



北京航空航天大学

2022-2023 学年 第一学期期末

《编译技术》

考试 A 卷

班 级 \_\_\_\_\_ 学 号 \_\_\_\_\_

姓 名 \_\_\_\_\_ 成 绩 \_\_\_\_\_

考试地点 \_\_\_\_\_

2022 年 12 月 8 日



## 北京航空航天大学线上考试诚信承诺书

为严肃考风考纪，维护考试良好秩序，营造公平、公正的考试环境，本次考试将使用信息技术手段对考试过程和试卷实施监控和管理，请各位同学树立诚信考试观念，考前做出如下承诺：

1. 严格遵守学校考试有关规定和纪律要求，自觉维护考试秩序，认真履行考试程序，坚决服从监考老师和工作人员管理；
2. 不提供、不传递、不抄袭与考试课程内容有关的信息资料，不抄袭他人答卷或协助他人抄袭；
3. 不请他人代考或为他人代考、组织作弊、使用通讯设备和工具作弊；
4. 如有考试违规行为，自愿按有关规定接受处理。

**请同学们考前在答卷上书写以下文字：**

本人已知悉并将遵守《北京航空航天大学线上考试诚信承诺书》相关内容。



班号\_\_\_\_\_ 学号\_\_\_\_\_ 姓名\_\_\_\_\_ 成绩\_\_\_\_\_

## 《 编译技术 》期末考试卷

题目：

- 一、填空题..... ( 20 分)
- 二、正则文法与自动机..... ( 12 分)
- 三、符号表构造与运行时存储管理..... ( 15 分)
- 四、LL (1) 分析法 ..... ( 8 分)
- 五、算符优先分析法 ..... ( 10 分)
- 六、SLR 分析法..... ( 15 分)
- 七、代码优化..... ( 20 分)

题号	得分	教师签字
1		
2		
3		
4		
5		
6		
7		
总分		



注:

1. 试卷共 10 页 (含封面和目录), 请仔细检查;
2. 在监考老师统计完试卷后, 再离开考场;
3. 所有答案写在答题纸上, 每页答题纸顶部都需要写明 “考场考号-学号-姓名-页号”, 答题结束后, 请在答题纸首页写上总页数。

一. 填空题 (共 20 分, 每空 1 分)

1. 编译过程的\_\_\_\_\_阶段, 其理论基础主要与 3 型文法和有穷自动机相关;  
\_\_\_\_\_阶段, 其理论基础与 3 型文法关系较弱但主要与上下文无关文法相关。
2. 按照编译错误的类型分类, C 语言程序在编译过程中, 如果发现一个变量未定义就被使用, 这属于\_\_\_\_\_错误; 如果发现一个函数缺乏最后的右侧大括号, 这属于\_\_\_\_\_错误。
3.  $S=\{0,1,2,3\}$  为符号串集合,  $S^*$  中长度最短的符号串为\_\_\_\_\_,  $S^+$  中长度最短的符号串为 (需穷举) \_\_\_\_\_。
4. A、B 均为符号串集合, 其中  $A = \{a, b\}$ ,  $B = \{\epsilon, 0, 1\}$ , AB 中长度最短的 2 个符号串为\_\_\_\_\_和\_\_\_\_\_。
5. 有文法  $G[E]: E ::= 0E0|1E1|1$ , 句型 10E01 的短语有\_\_\_\_\_, 其中简单短语为\_\_\_\_\_, 句柄为\_\_\_\_\_。
6. 上一题的文法, 如改写为翻译文法  $G[E]: E ::= 0E0@a|1E1@b|1$  后, 当其输入序列为 10101, 其活动序列为\_\_\_\_\_, 翻译输出的字符串为\_\_\_\_\_。
7. 左递归文法  $G[S]: S ::= Sab|Sba|aa$ , 采用扩充的 BNF 表示法消除左递归后得到的文法为\_\_\_\_\_。
8. 语句  $X = A+B*C+D$  的波兰后缀表示为\_\_\_\_\_, 四元式表示为\_\_\_\_\_。
9. 寄存器分配时, 编译器通常会把通用寄存器分为\_\_\_\_\_寄存器和\_\_\_\_\_寄存器。
10. 大多数微处理器体系结构上, 为当前函数申请活动记录空间都通过操作\_\_\_\_\_寄存器来完成, 该空间通常位于 (高或低) \_\_\_\_\_地址段中。

二. 设有非确定的有限自动机  $M'$  如图所示 (12 分)

1. 请将其确定化, 并画出其新的状态图。(6 分)
2. 请将其最小化。(6 分)

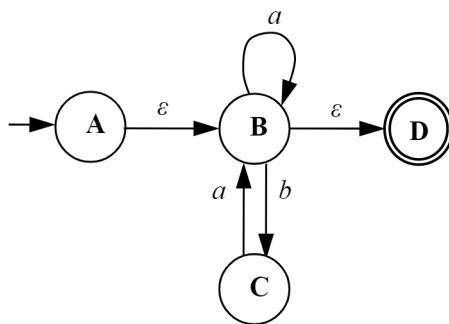


图 NFA



三. 有如下程序段, (共 15 分)

1. 说明符号表的内容和分程序索引表的作用。(2 分)

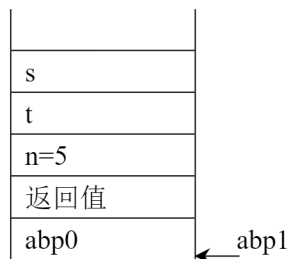
2. 说明动态存储分配和静态存储分配的区别。(3 分)

3. 下图是递归计算 Fibonacci 数列的 C 语言代码, (10 分, 每个问题 5 分)

```
int f(int n){  
    int t,s;  
    if(n<2) return 1;  
    s=f(n-1);  
    t=f(n-2);  
    return s+t;  
}
```

假设初始调用是 f(5), 调用 f(5)时, 运行栈的内容如下:

上一页



(1) 请画出当第一个 f(3)调用即将返回时运行栈内容 (不考虑 Display 区,只有形式参数给出值)

(2) 请画出当第二个 f(3)调用即将返回时运行栈内容 (不考虑 Display 区,只有形式参数给出值)



四. 有如下文法  $G[S]$ : (共 8 分)

$$S \rightarrow cAtSB|a$$

$$B \rightarrow eA|\varepsilon$$

$$A \rightarrow b$$

1. 计算每个产生式右端符号串的 FIRST 集和每个非终结符的 FOLLOW 集 (用#代表输入结束)。(4 分)

2. 说明该文法是否为 LL(1) 文法, 并给出依据。(4 分)



五. 有如下文法  $G[E]$ : (共 10 分)

$$E \rightarrow a \mid b \mid (A)$$

$$A \rightarrow EdA \mid E$$

1. 求各非终结符的 FIRSTVT 和 LASTVT 集合。(4 分)

2. 构造文法  $G$  的优先关系矩阵, 并判断该文法是否是算符优先文法。(6 分)

右终结符 (栈外) 左终结符 (栈内)	a	b	(	)	d	#
a						
b						
(						
)						
d						
#						





六. 有如下文法  $G[S]$ : (共 15 分)

$$S \rightarrow S0S0 \mid S1S1 \mid *$$

1. 拓广文法, 并求拓广后的文法中所有非终结符的 FIRST 和 FOLLOW 集合。(5 分)

2. 求 LR(0)项目集规范族  $\{I_0, I_1, \dots\}$ , 并分别给出能够识别活前缀  $S0S$  和  $S0S0$  的有效项目集。(5 分)

3. 按照以下格式构造 Action 表和 Goto 表: (5 分)

	0	1	*	#	S
$I_0$ :				放大	

七. 有如下程序, 其中  $x$ 、 $y$  是  $\text{int}$  类型的函数形参, 函数返回值是  $\text{int}$  类型。其中,  $a$  是全局变量,  $i$ 、 $j$ 、 $r$  是局部变量,  $t1 \sim t7$  是由中间代码生成阶段产生的临时变量。(共 20 分)

```

        j=0                ①
        r=0
_L1:    i=0
        if i<8 goto _L3
_L2:    t1=x+i
        t2=y+i
        t3=4*t1
        t4=a+t3
        t5=val(t4)         ②
        t3=4*t2
        t4=a+t4
        t6=val(t4)
        t7=t5+t6
_L3:    if x=0 goto _L4
        write t7
        goto _L5
_L4:    write i,j
        r=r+t7
        i=i+1
        if i<8 goto _L2
_L5:    j=j+1
        if j<=3 goto _L1
        return r

```

1. 本程序段切分基本块后, 以①所在的基本块命名为  $B1$ , 按程序出现先后顺序分别命名各个基本块为  $B2$ 、 $B3 \dots$ , 请画出流图。(4 分)
2. 对②所在的基本块用 DAG 做公共子表达式删除, 请绘制 DAG 图并给出消除基本块后优化后的代码。其中,  $\text{val}$  是一个单目运算符。(4 分)
3. 对整个函数做活跃变量分析, 写出每个基本块的  $\text{def}$  和  $\text{use}$  集合, 以及  $\text{in}$  和  $\text{out}$  集合, 根据分析信息对参与全局寄存器分配的变量绘制冲突图。注意: 变量  $A$ 、 $B$  冲突的标准为, 变量  $B$  的定义点处变量  $A$  活跃。(8 分)
4. 假设全局寄存器为  $\{X1, X2, X3, X4\}$ , 试采用图着色方法对 (3) 中参与分配的变量做全局寄存器分配, 写出分析过程, 给出一种可行的寄存器分配方案。(4 分)