

# 编译原理 stage 3 报告

傅子轩 2020010742 计01

## 主要改动

- 1. scanner 中增加了对 continue 的识别
- 2. parser 中增加了对 for, do-while 的语法生成规则
- 3. build\_sym 中访问 for 和 do-while 来设置作用域和生成符号
- 4. type\_check 中使用 loop\_depth 追踪循环深度，以发现在循环体外的 break、continue
- 5. translation 增加对 for, do-while, continue 的翻译，设置当前 continue 应该跳到的标签

## 思考题

### step 7

翻译后的代码

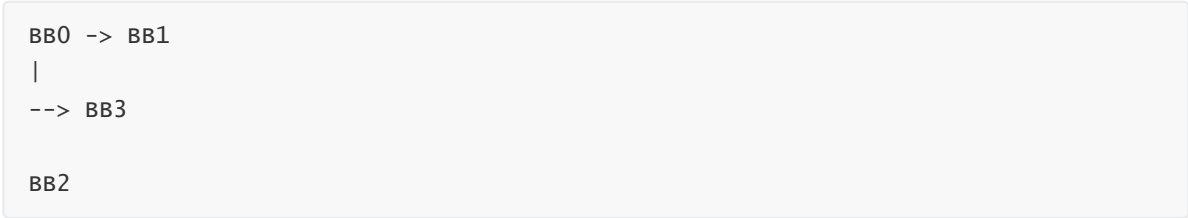
```
_main:
    T1 <- 2
    T0 <- T1
    T2 <- 3
    T3 <- (T0 < T2)
    if (T3 == 0) jump __L1      # Basic Block 0

    T5 <- 3
    T4 <- T5
    return T4                  # Basic Block 1

    return T0                  # Basic Block 2

__L1:
    T6 <- 0
    return T6                  # Basic Block 3
```

控制流图



## step 8

若执行  $n$  次循环，第一种执行指令数  $4n+2$ ，第二种执行指令数  $3n+2$ ，从指令数来说第二种更好。