

### 诚信保证

本人知晓我校考场规则和违纪处分条例的有关规定，保证遵守考场规则，诚实做人。

本人签字：\_\_\_\_\_

编号：\_\_\_\_\_

西北工业大学考试试题（卷）

2020 —2021 学年第 一 学期

开课学院\_\_\_\_\_

课程 编译原理

学时 48

考试日期2020 年 月 11 日

考试时间 2 小时

考试形式（闭）（A）卷

题号	一	二	三	四	五	六	七	总分
得分								

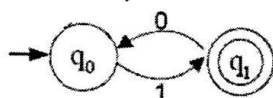
考生班级

学 号

姓 名

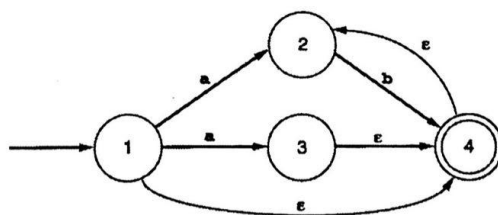
一. 选择题（单选题，每题 3 分，共 15 分）

1. 下述DFA接受集与正规表达式 13 定义的正规集相同。



A.  $1|(10)^*$  B.  $1(01)^*$  C.  $1|(101)^*$  D.  $1(101)^*$

2. 对于下图中的NFA，根据由NFA构造DFA的算法，构造的与之等价的DFA的状态为 C。



A.  $\{1, 4\}, \{3, 4\}, \{4, 2\}$

B.  $\{1, 4\}, \{3, 4, 2\}, \{4, 2\}$

C.  $\{1, 4, 2\}, \{3, 4, 2\}, \{4, 2\}$

D.  $\{1, 4, 2\}, \{3, 4\}, \{4\}$

注：1. 命题纸上一般不留答题位置，试题请用小四、宋体打印且不出框。

2. 命题教师和审题教师姓名应在试卷存档时填写。

共 7 页 第 1 页

西北工业大学命题专用纸

3. 设有文法G[lexp]

- (1)  $lexp \rightarrow \text{number}$  (2)  $lexp \rightarrow (op\ lexp\ seq)$  (3)  $op \rightarrow +$  (4)  $op \rightarrow -$  (5)  $op \rightarrow *$   
 (6)  $lexp\ seq \rightarrow lexp\ seq\ lexp$  (7)  $lexp\ seq \rightarrow lexp$

则  $\text{Follow}(lexp) = \underline{A}$  .

- A.  $\{), \$\}$  B.  $\{ \$, ), ( \}$  C.  $\{ (, ), \text{number} \}$  D.  $\{ \$, (, ), \text{number} \}$

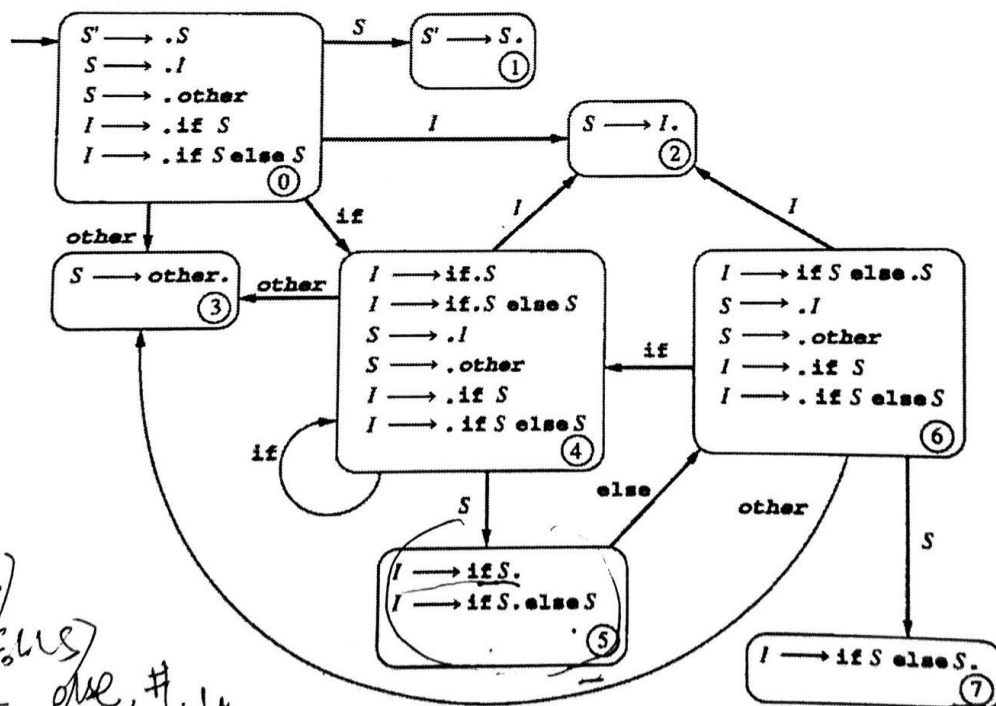
4. 下列C或Java程序错误属于编译器检查的静态语义错误的是 ①②③④⑤⑥

- ① 加运算两个操作数的类型不匹配  
 ② 函数调用的实参类型与该函数声明的形参类型不匹配  
 ③ 变量被引用之前未定义  
 ④ 数组使用越界  
 ⑤ 变量被重复定义  
 ⑥ 空指针异常

- A. ①②③④⑤ B. ①②③⑤ C. ①②③④⑤⑥ D. ①③⑤

5. 对于下述文法G[S'] (包含r0-r4等5个产生式规则)和识别其活前缀项目集合的DFA, 关于LR(0)和SLR(1)的说法, 描述错误的是 A .

- r0:  $S' \rightarrow S$  r1:  $S \rightarrow I$  r2:  $S \rightarrow \text{other}$   
 r3:  $I \rightarrow \text{if } S$  r4:  $I \rightarrow \text{if } S \text{ else } S$



$\text{Follow}(I)$   
 =  $\text{Follow}(S)$   
 =  $\{ \$, \text{if}, \text{else} \}$

- A. 如果构造SLR(1)分析表, 则以状态5和if作索引的元素位置填r3  
 B. 如果构造LR(0)分析表, 则以状态5和if作索引的元素位置填r3  
 C. SLR(1)文法与LR(0)文法识别其活前缀项目集合的DFA是一样的, 区别在于它们构造分析表的规则不一样  
 D. 如果构造SLR(1)分析表, 则以状态7和else作索引的元素位置填r4

二. (24 分) 论述题 构造上下文无关文法 (1/2分)

1) (共7分) 描述确定性有穷自动机的定义 (4分), 并以实际需求举例说明自动机可以解决的问题 (3分)。

2) (共9分) 对于上下文无关文法  $G[lexp]$ , 其中,  $r1$  至  $r7$  是产生式标号。

$r1: lexp \rightarrow \text{number}$

$r2: lexp \rightarrow (op\ lexp\ seq)$

$r3: op \rightarrow +$

$r4: op \rightarrow -$

$r5: op \rightarrow *$

$r6: lexp\ seq \rightarrow lexp\ seq\ lexp$

$r7: lexp\ seq \rightarrow lexp$

(1) 以该文法为例, 阐述构成上下文无关文法的核心要素 (4分)

(2) 描述文法  $G[lexp]$  定义的语言具有什么特征 (3分); number 是终结符, +, -, \* 是非终结符

(3) 阐述上下文无关文法在语法分析中的作用 (2分)

3) (共8分)

(1) 为以下C程序的运行时环境画出存储组织结构图(4分)。

```
int a[4]={1, 2, 3, 4};
```

```
int f(int i, int b[])
```

```
{ int j=i;
```

```
  A:{ int i=j;
```

```
      char c = b[i];
```

```
      ...
```

```
    }
```

```
    return 0;
```

```
}
```

```
main ()
```

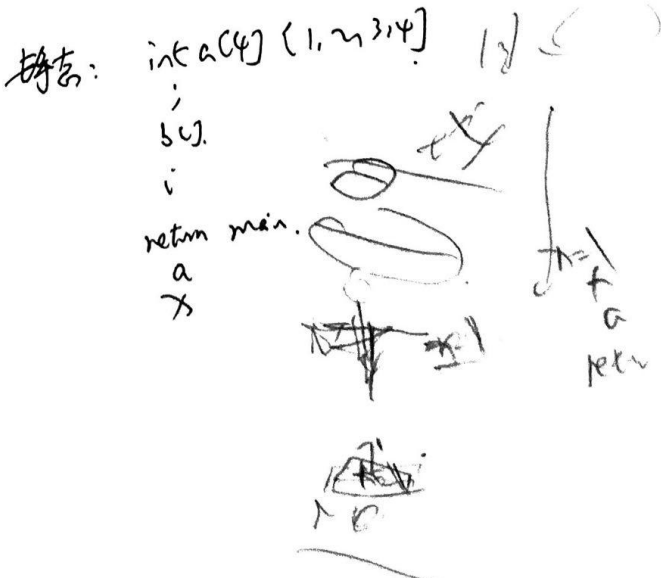
```
{ int x=1;
```

```
  x = f(x,a);
```

```
  return 0;
```

```
}
```

int x=1  
x=f(x,a)



- (2) 某学生在调试程序发现问题: newTreeNode 执行完毕后返回节点 t, 但是节点 t 中 type 成员的值不符合预期。请根据 C 语言运行时内存组织, 分析下列程序中存在的问题 (4 分)。

```

.....
struct treeNode *newTreeNode(NodeType type) {
    struct treeNode t;
    t.type = type;
    t.children[0] = NULL;
    t.children[1] = NULL;
    t.children[2] = NULL;
    t.rowNum = 0;
    for(int i = 0; i < 30; i++){
        t.content[i] = '\0';
    }
    t.sibling = NULL;
    return &t;
}
.....

```



好是在函数内部

定义 t.

所以在函数外部使用,

会报错, 然后串错

为访问.

因此, return 也是返回 &t.

t 地址, 但对方为访问

已释放.

### 三. 计算题 (15 分):

构造文法  $G[S]$  ( $r1$  至  $r5$  是产生式编号) 的 LR(1) 项目集的 DFA, 如下图所示:

$r1: S \rightarrow id$

$r2: S \rightarrow V := E$

$r3: V \rightarrow id$

$r4: E \rightarrow V$

$r5: E \rightarrow n$

请按照下述格式, 撰写用 LR(1) 分析算法对输入串 id := id 的分析步骤:

步骤	状态栈 S	符号栈 X	输入符号	动作
----	-------	-------	------	----

1

#

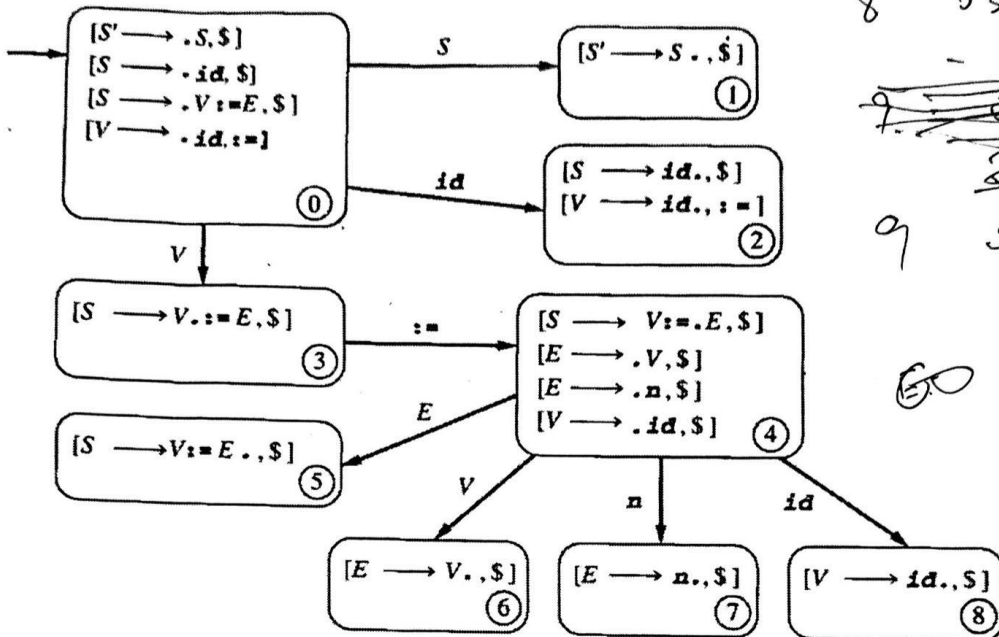
id

共 7 页

第 4 页

5 0348 # V := id # V3 goto 6.  
 6. 0346 # V := V. # V4 goto 5.  
 7. 0345 # V := E, # V2 goto 4.  
 8. 0344 # V := S. # V1 goto 3.

西北工业大学命题专用纸



四. (12分) 设有如下文法  $G[L]$ , 其中,  $r1$  至  $r5$  是产生式编号。

$r1: L \rightarrow \text{num}$

$r2: L \rightarrow \text{id}$

$r3: L \rightarrow (S)$

$r4: S \rightarrow SL$

$r5: S \rightarrow L$

- (3分) 将左递归文法  $G[L]$  改造成与之等价的  $LL(1)$  文法。
- (5分) 对于步骤1) 得到的文法, 构造其  $LL(1)$  分析表。
- (4分) 请写出  $LL(1)$  算法对输入串  $(id$  的分析步骤。

中改文法:

$L \rightarrow \text{num} \rightarrow \text{First}^+ \rightarrow \text{num}$   
 $L \rightarrow \text{id} \rightarrow \text{id.}$   
 $L \rightarrow (S) \rightarrow ($   
 $S \rightarrow LS' \rightarrow L.$   
 $S' \rightarrow LS' \rightarrow L.$

$\text{num id ( ) \#}$   
 $L \rightarrow \text{num} \rightarrow \text{id} \rightarrow (S)$   
 $S \rightarrow LS' \rightarrow LS' \rightarrow LS'$   
 $S' \rightarrow LS' \rightarrow LS' \rightarrow LS'$   
 $S' \rightarrow L.$

教务处印制

共7页

第5页

$S' \rightarrow L$

$\rightarrow \text{Follow}(S) = )$

388.

分析表

输入串

所用

1.

#E

(id #

E → L

## 五. (34 分) 语法、语义、中间代码生成的综合题目。

对于表达式的文法  $G[\text{program}]$ , 请分别完成 (一) 至 (三) 题:

$r1: \text{program} \rightarrow \text{var-decls}; \text{stmts}$

$r2: \text{var-decls} \rightarrow \text{var-decls}; \text{var-decl}$

$r3: \text{var-decls} \rightarrow \text{var-decl}$

$r4: \text{var-decl} \rightarrow \text{id} : \text{type-exp}$

$r5: \text{type-exp} \rightarrow \text{int}$

$r6: \text{type-exp} \rightarrow \text{bool}$

$r7: \text{stmts} \rightarrow \text{stmts}; \text{stmt}$

$r8: \text{stmts} \rightarrow \text{stmt}$

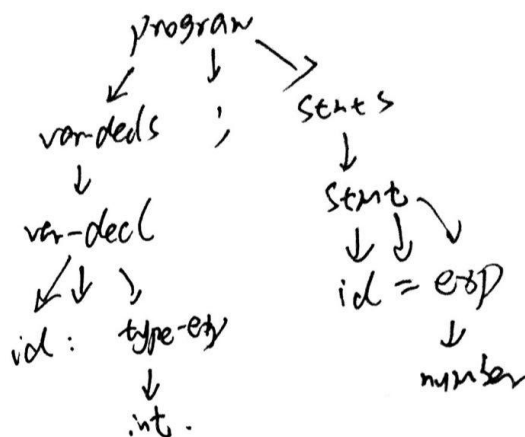
$r9: \text{stmt} \rightarrow \text{if-exp then stmt}$

$r10: \text{exp} \rightarrow \text{true}$

$r11: \text{exp} \rightarrow \text{false}$

$r12: \text{exp} \rightarrow \text{number}$

$r13: \text{stmt} \rightarrow \text{id} = \text{exp}$



## (一) 推导和抽象语法树部分(12分)

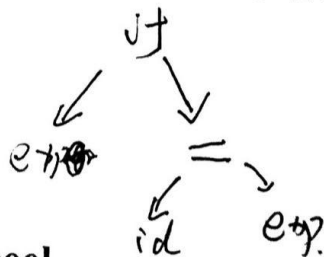
- (8分) 请为句子 **id : int; id = number** 写出其最左推导过程, 并绘制最左推导过程对应的分析树;
- (4分) 请设计上述文法中两类语句 (即第9个产生式  $r9$  对应的if语句和第13个产生式  $r13$  对应的赋值语句) 对应的抽象语法树结构, 给出设计的可视化图示。

## (二) 属性文法部分(10分)

已知:

- **number**的数据类型是**int**
- **false**或**true**对应的数据类型是**bool**
- **exp**的数据类型是**int**或**bool**
- **id**对应的数据类型可以调用函数 $\text{lookup}(\text{id.name})$ 进行查询, 该数据返回**int**或**bool**;
- 辅助函数  $\text{Boolean typeEqual}(\text{Type } t1, \text{Type } t2)$ 接受两个类型 $t1, t2$ , 会根据语言的类型等价规则判断 $t1$ 和 $t2$ 是否是相同的类型, 类型相同则返回**true**, 否则返回**false**。

具体要求:



西北工业大学命题专用纸

撰写文法G[program]的属性文法，属性文法的作用：

- 为r4-r6写属性等式，计算id的数据类型Type，通过函数insert(id.name)插入符号表；
- 为r13写属性等式，确保赋值语句的左右操作数的类型一致。
- 为r9写属性等式，确保if条件表达式类型是bool类型。

(三) 中间代码生成部分(12分)

已知：

- 赋值翻译的中间代码格式为：  $x=y$
- if 翻译的中间代码格式为：

```

<code to evaluate E to t1> //求逻辑表达式E的值的三地址码序列将逻辑表达式的值赋给临时变量t1
if_false t1 goto L1 //如果t1是false，则跳转到标记 L1
<code for S1> // true情况下执行的三地址码序列
label L1
    
```

- emit(): 函数emit() 将三地址语句作为函数实参，该函数将其输出到文件中；
- Newtemp(): 函数Newtemp()产生一个临时名字，例如，第一次调用产生临时变量名t1，第二次调用产生临时变量名t2，等等。

具体要求：

采用C语言（或java或C++）定义（一）中规定的抽象语法树的节点STreeNode，并撰写文法G[program]的中间代码（三地址码）生成的递归函数，递归遍历产生式规则r9和r13对应的抽象语法树，其他产生式对应的抽象语法节不予考虑。

```

GenCode(STreeNode * t){
    
```

```

}
    
```