

编译原理实验三

211220028 党任飞

一、实现功能

- 给符合 c-- 语言词法、语法、语义规范的程序文件输出其中间代码。
- 选做1：额外支持为包含结构体变量的程序生成中间代码。
- 选做2：对包含高维数组的程序报错，且不生成中间代码。

二、运行方式

进入 Lab3/Code/ 目录之后，运行：

```
> make
> ./parser test.cmm out.ir
```

即可分析文件 test.cmm，并将生成的中间代码存于 out.ir 文件中。如果遇到高维数组则同时在控制台输出报错信息。

三、核心实现方法

在上一次实验的基础上，增加 interRepres.c、interRepres.h 两个文件。利用之前建好的语法树和符号表，重新对语法树进行一次遍历，逐步生成中间代码。

基本数据结构设计：

- Operand_：中间代码里的操作数。分为变量VAR、常数CONS、临时变量TEMP、函数FUNC、标签LABEL。里面记录操作数的相关属性和名称，便于分析与打印。
- InterCode_：一行中间代码。分为单操作数、双操作数、三操作数、跳转共四个大类，共19种中间代码。
- InterCodes_：保存中间代码的循环链表。在遍历语法树的代码生成过程中并不打印，而是把所有代码暂时保存在循环链表中，最后统一打印。设置一个哨兵头结点，遍历的时候如果遍历到了头结

点则认为打印完成。

- Operand VAR_Operands[MAXOPS]; : 记录所有出现过的变量，在遇到一个变量的时候查看是否已经生成过，避免重复生成。

四、感想

这次实验思考难度不如上次大，但是工作量好像比上次多，单文件破千行了。