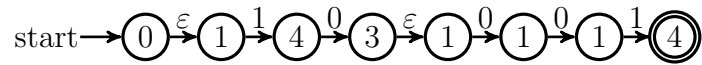


武汉大学计算机学院2011-2012学年第一学期
2009级《编译原理》参考答案

一、 (1)



(2)

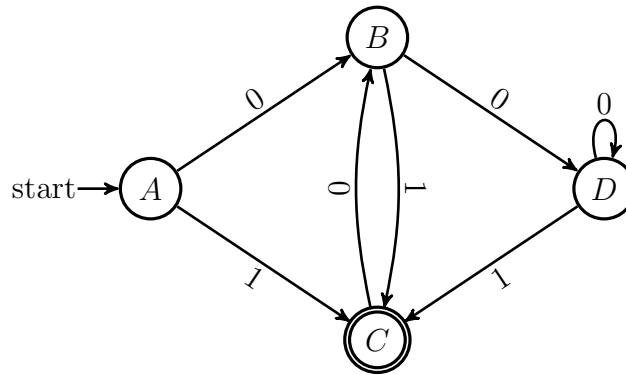
$$A = \{0, 1, 2\}$$

$$B = \{1, 2, 3\}$$

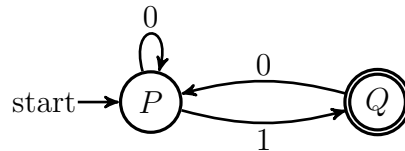
$$C = \{4, 5\}$$

$$D = \{1, 2\}$$

状态转换图为:



(3) 两个等价状态集合 $P = \{A, B, D\}$, $Q = \{C\}$. 最小状态自动机如下:



(4) 以1结尾, 且没有连续的1.

(5) $r = 0^*(100^*)^*1$

二、 (1) 最左推导如下:

$$\begin{aligned}
 D &\xRightarrow{lm} TL \\
 &\xRightarrow{lm} \text{int } L \\
 &\xRightarrow{lm} \text{int } L, \text{id} \\
 &\xRightarrow{lm} \text{int id, id}
 \end{aligned}$$

(2) 消除左递归后的文法如下：

$$\begin{aligned} D &\rightarrow T L \\ T &\rightarrow \text{int} \mid \text{char} \\ L &\rightarrow \text{id} L' \\ L' &\rightarrow , \text{id} \mid \varepsilon \end{aligned}$$

(3) $\text{First}(D) = \text{First}(T) = \{\text{int}, \text{char}\}$; $\text{First}(L) = \{\text{id}\}$; $\text{First}(L') = \{\varepsilon, ', '\}$.
 $\text{Follow}(S) = \text{Follow}(L) = \text{Follow } L' = \{\$ \}$; $\text{Follow}(T) = \{\text{id}\}$.

(4)

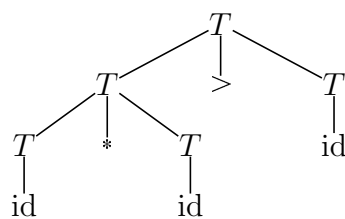
| | id | , | int | char | \$ |
|------|------------------------------|---------------------------------|----------------------------|-----------------------------|------------------------------|
| D | | | $D \rightarrow TL$ | $D \rightarrow TL$ | |
| T | | | $T \rightarrow \text{int}$ | $T \rightarrow \text{char}$ | |
| L | $L \rightarrow \text{id} L'$ | | | | |
| L' | | $L' \rightarrow , \text{id} L'$ | | | $L' \rightarrow \varepsilon$ |

(5) 语句“int id, id”的分析过程如下所示：

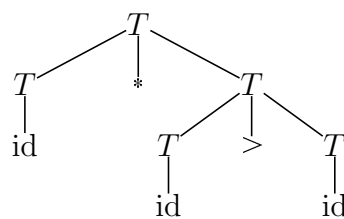
| 剩余串 | 分析栈 | 分析动作 |
|---------------|-------------|----------------------------------|
| int id, id \$ | $D\$$ | $D \rightarrow TL$ |
| int id, id \$ | $TL\$$ | $T \rightarrow \text{int}$ |
| int id, id \$ | int $L\$$ | match-advance |
| id, id \$ | $L\$$ | $L \rightarrow \text{id} L'$ |
| id, id \$ | id $L'\$$ | match-advance |
| , id \$ | $L'\$$ | $L' \rightarrow , \text{id} L''$ |
| , id \$ | , id $L'\$$ | match-advance |
| id \$ | id $L'\$$ | match-advance |
| \$ | $L'\$$ | $L' \rightarrow \varepsilon$ |
| \$ | \$ | 分析成功 |

三、 (1) “id * id > id”的两颗不同的语法树：

语法树1：



语法树2：



(2)

$$\begin{aligned} T &\rightarrow T * F \mid F \\ F &\rightarrow \text{id} > F \mid \text{id} \end{aligned}$$

- 四、 (1) 识别活前缀的自动机在吃进 id 之后到达状态 I_6 ，不能再接受任何符号，因此 id * id > 不是活前缀；而识别活前缀的自动机在吃进 $T * T >$ 之后到达状态 I_4 ，所对应的LR(0)项目集即是其有效项目集：

$$\begin{aligned} & \overline{\{T \rightarrow T \bullet > T\}} \\ & = \{T \rightarrow T \bullet > T, T \rightarrow \bullet T * T, T \rightarrow \bullet T > T, T \rightarrow \bullet \text{id}\} \end{aligned}$$

(2) $\text{First}(T) = \{\text{id}\}$, $\text{Follow}(T) = \{*, >, \$\}$.

(3) SLR分析表如下所示：

| | action | | | | goto |
|----|-----------|-----------|----|-----|------|
| 状态 | * | > | id | \$ | E |
| 0 | | | s6 | | 1 |
| 1 | s2 | s4 | | acc | |
| 2 | | | s6 | | 3 |
| 3 | r1 | s4 | | r1 | |
| 4 | | | s6 | | 5 |
| 5 | s2 | s4 | | r2 | |
| 6 | r3 | r3 | | r3 | |

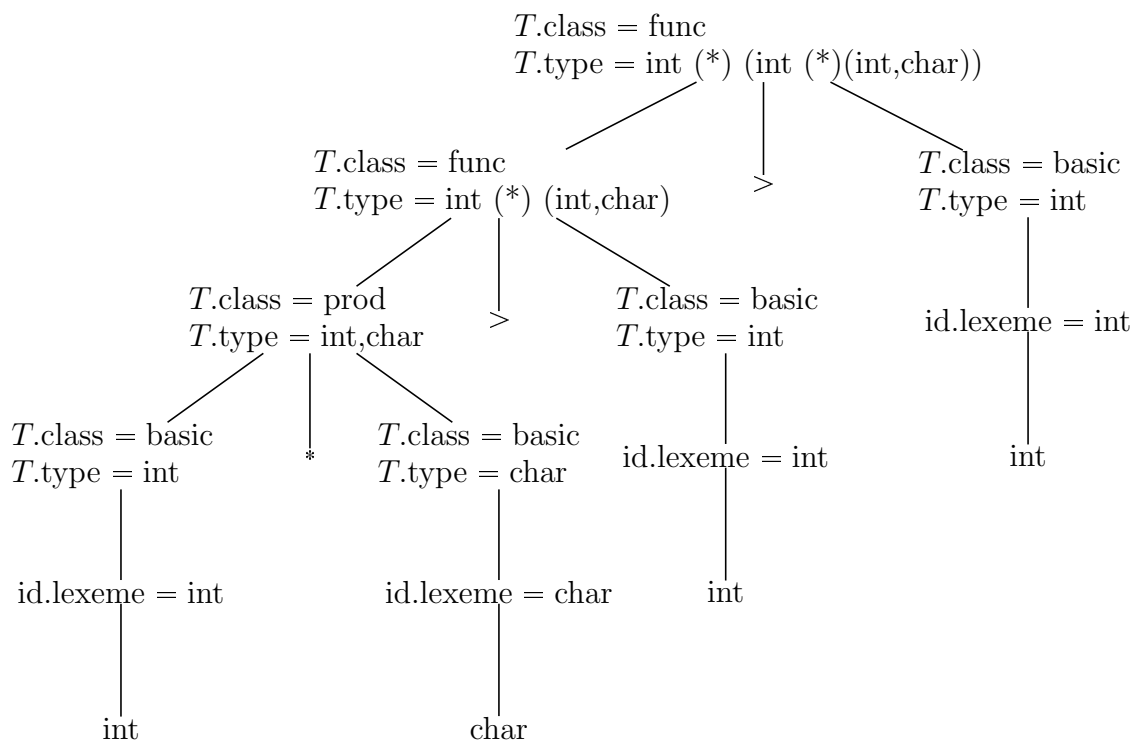
(4) “id * id > id”的分析过程如下所示：

| 剩余串 | 分析栈 | 分析动作 |
|----------------|------------------|----------------------------------|
| id * id > id\$ | 0 | shift |
| *id > id\$ | 0id6 | reduce $T \rightarrow \text{id}$ |
| *id > id\$ | 0T1 | shift |
| id > id\$ | 0T1 * 2 | shift |
| > id\$ | 0T1 * 2id6 | reduce $T \rightarrow \text{id}$ |
| > id\$ | 0T1 * 2T3 | shift |
| id\$ | 0T1 * 2T3 > 4 | shift |
| \$ | 0T1 * 2T3 > 4id6 | reduce $T \rightarrow \text{id}$ |
| \$ | 0T1 * 2T3 > 4T5 | reduce $T \rightarrow T > T$ |
| \$ | 0T1 * 2T3 | reduce $T \rightarrow T * T$ |
| \$ | 0T1 | 分析成功 |

五、 (1)

| 产生式 | 语义规则 |
|---------------------------|---|
| $T \rightarrow T_1 * T_2$ | $T.\text{class} = \text{prod}.$ $T.\text{type} = T_1.\text{type} + “,” + T_2.\text{type}.$ |
| $T \rightarrow T_1 > T_2$ | if $T_2.\text{class} = \text{prod}$ then error else $T.\text{class} = \text{func}.$ $T.\text{type} = T_2.\text{type} + “(*) (“ + T_1.\text{type} + “)”.$ |
| $T \rightarrow (T_1)$ | $T.\text{type} = T_1.\text{type}.$ $T.\text{class} = T_1.\text{class}.$ |
| $T \rightarrow \text{id}$ | $T.\text{class} = \text{basic}.$ $T.\text{type} = \text{id.lexme}.$ |

- (2) “((int * char) > int) > int”的附注语法树(注：限于篇幅，省略了语法树中的括号部分):



六、

```

13: ifnot (a>b) goto 14      10: goto 13
    if (c>d) goto 14        11: ifnot (m>n) goto 12
    ifnot (e>f) goto 10      goto 14
    if (i>j) goto 10        12: t1 := y + 2
    t0 := x + 1              y := t1
    x := t0                  goto 13
    goto 11                  14:
  
```

七、 Program 1中的main用传值方式调用foo时，首先对实参1, A和B 求值，即将指向函数A和B的指针传给foo，并没有调用函数A和B。执行((b? f: g)) 后结果为指向函数A的指针，((b? f: g))()即调用函数A，故返回10正常退出。即相当于传名。

而Program 2中的main用传值方式调用foo时，首先对实参1, x + y和x / y求值求值，而y = 0, 因此报浮点数异常。