### 计算机学院、网安学院2020－2021学年第一学期

### 本科生编译系统原理期末考试试卷(A卷)

专业：\_\_\_\_\_\_\_\_\_\_\_\_\_\_年级：\_\_\_\_\_\_\_\_\_\_\_\_\_\_学号：\_\_\_\_\_\_\_\_\_\_\_\_\_\_

姓名：\_\_\_\_\_\_\_\_\_\_\_\_\_\_成绩：\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |
| --- |
| **得 分** |
|  |

1. 单项选择题（每空2分，共24分）
   * + 1. C++编译器进行常数的类型转换是在 阶段，过滤注释是在  
           阶段，消除公共子表达式是在 阶段。

A．词法分析 B．语法分析

C．语义分析 D．代码优化

* + - 1. GCC-ARM将C++程序转换为ARM机器码程序，模拟器在X86平台上模拟执行ARM机器码程序是采用读取一条指令——转换为相应的X86机器码——执行，如此往复的方式，则GCC-ARM是一种\_\_\_\_\_\_\_\_\_\_，模拟器是一种\_\_\_\_\_\_\_\_\_\_。

A．预处理器 B．编译器

C．链接器 D．解释器

* + - 1. 在词法分析和语法分析中，****为\_\_\_\_\_\_\_\_\_\_。

A．终结符

B．非终结符

C．空符号

D．空符号串

* + - 1. 正则表达式间的“**|**”运算支持交换律，这是因为正则表达式本质上描述的是\_\_\_\_\_\_\_\_\_\_\_\_。

A．符号 B．符号集合

C．符号串 D．符号串集合

* + - 1. 下列语言可以用DFA识别的是\_\_\_\_\_\_\_\_\_\_，可用CFG识别的是\_\_\_\_\_\_\_\_\_\_。

A．形如**xx**的0、1串集合

B．形如**anbncn**（n≥1）的串的集合

C．正则表达式集合

D．SLR(1)文法的活前缀集合

* + - 1. 对下面CFG，说法**错误的**是\_\_\_\_\_\_\_\_\_\_。

**S → 0X X → Y1 Y → Y0 | 0 Z → X1**

A．**Z**是无用的 B．与**00+1**对应相同的语言

C．符合算符文法定义 D．**001**是其活前缀

* + - 1. 如果将物种视为类型，且对类型采用**名字等价**判定，则“披着羊皮的灰太狼”会被认为\_\_\_\_\_\_\_\_\_。

A．是类型“羊”

B．是类型“狼”

C．是类型“灰太狼”

D．可转换为类型“红太狼”

* + - 1. L-属性定义的翻译更容易和\_\_\_\_\_\_\_\_\_\_相结合。

A．预测分析 B．算符优先分析

C．SLR分析 D．规范LR分析

|  |
| --- |
| **得 分** |
|  |

1. 设计题（每题6分，共24分）

1. 描述下面正则表达式接受什么符号串集合（注意：转义符\表示后面的**"**应视为普通字符，其他符号均为正则表达式运算符）。

**\"([^\"])\*\"**

2. 设计接受IPv4地址的正则表达式（可用正则定义）。

3. 设计接受语言{a*i*b*j*a*k*b*l* | *i*+*j*=*k+l*, *i*, *j*, *k*, *l*>=0}的上下文无关文法。

4. 设计接受C++数组声明语句的上下文无关文法，其中数组元素类型限定为int、char及它们的指针，数组维数可以是任意维。

|  |
| --- |
| **得 分** |
|  |

1. （22分）对下面的正则表达式。

**(0 | 1)\*110(0 | 1)\***

1. 用**Thompson构造法**将其转换为NFA，识别0111010。（8分）

2. 用**子集构造法**将得到的NFA转换为DFA，画出最终的状态转换图，识别0111010。（10分）

3. 将DFA最小化，画出最终的状态转换图。（4分）

|  |
| --- |
| **得 分** |
|  |

1. （15分）对下面文法：

**S→Ac | Bc**

**A→Aa | **

**B→Bb | **

1. 指出其终结符集合、非终结符集合、开始符号（3分）

2. 消除文法左递归（4分）

3. 构造预测分析表，对句子aac进行分析（8分）。

|  |
| --- |
| **得 分** |
|  |

1. （5分）对下面流图，指出所有回边及每条回边对应的循环包含哪些顶点。



|  |
| --- |
| **得 分** |
|  |

1. （10分）下面文法描述了类型表达式，设计语法制导定义实现构造类型表达式对应的表达式树。使用属性p保存每个语法符号对应的表达式树的根节点指针，假设已有辅助函数mkleaf(basic\_type)（及mkleaf(num, val)）和mknode(op, child1, child2)分别为基本类型（及数值）和构造类型创建叶节点和内部节点，直接使用即可。

T → integer | char | real | void | array(num, T) | T '╳' T | pointer(T) | T '→' T