实验报告

171240501 匡舒磊

kuangsl@foxmail.com 基础班

已实现内容

完成了所有必做内容与选做内容,并对中间代码进行了少量的优化。

代码框架

lexical.l : 词法内容识别

syntax.y : 语法内容分析，并建立相应的语法树节点

common.h : 函数定义与相关结构体定义

ast.c : 语法树的相关函数，包括添加节点打印语法树等

main.c: 运行分析

openhash.c 与哈希表操作相关，包括插入删除查询等操作

ir.c 中间代码生成与优化过程

ir_file.c 中间代码输出

optimize.c 代码优化

如何编译

使用助教提供的Makefile进行编译生成parser

实验环境

GNU Linux Realease: Ubuntu 18.04.4 LTS

GCC version 7.5.0

GNU Flex 2.6.4

GNU Bison 3.0.4

实现的重点细节

1. 实现过程

   将语义分析的流程类似，将代码去除掉报错，并加上一些翻译过程即得到了最终代码。代码维护了一个双向链表来存储ir代码。在符号表上还是大部分维持着lab2的实验代码过程。

2. 数组赋值的实现

   考虑到数组赋值的实现，分在两部分进行，一种是类似于 `a=b` 这种，会识别为ID的，另一种情况是高维数组`a[3]=b[3]`的这种情况，在两处实现，但是实现方式类似，均为考虑调用`irExp`后判断类型是否为数组，然后获取两个数组中较小的那个大小，然后按地址逐渐加4，逐渐赋值。前者的代码如下（后者情况类似）：

```
                Operand t1 = irOpTemp();
                  Operand t2 = irOpTemp();
                  Type a = irExp(root->child,t1);
                  Type b = irExp(c1s2(root),t2);
                  int a_size = irTypeSize(a);
                  int b_size = irTypeSize(b);
                  Operand t3 = irOpTemp();
                  for(int i=0;i<a_size&&i<b_size;i+=4){
                      irCodeOp2(CODE_DEREF,t3,t2);
                      irCodeOp2(CODE_DEREF_ASSIGN,t1,t3);
                      irCodeOp3(CODE_ADD,t1,t1,irOpConstant(4));
                      irCodeOp3(CODE_ADD,t2,t2,irOpConstant(4));
                  }
```

实验中遇到的困难/不足

1. 非常遗憾，没有做太多的优化，很多优化是可以做的，但是因为bug太多，加上自己喜欢拖ddl，导致自己没有很好的考虑优化。
2. 数组和结构体的赋值与引用一开始没有思考清楚或者少考虑了一些情况，导致de了很久的bug，希望下次更加细心。