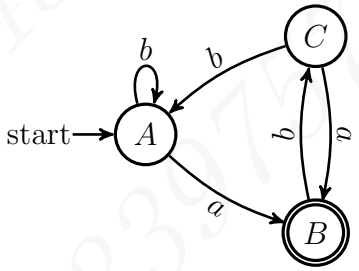
武汉大学计算机学院2010-2011学年第一学期
2008级《编译原理》参考答案

一、 (1)

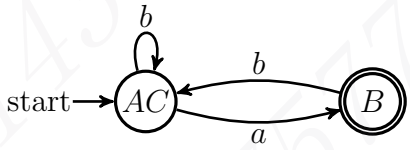
(2)

$$\begin{aligned} A &= \{0,3,4\} \\ B &= \{1,2,5\} \\ C &= \{1,3,4\} \end{aligned}$$

状态转换图为：



(3) 最小状态自动机如下：



(4) 以a结尾，且没有连续的a.

二、 (1) 最左推导如下：

$$\begin{aligned} S &\underset{lm}{\Longrightarrow} aAaBb \\ &\underset{lm}{\Longrightarrow} aaAaBb \\ &\underset{lm}{\Longrightarrow} aaaBb \\ &\underset{lm}{\Longrightarrow} aaabBb \\ &\underset{lm}{\Longrightarrow} aaabb \end{aligned}$$

(2) $\text{First}(S) = \{a, b\}$; $\text{First}(A) = \{\varepsilon, a\}$; $\text{First}(B) = \{\varepsilon, b\}$.
$\text{Follow}(S) = \{\$\}$; $\text{Follow}(A) = \{a\}$; $\text{Follow}(B) = \{b\}$.

(3)

	a	b	\$
S	$S \to aAaBb$	$S \to bBbAa$	
A	$A \to aA$; $A \to \varepsilon$		
B		$b \to bB$; $B \to \varepsilon$	

(4) $r = aaa^*bb^*|bbb^*aa^*$ 或$r = aa^*ab^*b|bb^*ba^*a$.

(5) 与$G(S)$等价的LL(1)文法：

$$\begin{aligned} S &\rightarrow A \mid D \\ A &\rightarrow aaB \\ B &\rightarrow aB \mid bC \\ C &\rightarrow bC \mid \varepsilon \\ D &\rightarrow bbE \\ E &\rightarrow bE \mid aF \\ F &\rightarrow aF \mid \varepsilon \end{aligned}$$

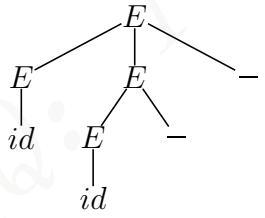
或与$G(S)$等价的LL(1)文法：

$$\begin{aligned} S &\rightarrow aaAbB \mid bbBaA \\ A &\rightarrow aA \mid \varepsilon \\ B &\rightarrow bB \mid \varepsilon \end{aligned}$$

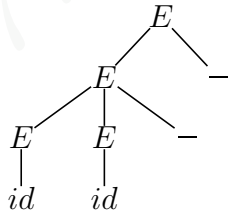
(6) SLR(1)分析器在移进a后进入状态$\{ S \rightarrow a \bullet AaBb,\ A \rightarrow \bullet aA,\ A \rightarrow \bullet \}$，面对输入$a$可移进到状态$\{ A \rightarrow a \bullet A,\ A \rightarrow \bullet aA,\ A \rightarrow \bullet \}$；而$a \in \text{Follow}(A)$，所以也可以用产生式$A \rightarrow \varepsilon$归约, 即有移进/归约冲突。

三、 (1) "$id\ id\ -\ -$"的两颗不同的语法树：

语法树1：



语法树2：



(2)

$$\begin{aligned} E &\rightarrow E- \mid M \\ M &\rightarrow MM- \mid id \end{aligned}$$

四、 (1) 识别活前缀的自动机在吃进 $EEE+$ 之后到达状态I_4，不能再接受任何符号，因此 $EEE + E$ 不是活前缀；而识别活前缀的自动机在吃进 $EEEE$ 之后到达状态I_2, 所对应的LR(0)项目集即是其有效项目集：

$$\begin{aligned} &\overline{\{E \rightarrow EE \bullet +,\ E \rightarrow EE \bullet *,\ E \rightarrow E \bullet E+,\ E \rightarrow E \bullet E*\}} \\ =&\{E \rightarrow EE \bullet +,\ E \rightarrow EE \bullet *,\ E \rightarrow E \bullet E+,\ E \rightarrow E \bullet E*, \\ &E \rightarrow \bullet EE+,\ E \rightarrow \bullet EE+,\ E \rightarrow \bullet id\} \end{aligned}$$

(2) First$(E) = \{ id \}$, Follow$(E) = \{ +, *, id, \$ \}$.

(3) SLR分析表如下所示：

	action				goto
状态	$*$	$+$	id	\$	E
0			s5		1
1			s5	acc	2
2	s3	s4	s5		2
3	r2	r2	r2	r2	
4	r1	r1	r1	r1	
5	r3	r3	r3	r3	

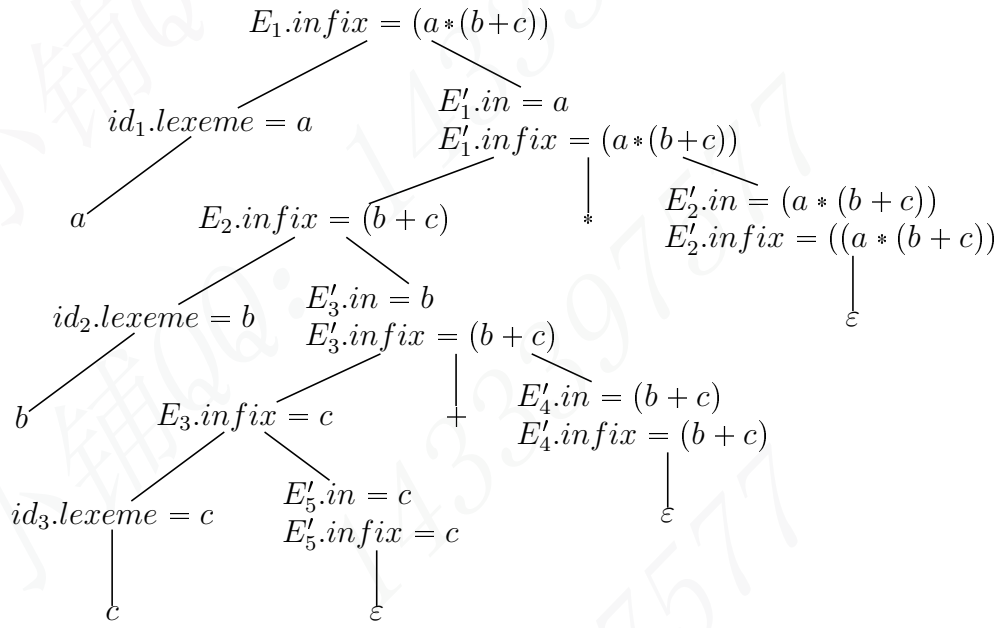
(4) "$id\ id + id*$"的分析过程如下所示：

剩余串	分析栈	分析动作
$id\ id + id * \$$	0	shift
$id + id * \$$	$0id5$	reduce $E \to id$
$id + id * \$$	$0E1$	shift
$+id * \$$	$0E1id5$	reduce $E \to id$
$+id * \$$	$0E1E2$	shift
$id * \$$	$0E1E2 + 4$	reduce $E \to EE+$
$id * \$$	$0E1$	shift
$*\$$	$0E1id5$	reduce $E \to id$
$*\$$	$0E1E2$	shift
\$	$0E1E2 * 3$	reduce $E \to EE*$
\$	$0E1$	分析成功

五、 (1)

产生式	语义规则
$E \to id\, E'$	$E.infix = E'.infix$ $E'.in = id.lexeme$
$E' \to E + E'_1$	$E'.infix = E'_1.infix$ $E'_1.in =$ "(" $+ E'.in +$ " + " $+ E.infix +$ ")"
$E' \to E * E'_1$	$E'.infix = E'_1.infix$ $E'_1.in =$ "(" $+ E'.in +$ " $*$ " $+ E.infix +$ ")"
$E' \to \varepsilon$	$E'.infix = E'.in$

(2) "$abc + *$"的附注语法树:



六、

```
l7:  if (a > b) goto l0
     goto l9
l0:  if (c > d) goto l8
     goto l9
l8:  if (e > f) goto l1
     goto l2
l1:  if (i > j) goto l2
     goto l3
l2:  t0 := x + 1
     x := t0
     goto l5
l3:  goto l9
l5:  if (m > n) goto l4
     goto l6
l4:  goto l7
l6:  t1 := y + 2
     y := t1
     goto l7
l9:
```

七、 程序在调用foo()后的内存格局如下：

address	memory	note
	.....	
x	0	← i
x-4	20	← a[1]
x-8	10	← a[0]
	.....	

当for语句执行到i=2时，循环条件满足，这时数组已经越界，a[2]实际上是i的地址。语句“a[i] -= 3;”把i修改为-1，当程序执行到i++时，i又被修改为0. 循环条件再次成立，如此反复，循环不能退出。