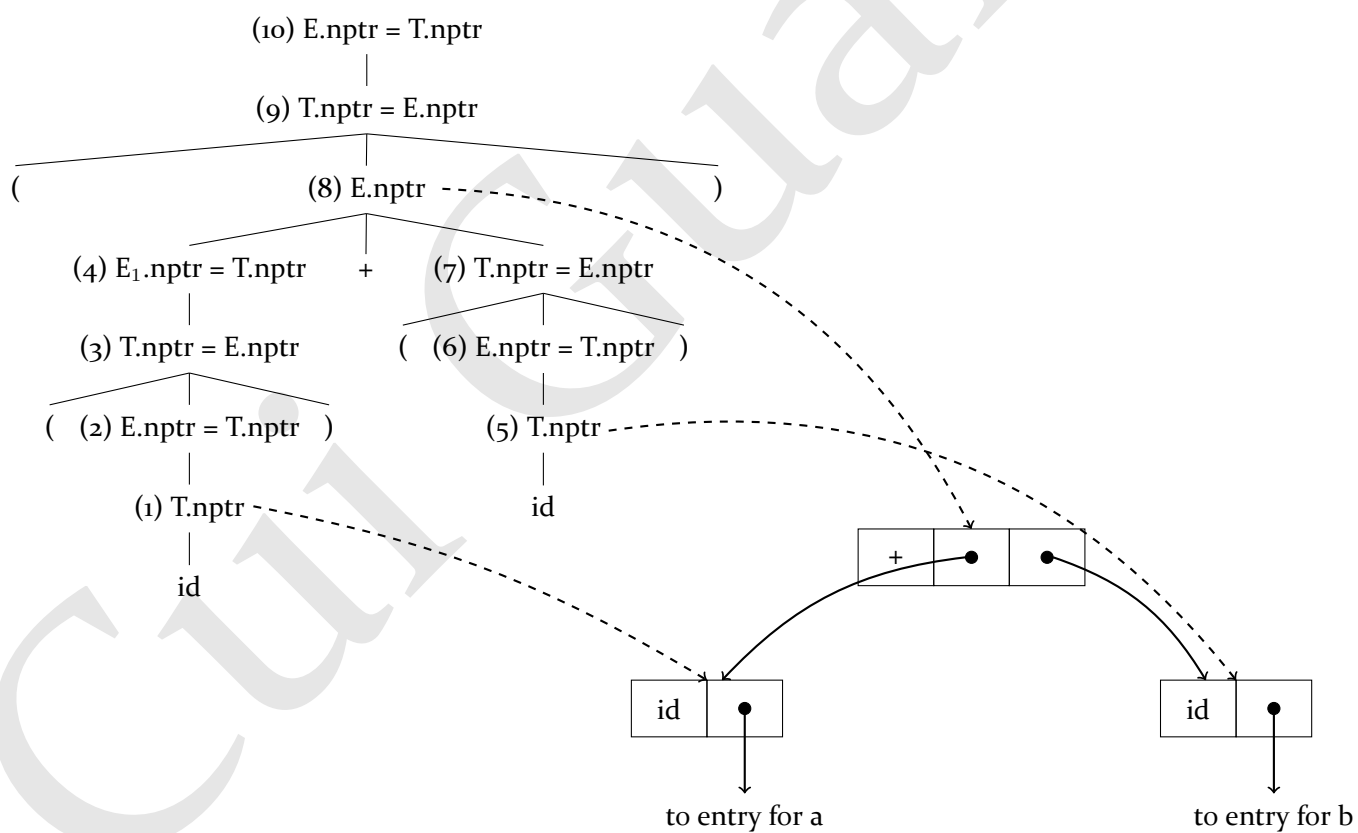


表 6.4 所示的属性文法是一个 S-属性文法，根据语法树和属性文法计算抽象语法树 (右下):



(2) 先给出文法对应的语法树，如下所示:





**P164.7** 下列文法由开始符号  $S$  产生一个二进制数，令综合属性  $val$  给出该数的值：

- $S \rightarrow L . L \mid L$
- $L \rightarrow LB \mid B$
- $B \rightarrow 0 \mid 1$

试设计求  $S.val$  的属性文法，其中，已知  $B$  的综合属性  $c$ ，给出由  $B$  产生的二进制位的结果值。例如，输入 101.101 时， $S.val = 5.625$ ，其中第一个二进制位的值是 4，最后一个二进制位的值是 0.125。

**解：** $B.c$  即为该位上的权值，模仿二进制转十进制的算法，容易设计出一个  $S$ -属性文法来求  $S.val$ 。

- $S \rightarrow L_1 . L_2 \{S.val = L_1.val + L_2.val\}$
- $S \rightarrow L \{S.val = L.val\}$
- $L \rightarrow L_1 B \{L_1.val + B.val\}$
- $L \rightarrow B \{L.val = B.val\}$
- $B \rightarrow 0 \{B.val = 0 * B.c\}$
- $B \rightarrow 1 \{B.val = 1 * B.c\}$

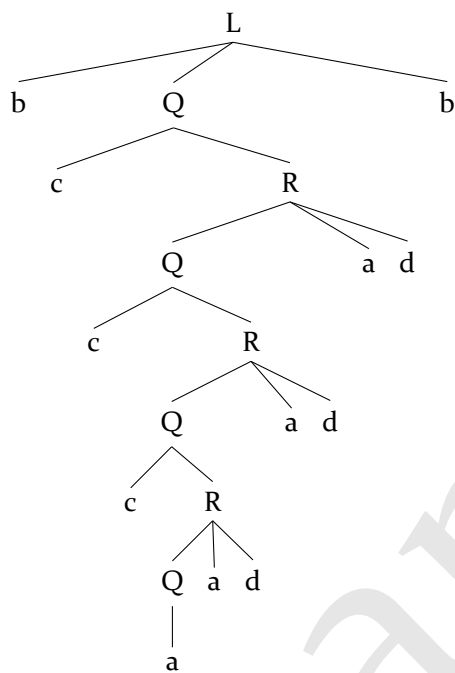
## 二、补充题

1. 文法及其相应的翻译方案为：

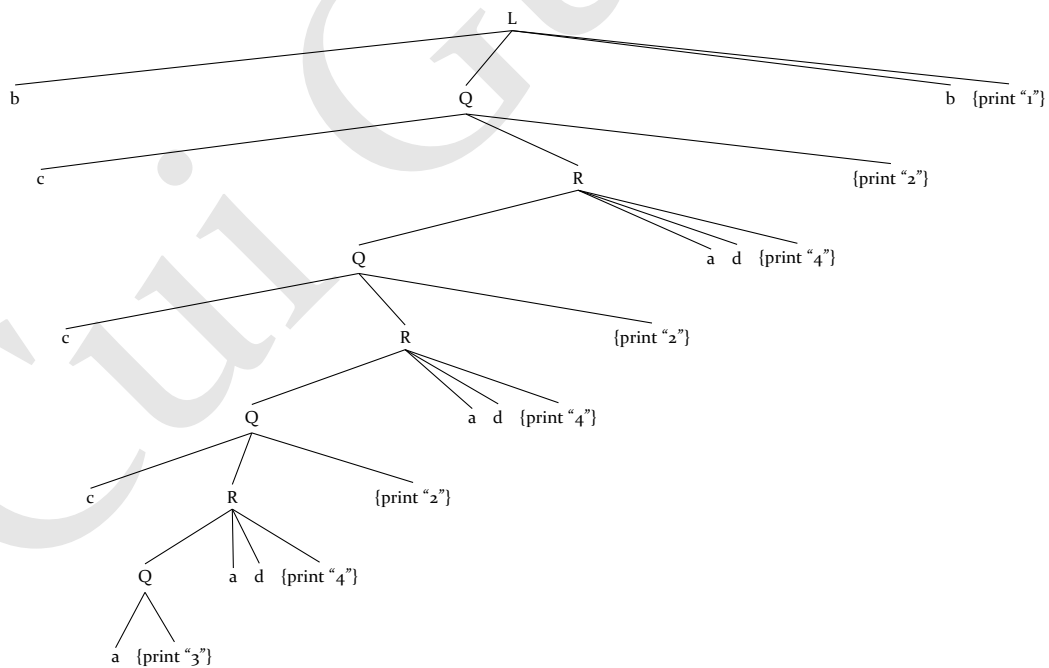
- $L \rightarrow bQb \{ \text{print "1"} \}$
- $Q \rightarrow cR \{ \text{print "2"} \}$
- $Q \rightarrow a \{ \text{print "3"} \}$
- $R \rightarrow Qad \{ \text{print "4"} \}$

求解，输入符号串为 **bcccaadadadb**，该输入符号串的输出是什么？

**解：**先画出语法树：



类似于后缀中缀表达式转后缀表达式，将每个动作加到产生式后，得到：



遍历树上的动作，得到输出为 **34242421**。