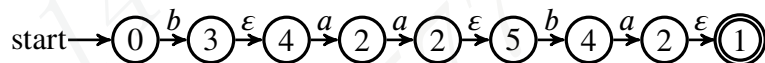


武汉大学计算机学院 2020 - 2021 学年第一学期
2018 级《编译原理》(期末考试参考答案 A)

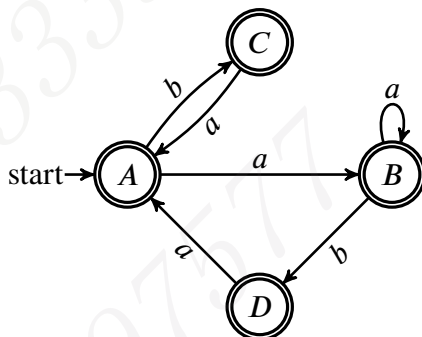
一、(1)



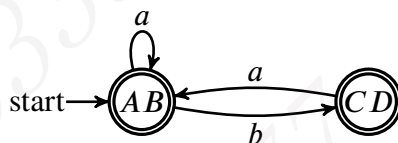
(2)

$$A = \{0, 1, 2, 5\}, B = \{1, 2, 5\}, C = \{3, 4\}, D = \{4\}.$$

状态转换图为:



(3) 最小 DFA 如下所示:



(4) 无连续的 b .

(5) $(b \mid \epsilon)(a \mid ab)^*$.

二、(1) 语句 “ $\neg(a \rightarrow a)$ ” 的最左推导如下:

$$\begin{aligned} F &\xRightarrow{lm} \neg F && \xRightarrow{lm} \neg(a \rightarrow F) \\ &\xRightarrow{lm} \neg(F) && \xRightarrow{lm} \neg(a \rightarrow a) \\ &\xRightarrow{lm} \neg(F \rightarrow F) \end{aligned}$$

(2) 消除左递归和左公因子后的文法如下:

$$\begin{aligned} F &\rightarrow \neg F F' \mid a F' \mid (F) F' \\ F' &\rightarrow \rightarrow F F' \mid \epsilon \end{aligned}$$

(3)

非终结符	First	Follow
F	$a, (, \neg$	$\rightarrow,), \$$
F'	\rightarrow, ϵ	$\rightarrow,), \$$

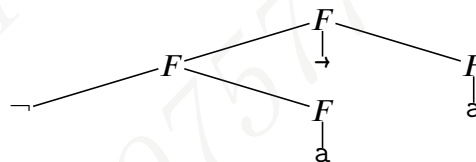
(4) LL(1) 分析表如下所示

	a	\neg	\rightarrow	()	\$
F	$F \rightarrow aF'$	$F \rightarrow \neg FF'$		$F \rightarrow (F)F'$		
F'			$F' \rightarrow \rightarrow FF' \mid \epsilon$		$F' \rightarrow \epsilon$	$F' \rightarrow \epsilon$

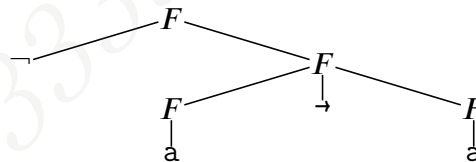
(5) 语句 “ $\neg(a \rightarrow a)$ ” 的分析过程如下所示:

剩余串	分析栈	分析动作
$\neg(a \rightarrow a)\$$	$F\$$	$F \rightarrow \neg FF'$
$\neg(a \rightarrow a)\$$	$\neg FF'\$$	match-advance
$(a \rightarrow a)\$$	$FF'\$$	$F \rightarrow (F)F'$
$(a \rightarrow a)\$$	$(F)F'F'\$$	match-advance
$a \rightarrow a)\$$	$F)F'F'\$$	$F \rightarrow aF'$
$a \rightarrow a)\$$	$aF')F'F'\$$	match-advance
$\rightarrow a)\$$	$F')F'F'\$$	$F' \rightarrow \rightarrow FF'$
$\rightarrow a)\$$	$\rightarrow FF')F'F'\$$	match-advance
$a)\$$	$FF')F'F'\$$	$F \rightarrow aF'$
$a)\$$	$aF'F')F'F'\$$	match-advance
$)\$$	$F'F')F'F'\$$	$F' \rightarrow \epsilon$
$)\$$	$F')F'F'\$$	$F' \rightarrow \epsilon$
$)\$$	$)F'F'\$$	match-advance
$\$$	$F'F'\$$	$F' \rightarrow \epsilon$
$\$$	$F'\$$	$F' \rightarrow \epsilon$
$\$$	$\$$	分析成功

三、 (1) 语句 “ $\neg a \rightarrow a$ ” 的两棵不同的语法树为:
语法树 1:



语法树 2:



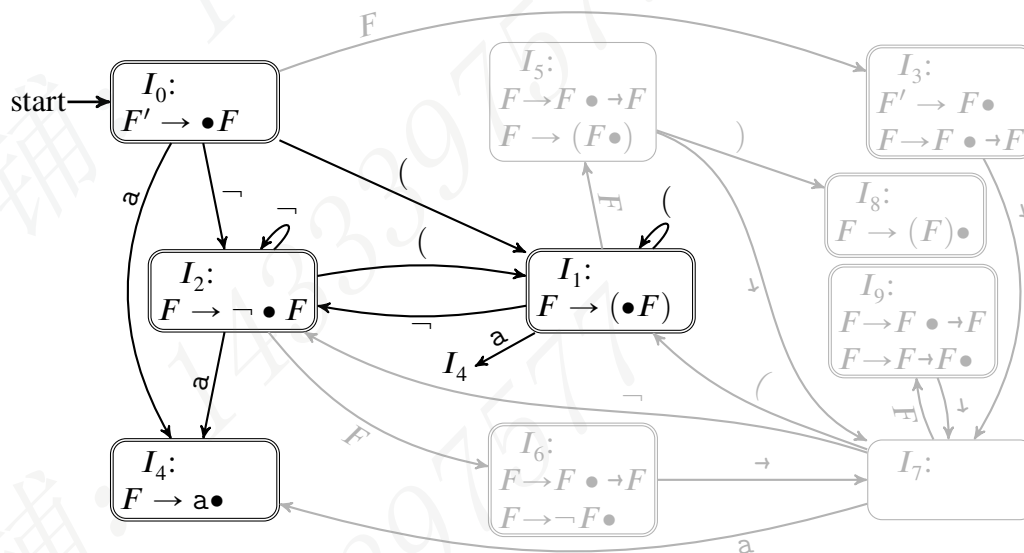
(2) 无二义文法:

$$\begin{aligned}
 F &\rightarrow N \rightarrow F \mid N \\
 N &\rightarrow \neg N \mid A \\
 A &\rightarrow (F) \mid a
 \end{aligned}$$

四、 (1) 状态 I_7 的 LR(0) 项目集为

$$\overline{\{F \rightarrow F \rightarrow \bullet F\}} \\ = \{F \rightarrow F \rightarrow \bullet F, F \rightarrow \bullet F \rightarrow F, F \rightarrow \bullet \neg F, F \rightarrow \bullet (F), F \rightarrow \bullet a\}.$$

(2) 仅由终结符号组成的活前缀对应的前缀 DFA 为:



其对应的正则表达式为: $(\neg | ()^* a ?$.

(3) $\text{Follow}(F) = \{ \rightarrow,), \$ \}$. 状态 I_6 和状态 I_9 面对 ' \rightarrow ' 有移进/归约冲突. 分析表如下所示:

状态	action						goto
	a	\neg	\rightarrow	()	\$	F
0	s4	s2		s1			3
1	s4	s2		s1			5
2	s4	s2		s1			6
3			s7			acc	
4			r4		r4	r4	
5			s7		s8		
6			r2		r2	r2	
7	s4	s2		s1			9
8			r3		r3	r3	
9			s7		r1	r1	

(4) 语句 " $\neg a \rightarrow a$ " 的分析过程如下所示:

剩余串	分析栈	分析动作
$\neg a \rightarrow a \$$	0	shift
$a \rightarrow a \$$	0 \neg 2	shift
$\rightarrow a \$$	0 \neg 2a4	reduce $F \rightarrow a$

$\rightarrow a\$$	0 \rightarrow 2F6	reduce $F \rightarrow \neg F$
$\rightarrow a\$$	0F3	shift
a\$	0F3 \rightarrow 7	shift
\$	0F3 \rightarrow 7a4	reduce $F \rightarrow a$
\$	0F3 \rightarrow 7F9	reduce $F \rightarrow F \rightarrow F$
\$	0F3	reduce 分析成功

五、 (1)

产生式	语义规则
$F \rightarrow F_1 \rightarrow F_2$	if ($F.is_neg == True$) then $F.nnf = "(" + F_1.nnf + ")" \wedge "(" + F_2.nnf + ")"$ else $F.nnf = F_1.nnf + "\vee" + F_2.nnf$
$F \rightarrow \neg F_1$	$F.nnf = F_1.nnf$
$F \rightarrow (F_1)$	$F.nnf = F_1.nnf$
$F \rightarrow a$	if ($F.is_neg == True$) then $F.nnf = "\neg" + a.lexeme$ else $F.nnf = a.lexeme$

(2) $((A) \wedge (B) \vee C) \wedge (\neg D \vee E)$

六、

```

L0: if (g<h) goto L1      |      z := t1
    ifnot (i<j) goto L1   | L1: ifnot (a<b) goto L0
    t0 := y+1              |      ifnot (c<d) goto L2
    x := t0                |      if (e<f) goto L0
    t1 := x+y              | L2:

```

七、 `char a[][8] = "Stay", "Safe", "From", "Covid"` 把字符串保存在局部数组中, `init(s)` 初始化 `s[i]` 为指向局部数组的指针, 造成 `dangling reference`. 而 `char *a[] = "Stay", "Safe", "From", "Covid"` 初始化 `a` 为指向字符串常量的指针, 不会产生 `dangling reference`.