

参考: <https://xjtu.men/t/topic/12940>

一、选择题 (10*2)

- 1、下面哪个不能识别正则语言
- 2、编译的几个步骤, 前后端接口
- 3、一个字母表不能组成哪个字符串
- 4、递归下降分析表的生成
- 5、控制流代码翻译, 考了 **and** (布尔运算)
- 6、PDA 的状态转移 (PDA 有关的选择題不少于 2 道)
- 7、编译器前后端中间组件
- 8、什么不属于中间语言 (选项里有一个 **sw** 命令, 其他是三元、四元)

二、将 NFA 转换为最小 DFA 再转 RE (15)

规范规约

三、词法分析 (5)

已给出一个简单的词法分析 DFA

- (1) 根据输入串写出输出
- (2) 判断事实优先级 (**AAA** 和 **ADD**, 即自加运算和加法运算)

四、文法修剪 (15)

- (1) 修改文法, 消除无效符号
- (2) 修改文法, 消除左递归
- (3) 求某文法的 **first**、**follow** 集并判断其是否为 LL (1) 文法。

五、语法分析 (10)

已知句子 $+2++3*(+4)$ 和对应的语法树 (树已给出)

文法: $E \rightarrow E * E \mid E + E \mid +i \mid (E)$

- (1) 对这个句子进行规范规约; 写出语法树的直接短语、句柄
- (2) 判断文法是否为歧义, 为什么

六、写出 itemDFA 并判断是否存在冲突, 写出所有存在的冲突, 任意消除其中一个。(10)

文法为: $S' \rightarrow E \quad E \rightarrow E + E \mid +E \mid +i$

七、语义分析 (15)

有程序段:

```
int a[2, 3];
void f(int b(), int k){
    if(k > 0 && k < 3) a[k-1, k] = b(k, 1)
    else print 0;
}
```

- (1) 对 **f** 函数声明行进行语义分析 (自然语言或者符号表均可)
- (2) 对 **【if(k > 0 && k < 3) a[k-1, k] = b(k, 1) else print 0;】** 进行分析。写出三地址

/四元式。（未像 2017 题一样给出语义动作）

八、运行时存储空间组织（10）

有程序段：

```
int foo(int y;){
    int z;
    void bar(int x; int soo())
    {
        if(x > 3) bar(x/3, soo());
        else z = soo(x);
        print z;
    }
    int row(int x)
    {
        y = x + 5;
        return y;
    }
    bar(y, raw());
}
foo(6);
```

栈结构：

形参
访问链
控制链
返回地址
变量

写出执行到 `return y;` 时候的栈快照。（未像 2017 题一样给出表格，需要自己画）