

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

一、 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___C___阶段,过滤注释是在___A___阶段,消除公共子表达式是在___D___阶段。
A. 词法分析 B. 语法分析
C. 语义分析 D. 代码优化
2. 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___B___,模拟器是一种___D___。
A. 预处理器 B. 编译器
C. 链接器 D. 解释器
3. 在词法分析和语法分析中, ϵ 为___D___。
A. 终结符
B. 非终结符
C. 空符号
D. 空符号串
4. 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___D___。
A. 符号 B. 符号集合
C. 符号串 D. 符号串集合
5. 下列语言可以用 DFA 识别的是___D___,可用 CFG 识别的是___C___。
A. 形如 xx 的 0、1 串集合
B. 形如 $a^n b^n c^n$ ($n \geq 1$) 的串的集合
C. 正则表达式集合
D. SLR(1)文法的活前缀集合

6. 对下面 CFG, 说法**错误**的是___D___。

$S \rightarrow 0X \quad X \rightarrow Y1 \quad Y \rightarrow Y0 \mid 0 \quad Z \rightarrow X1$

- 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

7. 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___。

- 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”

8. L-属性定义的翻译更容易和___A___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

二、设计题 (每题 6 分, 共 24 分)

得 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)。
 $\backslash"([^\backslash"])*\backslash"$

答: 接受双引号包围的字符串, 字符串中不包含双引号。

2. 设计接受 IPv4 地址的正则表达式 (可用正则定义)。

答:

$D \rightarrow [0-9]$

$T \rightarrow [1-9][0-9]$

$H \rightarrow 1[0-9][0-9]$

$B \rightarrow 2([0-4][0-9] \mid 5[0-5])$

$S \rightarrow D \mid T \mid H \mid B$

$IPAddr \rightarrow S \backslash . S \backslash . S \backslash . S$

3. 设计接受语言 $\{a^i b^j a^k b^l \mid i+j=k+l, i, j, k, l \geq 0\}$ 的上下文无关文法。

答:

$S \rightarrow aSb \mid A \mid B \mid M$

$A \rightarrow aAa \mid M$

$B \rightarrow bBb \mid M$

$M \rightarrow bMa \mid \varepsilon$

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。

答：

$D \rightarrow T \text{ id } M ;$

$T \rightarrow \text{int} \mid \text{char} \mid T^*$

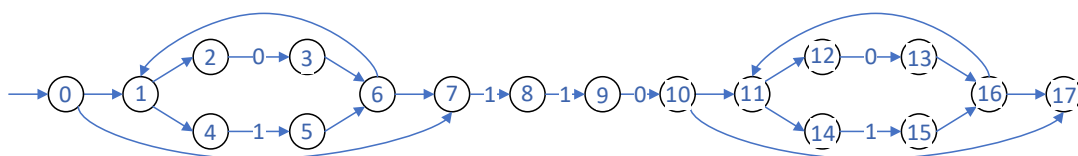
$M \rightarrow M [\text{num}] \mid [\text{num}]$

得 分

三、（22 分）对下面的正则表达式。

$(0 \mid 1)^* 110(0 \mid 1)^*$

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



识别 0111010 过程：

0→1→2→3→6→1→4→5→6→7→8→9→10→11→14→15→16→11
→12→13→16→17

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）

$\epsilon_closure(\{0\}) = \{0, 1, 2, 4, 7\} = A$

$\delta(A, 0) = \epsilon_closure(\delta(\{0, 1, 2, 4, 7\}, 0)) = \{1, 2, 3, 4, 6, 7\} = B$

$\delta(A, 1) = \epsilon_closure(\delta(\{0, 1, 2, 4, 7\}, 1)) = \{1, 2, 4, 5, 6, 7, 8\} = C$

$\delta(B, 0) = \epsilon_closure(\delta(\{1, 2, 3, 4, 6, 7\}, 0)) = \{1, 2, 3, 4, 6, 7\} = B$

$\delta(B, 1) = \epsilon_closure(\delta(\{1, 2, 3, 4, 6, 7\}, 1)) = \{1, 2, 4, 5, 6, 7, 8\} = C$

$\delta(C, 0) = \epsilon_closure(\delta(\{1, 2, 4, 5, 6, 7, 8\}, 0)) = \{1, 2, 3, 4, 6, 7\} = B$

$\delta(C, 1) = \epsilon_closure(\delta(\{1, 2, 4, 5, 6, 7, 8\}, 1)) = \{1, 2, 4, 5, 6, 7, 8, 9\} = D$

$\delta(D, 0) = \epsilon_closure(\delta(\{1, 2, 4, 5, 6, 7, 8, 9\}, 0)) = \{1, 2, 3, 4, 6, 7, 10, 11, 12, 14, 17\} = E$

$\delta(D, 1) = \epsilon_closure(\delta(\{1, 2, 4, 5, 6, 7, 8, 9\}, 1)) = \{1, 2, 4, 5, 6, 7, 8, 9\} = D$

$\delta(E, 0) = \epsilon_closure(\delta(\{1, 2, 3, 4, 6, 7, 10, 11, 12, 14, 17\}, 0)) = \{1, 2, 3, 4, 6, 7, 11, 12, 13, 14, 16, 17\} = F$

$\delta(E, 1) = \epsilon_closure(\delta(\{1, 2, 3, 4, 6, 7, 10, 11, 12, 14, 17\}, 1)) = \{1, 2, 4, 5, 6, 7, 8, 11, 12, 14, 15, 16, 17\} = G$

$\delta(F, 0) = \epsilon_closure(\delta(\{1, 2, 3, 4, 6, 7, 11, 12, 13, 14, 16, 17\}, 0)) = F$

$\delta(F, 1) = \epsilon_closure(\delta(\{1, 2, 3, 4, 6, 7, 11, 12, 13, 14, 16, 17\}, 1)) = G$

$\delta(G, 0) = \epsilon_closure(\delta(\{1, 2, 4, 5, 6, 7, 8, 11, 12, 14, 15, 16, 17\}, 0)) = F$

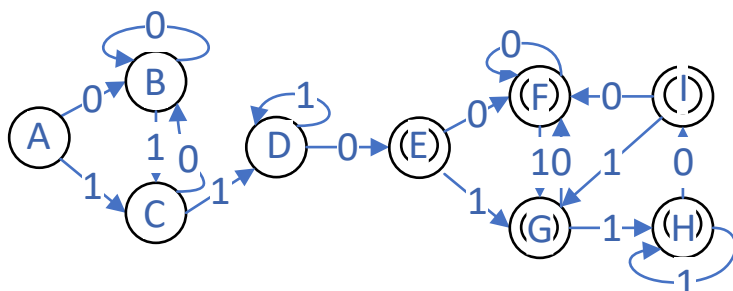
$\delta(G, 1) = \epsilon_closure(\delta(\{1, 2, 4, 5, 6, 7, 8, 11, 12, 14, 15, 16, 17\}, 1)) = \{1, 2, 4, 5, 6, 7, 8, 9, 11, 12, 14, 15, 16, 17\} = H$

$\delta(H, 0) = \varepsilon_closure(\delta(\{1, 2, 4, 5, 6, 7, 8, 9, 11, 12, 14, 15, 16, 17\}, 0)) = \{1, 2, 3, 4, 6, 7, 10, 11, 12, 13, 14, 16, 17\} = I$

$\delta(H, 1) = \varepsilon_closure(\delta(\{1, 2, 4, 5, 6, 7, 8, 9, 11, 12, 14, 15, 16, 17\}, 1)) = H$

$\delta(I, 0) = \varepsilon_closure(\delta(\{1, 2, 3, 4, 6, 7, 10, 11, 12, 13, 14, 16, 17\}, 0)) = F$

$\delta(I, 1) = \varepsilon_closure(\delta(\{1, 2, 3, 4, 6, 7, 10, 11, 12, 13, 14, 16, 17\}, 1)) = G$

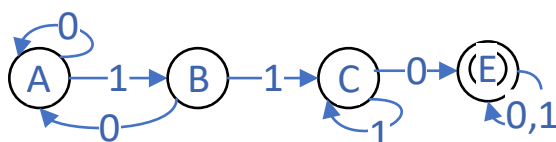


识别 0111010 过程: $A \rightarrow B \rightarrow C \rightarrow D \rightarrow D \rightarrow E \rightarrow G \rightarrow F$

3. 将 DFA 最小化, 画出最终的状态转换图。(4 分)

初始化 $\{A, B, C, D\}$, $\{E, F, G, H, I\}$, 后者不可再分

0 将前者分裂为 $\{A, B, C\}$ 和 $\{D\}$, 1 将前者分裂为 $\{A, B\}$ 和 $\{C\}$, 至此不可再分



四、 (15 分) 对下面文法:

得 分	$S \rightarrow Ac \mid Bc$
	$A \rightarrow Aa \mid \varepsilon$
	$B \rightarrow Bb \mid \varepsilon$

1. 指出其终结符集合、非终结符集合、开始符号 (3 分)

答:

终结符集合: a, b, c

非终结符集合: S, A, B

开始符号: S

2. 消除文法左递归 (4 分)

$S \rightarrow Ac \mid Bc$

$A \rightarrow aA \mid \varepsilon$

$B \rightarrow bB \mid \varepsilon$

3. 构造预测分析表，对句子 aac 进行分析（8 分）。

答：

FIRST(S)={a, b, c}

FIRST(A)={a, ϵ }

FIRST(B)={b, ϵ }

FOLLOW(S)={\$}

FOLLOW(A)={c}

FOLLOW(B)={c}

预测分析表：

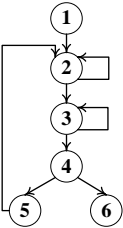
	a	b	c	\$
S	$S \rightarrow Ac$	$S \rightarrow Bc$		
A	$A \rightarrow aA$		$A \rightarrow \epsilon$	
B		$B \rightarrow bB$	$B \rightarrow \epsilon$	

分析过程：

栈	输入缓冲	动作
$\$S$	$aac\$$	$S \rightarrow Ac$
$\$cA$	$aac\$$	$A \rightarrow aA$
$\$cAa$	$aac\$$	
$\$cA$	$ac\$$	$A \rightarrow aA$
$\$cAa$	$ac\$$	
$\$cA$	$c\$$	$A \rightarrow \epsilon$
$\$c$	$c\$$	
$\$$	$\$$	accept

得分

五、（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。



答：

回边 $2 \rightarrow 2$ ，对应循环包括 2

回边 $3 \rightarrow 3$ ，对应循环包括 3

回边 $5 \rightarrow 2$ ，对应循环包括 2、3、4、5

得 分

六、（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 $mknnode(op, child1, child2)$ 分别为基本类型（及立即数）和构造类型创建叶节点和内部节点，直接使用即可。

$T \rightarrow integer \mid char \mid real \mid void \mid array(num, T) \mid T \times T \mid pointer(T) \mid T \rightarrow T$
答：

$T \rightarrow integer \quad \{ T.p = mkleaf(integer); \}$

$T \rightarrow char \quad \{ T.p = mkleaf(char); \}$

$T \rightarrow real \quad \{ T.p = mkleaf(real); \}$

$T \rightarrow void \quad \{ T.p = mkleaf(void); \}$

$T \rightarrow array(num, T_1) \{ T.p = mknnode(array, mkleaf(num, num.val), T_1.p); \}$

$T \rightarrow T_1 \times T_2 \quad \{ T.p = mknnode(prod, T_1.p, T_2.p); \}$

$T \rightarrow pointer(T_1) \quad \{ T.p = mknnode(pointer, T_1.p, NULL); \}$

$T \rightarrow T_1 \rightarrow T_2 \{ T.p = mknnode(func, T_1.p, T_2.p); \}$