

中间代码生成 实验报告

151220151 殷瀚 411629507@qq.com

1.实现的功能

完成了所有必做及选做任务：

1. 为 cmm 代码生成中间代码；
2. 可以出现结构体类型的变量，结构体变量可作为函数参数；
3. 可以出现高维数组类型的变量，高维数组变量可作为函数参数；

2.实现方法及数据结构

2.1 Operand 和 InterCode

本次实验中最主要的数据结构，Operand（操作数）和 InterCode（中间代码），主要就是根据实验手册中的中间代码语法来定义的。

使用双向链表来存放中间代码，借助 C++ STL 中的 list 来实现。

Operand_类中的静态 int 变量 tmp_num, label_num 用来生成临时变量和 label。

2.2 中间代码生成

将实验三有关的代码写在 InterCodeTranslator 这个类中，与之前的代码分开。深度遍历语法树，生成中间代码并依次添加到链表中，最后输出。

本次实验中最主要的工作量在 Exp 表达式的翻译上，大部分直接参考实验手册即可，自己需要考虑的是基本上只有数组和结构体的翻译。

2.2.1 数组

数组的产生式形如：Exp -> Exp1 LB Exp2 RB.

对于 Exp1，我们需要知道其类型，以确定单个元素的宽度，这可以用实验二中定义的 Type 来表示。对于 Exp2，我们需要知道其运算值，以确定偏移多少个单位宽度，可以用上述的 Operand 来表示。

因此，translate_Exp 函数的返回值应该包含一个 Type 和一个 Operand。由于我们要支持高维数组，因此 Exp1 可能是一个 ID，也可能是某个数组中的高几维，也可能是结构体中的域。同样，Exp2 也可能是一个复杂的表达式。但我们都可以用 translate_Exp 函数来统一进行分析，Exp1 返回的 Operand 就包含了当前的基地址，加上 Exp2 返回的 Operand 乘以 Exp1 返回 Type 的宽度即可得到当前元素的地址。

数组 exp 翻译的最后，是一个解引用，形如 place := *t，来得到目标地址上的值。对于高维数组，我们只有在最低维的翻译结束后，才需要在最后来一句解引用，为了去掉中间冗余的解引用，只要在翻译高维时，让 place 等于 NULL 即可。

比如我们要得到 int a[2][2]中的 a[0][1]这个元素的值，就会依次计算出

```
t1:=&a+0*8;  NULL:=*t1
t2:=t1+1*4;  place:=*t2
```

带有 NULL 的中间代码会在最后被统一删除。

2.2.2 结构体

结构体形如 `Exp -> Exp1 DOT ID`.

翻译起来跟数组类似，还要更简单一些，`translate_Exp(Exp1)`返回的 `Operand` 中包含了当前的基地址，在该结构体对应的符号表表项中寻找 `ID` 的位置，就能知道偏移量，最后加一句解引用即可。

结合上述的数组翻译，就可以支持数组和结构体的复合使用，比如结构体数组，或者结构体的某个域是数组类型。

2.2.3 数组和结构体作为参数

以数组为例，为了让数组类型的参数是传引用而非传值，我们需要在函数调用处传入变量的地址。这样我们在翻译函数体时，对于局部变量中数组和函数参数中的数组就得区别对待。假如数组 `a` 是局部变量，那获取数组起始地址的代码就是 `t:=&a`；如果数组 `a` 是在函数参数中声明的，由于我们传入的就是该数组的地址，那获取数组起始地址的代码就是 `t:=a`。

这样，我们就需要修改语义分析中的代码，对于符号表中的项，记录下它们是否来自函数参数的声明。这样，就可以在生成中间代码时正确处理数组/结构体类型的参数传递。

另外，尽管实验要求中只要求一维数组作为函数参数，不过按照上述实现后，直接就可以支持形如 `int sort(int a[2][2])` 这样的函数。

3. 编译运行方法

直接在 `Code` 目录下 `make` 即可，将依次调用 `flex` 和 `bison` 生成 `.c` 文件，再寻找 `Code` 目录下所有的 `.c` 和 `.cpp` 文件进行编译链接得到可执行文件 `parser` (生成的 `parser` 在 **Code 目录**)。

在 `Code` 目录下执行 `make test` 将调用 `parser`，对 `Test` 目录下所有 `.cmm` 文件进行分析，生成同名的 `.ir` 文件，在项目目录的 `InterCodes` 文件夹中（需要先 `mkdir`）。

在 `Code` 目录下执行 `make clean` 将 `Code` 目录下的临时文件清理，同时删除可执行文件。