

## 北京理工大学 2017—2018 学年第二学期

## 计算机科学与技术类《编译原理与设计》试卷 B (2018.06)

班级\_\_\_\_\_学号\_\_\_\_\_姓名\_\_\_\_\_成绩\_\_\_\_\_

题号	一	二	三	四	五	六	七	八	九	总分
成绩										

**\*\* 注意：各题均必须答在试卷上，书写不下可以写在试卷背面。**

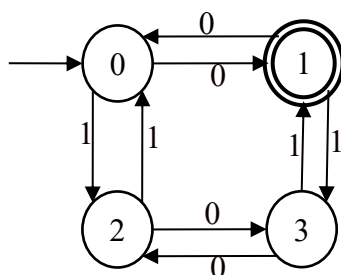
## 一. 判断题

(10 分)

在下面答题表中填上“√”或“×”。

题号	1	2	3	4	5	6	7	8	9	10
答案										

- 通常情况下 C 语言程序是编译执行的, Python 语言程序是解释执行的。
- 对集合  $A$ , 有  $A^+ = AA^*$ , 所以对任意集合  $A$ , 一定有  $A^+ \neq A^*$ 。
- DISPLAY 表用来记录每层过程的最新活动记录地址, 它的大小在运行时确定。
- 文法若存在左公因子, 则在自上而下语法分析过程中会因为无匹配操作而出现死循环。
- 二义文法一定不是上下文有关文法。
- 句柄是最长活前缀的一个后缀。
- 单词的属性字一般应该包括单词类别和单词的内码。
- LR 分析过程中使用的分析栈分为两部分: 状态栈和符号栈。
- DFA M 的状态表表示如下图, 则该 DFA 可以接受符号串 0100110000111。



- 语法制导翻译方法只能用来产生中间代码, 不能用来生成目标代码。

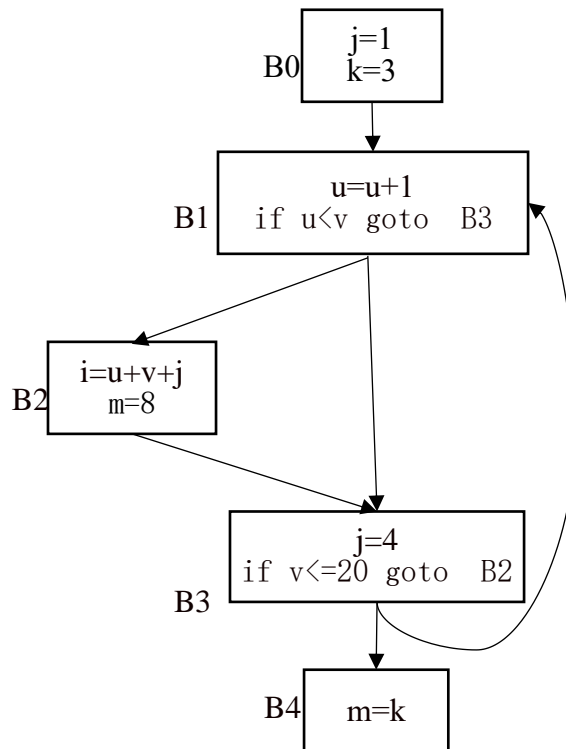
## 二、单项选择题（在下面答题表中填上答案）

（10分）

题号	1	2	3	4	5	6	7	8	9	10
答案										

1. 如下图所示，循环不变运算“j=4”不能提到循环外面的原因在于【 】

- A) j 所在的结点不是循环的所有出口结点的必经结点；
- B) j 在 L 中其它地方被定值；
- C) 循环中所有 j 的引用点不是只有循环中 j 的定值才能到达；
- D) j 在离开 L 后不再是活跃的；



2. Yacc 实现用到的算法是（ ）

- A) 子集法
- B) 正规式转为 NFA
- C) 划分法
- D) LALR(1)项目集规范簇构造

3. 第十章 PL/0 编译程序的语法分析程序用的语法分析方法为【 】

- A) LR 分析法
- B) 算符优先分析法
- C) LL(1)分析表
- D) 递归下降分析法

4. 基本块优化中不能完成的优化工作是【 】

- A) 删除公共子表达式
- B) 常量合并
- C) 删除无用赋值
- D) 运算强度削弱

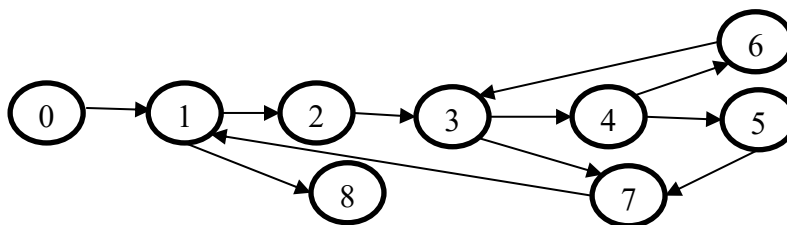
5. 引入中间代码的目的不包括【 】

- A) 方便生成目标代码
- B) 便于优化
- C) 减少对主存容量的要求
- D) 便于移植

6. 数据流方程 
$$\begin{cases} \text{out}(B) = (\text{in}(B) - \text{kill}(B)) \cup \text{gen}(B) \\ \text{in}(B) = \bigcup_{P \in \text{Pred}(B)} \text{out}(P) \end{cases}$$
 用来计算的数据流信息为【 】。

- A) 到达定值
- B) 活跃变量
- C) 可用表达式
- D) 以上都不是

7. 在语句  $Z=X*Y-H$  中，定义性出现的变量是【 】  
 A)Z                  B)X                  C)Y                  D)H
8. 设有文法  $S \rightarrow SS^*|SS+|a$ ，则下面的符号串是该文法的句子的是【 】  
 A) $aa+0^*$                   B) $aa+aa^*+a$                   C) $aa+aaa^*++$                   D) $aS+a^*$
9. 设有 C 语言程序段如下：  
 While( $i==j$ )  
 { $m=0$ ;  
 printf("m=%d\n",m);  
 $i++$ ;}  
 则经过词法分析后可以识别的单词个数是【 】  
 A)20                  B)21                  C)22                  D)23
10. 如下图所示的程序控制流图中有【 】个循环



- A)1                  B)2                  C)3                  D)4

### 三. 填空题

(23 分)

- 设有字母表  $\Sigma=\{a,b\}$  上的语言  $L(G)=\{a^i b^j | 1 \leq j \leq 2j\}$ ，描述该语言的上下文无关文法【 】。
- 字符串  $\alpha$  的长度为  $n$ ，则字符串  $\alpha$  的前缀有【 】个，字符串  $\alpha$  的子串有【 】个。
- EQN 的编译器是用 C 语言编写的将 EQN 文本转化为格式化的 TROFF 文本，用 T 型图描述怎么借助 PDP—11 机器上运行的 C 语言编译器得到 PDP—11 机器上可运行的 EQN 的编译器【 】。
- 设有文法  $G[A]$ :

$$A \rightarrow a|(R) \quad T \rightarrow A, T|A \quad R \rightarrow T$$

根据上述文法填写如下的算符优先关系表：

	$a$	$($	$)$	$,$	$\#$
$,$					

5. 设有文法  $G[A]$ :

$$A \rightarrow BCc|eDB \quad B \rightarrow \varepsilon|bcD \quad C \rightarrow DaB|ca \quad D \rightarrow \varepsilon|dD$$

根据上述文法填写如下 LL(1) 分析表的内容:

	$a$	$b$	$c$	$d$	$\#$
$D$					

6. 设有文法:

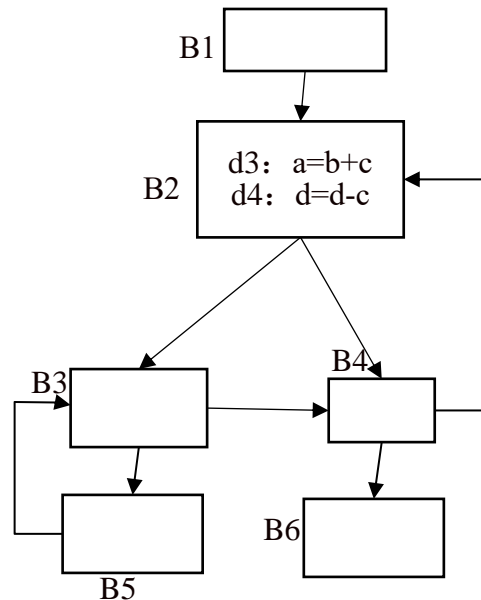
$$S \rightarrow aSDe|a \quad D \rightarrow Db|b$$

与该文法等价的 LL(1) 文法为

【\_\_\_\_\_】。

7. 下表为右图所示流图的到达定值信息, 则变量  $d$  在  $d4$  点的  $ud$  链为【\_\_\_\_\_】。

<b>in(B1)</b>	$\Phi$
<b>out(B1)</b>	1(b),2(c)
<b>in(B2)</b>	1(b),2(c),3(a),4(d),5(d),6(c),7(e)
<b>out(B2)</b>	1(b),2(c),3(a),4(d),6(c),7(e)



8.  $a+b*(c+d)*(e+f)$  的逆波兰式为 【\_\_\_\_\_】。

9. 设有文法  $G[A]$ :

$$A \rightarrow AB+|B \quad B \rightarrow BC*|C \quad C \rightarrow C\uparrow|a$$

该文法的句型  $BBC\uparrow*+$  的直接短语是 【\_\_\_\_\_】, 素短语是 【\_\_\_\_\_】, 句柄是 【\_\_\_\_\_】。

10. C 语言程序段:

$J=1;$

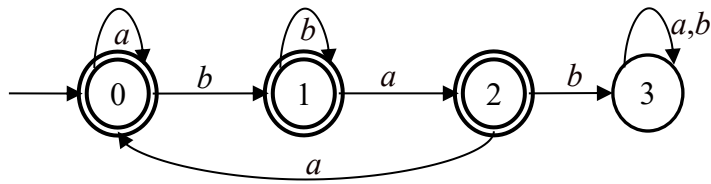
$B=(5 * J - 2)+J;$

优化后的四元式代码为 【\_\_\_\_\_】; 使用的优化技术是 【\_\_\_\_\_】。

#### 四. 解答题

(8 分)

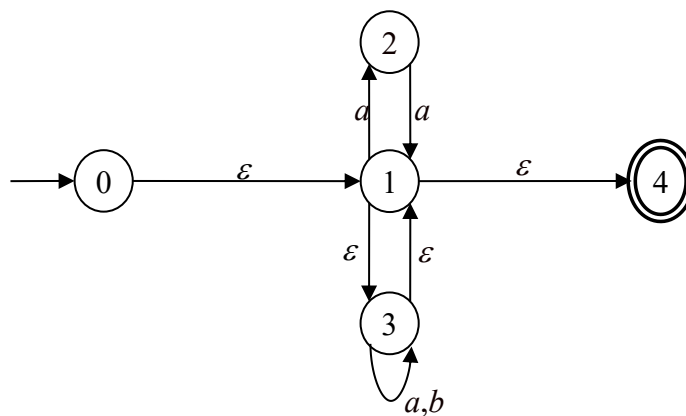
设 DFA  $M$  的状态图如下, 给出  $M$  所识别语言的正规式表示(要求给出求解的过程)



#### 五. 解答题

(9 分)

用子集法将下面的 **NFA** 确定化, 用划分法将确定化的 **DFA** 最小化 (要求给出确定化和最小化过程)。



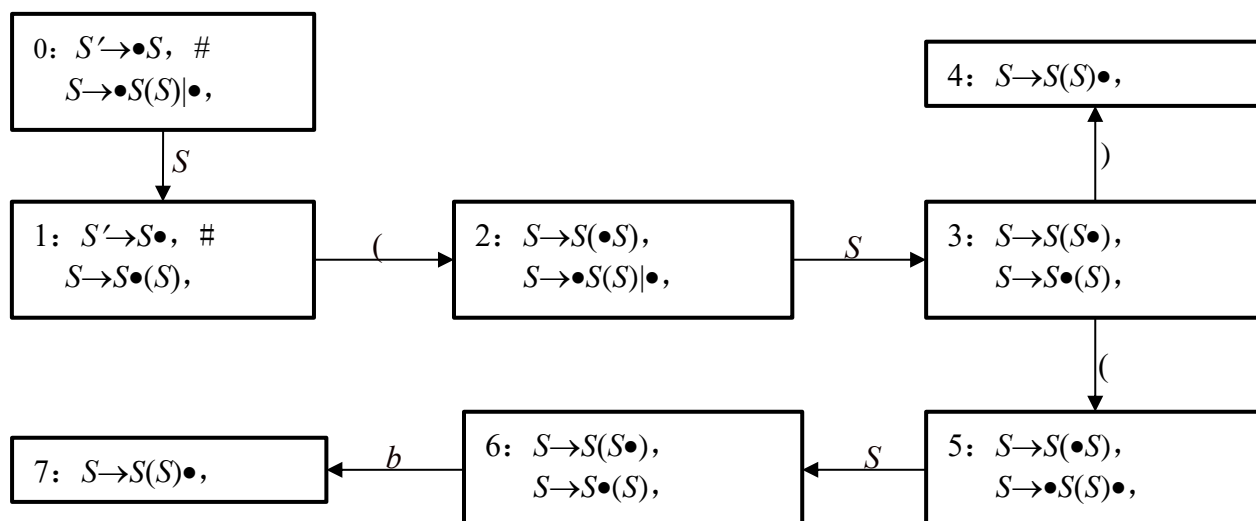
## 六. 解答题

(12 分)

设有文法  $G[S]$ :

$S \rightarrow S(S) | \varepsilon$  ①|②

1. 将下图的识别该文法 LR(1) 项目有效活前缀的 DFA 补充完整。



2. 根据上面的 DFA，填写该文法的如下部分 LR(1) 分析表。

state	ACTION			GOTO
	(	)	#	
0				1
1				
2				
3				

## 七. 解答题

(12 分)

设有文法:

$S \rightarrow S \wedge S | S \vee S | S \Rightarrow S | \neg S | 0 | 1$     ①|②|③|④|⑤|⑥

1.证明文法为二义性文法。

2.原则上二义文法不能使用 LR 分析,但是根据一定的约定原则,构造出无二义性的 LR 分析表,就可以应用于该文法的分析。设构造出的该文法的无二义性的 LR 分析表如下:

state	ACTION							GOTO
	$\wedge$	$\vee$	$\Rightarrow$	$\neg$	0	1	#	S
0				S2	S3	S4		1
1	S5	S6	S7				acc	
2				S2	S3	S4		8
3	r5	r5	r5	r5	r5	r5	r5	
4	r6	r6	r6	r6	r6	r6	r6	
5				S2	S3	S4		9
6				S2	S3	S4		10
7				S2	S3	S4		11
8	S5	S6	S7	r4	r4	r4	r4	
9	r1	r1	r1	r1	r1	r1	r1	
10	S5	r2	r2	r2	r2	r2	r2	
11	S5	S6	r3	r3	r3	r3	r3	

根据上述 LR 分析表,给出句子  $0 \Rightarrow 1 \wedge 1 \wedge 0 \neg 0 \vee 1$  的 LR 分析过程的前 11 步。

步骤	状态栈	符号栈	待扫描符号串	分析动作
0	0	#	$0 \Rightarrow 1 \wedge 1 \wedge 0 \neg 0 \vee 1 \#$	
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

3.画出句子  $0 \Rightarrow 1 \wedge 1 \wedge 0 \neg 0 \vee 1$  的语法分析树。



## 八. 解答题

(10 分)

有下列 C 语言程序段：

```
int j;
int sum = 0;
for(int i = 2; i < 100; i++){
    j = 2;
    while(1){
        if(j >= i) goto L2;
        if(i % j == 0) goto L1;
        j++;
    }
L1:
    sum++;
L2:
    ;
}
```

将该程序段对应的四元式形式的中间代码填入下表中；

**注意：**

A)无条件转移操作符用“j”表示，条件成立转移的操作符用“jr”表示，条件不成立转移的操作符用“jf”表示；

B)语句标号的地址使用四元式序列的序号表示，序号用①，②，…表示。

序号	四元式代码		四元式代码
1		14	
2		15	
3		16	
4		17	
5		18	
6		19	
7		20	
8		21	
9		22	
10		23	
11		24	
12		25	
13		26	

## 九. 解答题

(6 分)

对如下的 x86 汇编代码片段，划分基本块（基本块划分直接标记在源代码中），并给出程序流程图。

```
    mov    ecx, 2        //ecx 置为 2
L1:  mov    esi, 2        //esi 置为 2
    cmp    ecx, esi      //比较 ecx 和 esi
    jle    L3            //小于等于跳转
L2:  mov    eax, ecx      //eax 置为 ecx
    idiv   esi           //eax / esi, 余数在 edx
    test   edx, edx      //按位与
    je     L3            //等于跳转
    inc    esi           //esi 自增
    cmp    esi, ecx      //比较 esi 和 ecx
    jl     L2            //小于跳转
L3:  inc    ecx           //ecx 自增
    cmp    ecx, 100      //比较 ecx 和 100
    jl     L1            //小于跳转
```