

# 2023春编译原理

## 填空

共25分

1. (4分) 语法分析的输入是 ()、输出是 ()
2. (6分)

```
1 Expr1 -> Expr1 OP1 Expr2 | Expr2
2 Expr2 -> Expr3 OP2 Expr2 | Expr3
```

OP1和OP2中优先级高的是 () ; OP1是 () 结合、OP2是 () 结合

3. (5分)

```
1 Expr -> Des AsOp Expr { print("X ") }
2      | Des { print("0 ") }
3 Des  -> * { print("1 ") } Des { print("2 ") }
4      | & { print("3 ") } Des { print("4 ") }
5      | ++ { print("5 ") } Des { print("6 ") }
6      | Des2 { print("7 ") }
7 Des2 -> Des2 { print("8 ") } ++ { print("9 ") }
8      | Des3 { print("A ") }
9 Des3 -> ID { print("B ") }
10 AsOp -> = { print("C ") }
```

翻译 `x = *y = z++` 的输出 ()

4. (4分) 按行存放的 `B[i][j][k][l]`, `i` 的范围是0~5, `j` 的范围是2~4, `k` 的范围是1~4, `l` 的范围是4~8, 每个元素占8个字节, `B` 的首地址是1000, 则 `B[4][2][3][7]` 的起始位置是第 () 字节
5. (6分) C++中 `static` 变量位于 (); 若 `class *obj = new A()`, `obj->var` 位于 (); 若定义全局变量 `A obj`, `obj.var` 位于 () (填“栈区”、“堆区”、“静态区”)

## 简答

共75分

1. (8分) 用课本的方法将 `b(b|a)*` 转化为NFA, 再用子集构造转化为DFA (画图和转化表, 不必最小化)
2. (4分) 画出接受能被3整除的二进制数的DFA
3. (24分) (注: `x` 是终结符号)

```
1 e -> e + t
2 e -> t
3 t -> x ^ t
4 t -> x
```

1. 给出增广文法、LR(0)项集和自动机
2. 求增广文法中非终结符的FIRST和FOLLOW
3. 是SLR(0)文法吗? 为什么

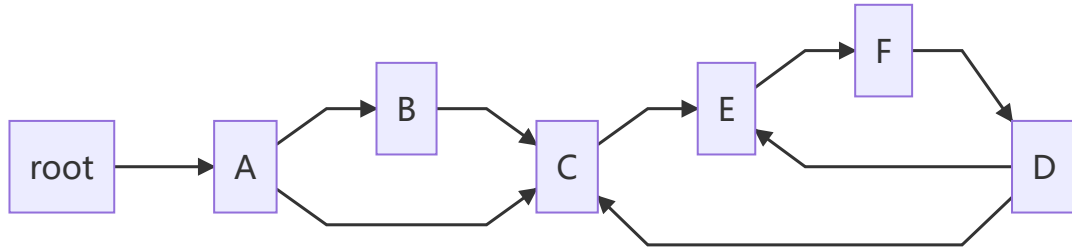
4. (8分)

```

1  B -> B_1 0 { B.val = B_1.val * 2 }
2    | B_1 1 { B.val = B_1.val * 2 + 1 }
3    | 1     { B.val = 1 }

```

1. 画出1001的注释语法树
  2. 消除左递归
  3. 设计消除左递归后的SDT
5. (6分) 使用引用计数



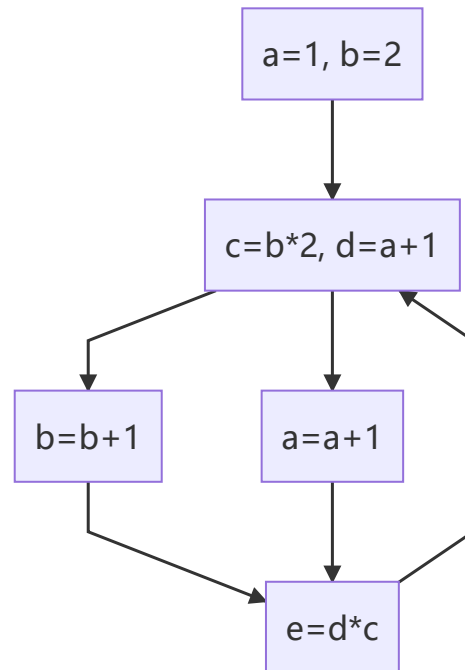
1. 删A, 回收哪些? 其他的计数怎样变化?
  2. 删E->F的指针, 回收哪些? 其他的计数怎样变化?
6. (12分) 3个空闲寄存器, 生成机器代码, 要求画出每一步(机器代码)后的寄存器描述符和地址描述符, a、b、c、x在出口处活跃

```

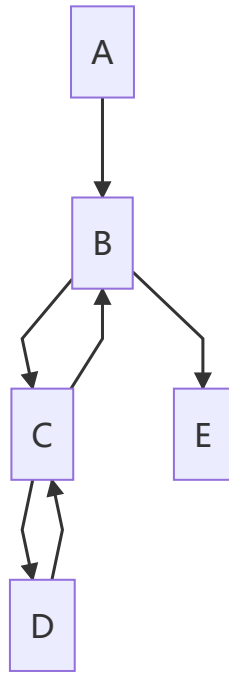
1  t1 = b * c
2  t2 = a + t1
3  x = t2

```

7. (11分) 跑活跃变量, 要求写出每个BB的use和def, IN和OUT



8. (2分) 画支配节点树



## 附加

---

最多拿3分，总分不超过100分

1. (1分) 做游戏时是第 ( ) 章，请了 ( ) 位同学上来
2. (1.5分) 讲了 ( ) 次特别课题，内容有 ( ) 和 ( ) (四选二：编译器测试、机器学习、并发状态记录、嵌入式建模)
3. (2.5分) 总共 ( ) 次实验，有 ( ) 次有选做，讲了 ( ) 次习题课， ( ) 次实验课， ( ) 次两个助教都在