**哈尔滨工业大学**

**编译原理 2024春**

**实验三**

|  |  |
| --- | --- |
| 学院： | 计算学部 |
| 姓名： | 徐柯炎 |
| 学号： | 2021110683 |
| 指导教师： | 单丽莉 |

# 实验目的

1. 巩固对中间代码生成的基本功能和原理的认识。
2. 能够基于语法指导翻译的知识进行中间代码生成。
3. 掌握类高级语言中基本语句所对应的语义动作。

# 实验环境

* GNU Linux Release: Ubuntu 12.04, kernel version 3.2.0–29。
* GCC version 4.6.3。
* GNU Flex version 2.5.35。
* GNU Bison version 2.5。

# 实验内容

## 实现功能

在实验一和实验二已经完成词法分析、语法分析和语义分析的基础上，本实验我们将完成中间代码的生成。由于实验一构造了语法树，实验二构建了符号表，我们可以在前两个实验的基础上进行中间代码的生成。中间代码生成的整体流程如下：

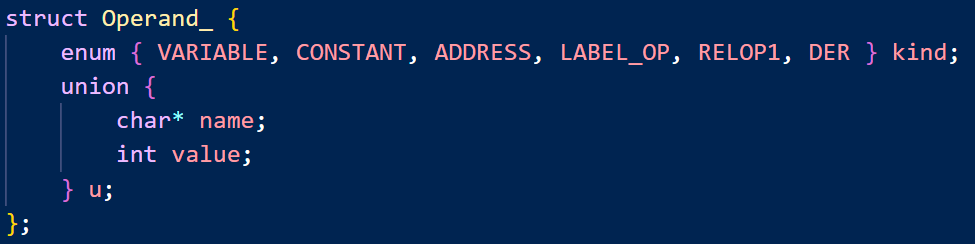
1. 从已经构造完成的语法分析树的根节点开始分析，采用深度优先搜索的方法调用每个根节点类型的translate函数，以此来递归翻译实现的所有中间代码的生成；
2. 将每个translate函数分析出的中间代码存放在InterCodes类型的链表中；
3. 最后调用intergenerate函数将链表中的内容翻译为中间代码，并将中间代码输出到out文件中。

## 数据结构

本次实验主要用到了两个自己定义的结构体，分别是操作数结构体和中间代码结构体。

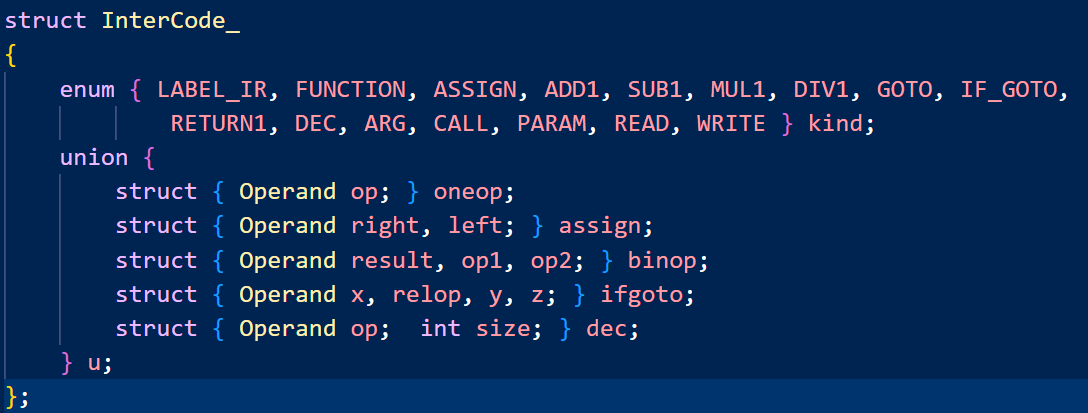
首先是操作数结构体，具体结构如下图所示。这个结构体包括两个部分，kind表示这个操作数的类型，u为此操作数的值，一共有如下几类：

* VARIABLE 表示操作数是一个变量，其值存储在 u.name 字段中。
* CONSTANT 表示操作数是一个常量，其值存储在 u.value 字段中。
* ADDRESS 表示操作数是一个地址，其值存储在 u.name 字段中。
* LABEL\_OP 表示操作数是一个标签，其值存储在 u.name 字段中。
* RELOP1 表示关系操作符，其值存储在 u.value 字段中。
* DER 表示间接寻址操作符，通常用于指针操作，表示操作数是通过另一个地址间接获取的值，其值存储在 u.name 字段中。



接着是中间代码结构体InterCode，同样包含两个部分，kind表示这个操作数的类型，u为此操作数的值，总的来说分为以下几类：

* oneop 用于表示一元操作符，比如一元运算或单个操作数的指令。
* assign 用于表示赋值操作，包括左操作数、右操作数。
* binop 用于表示二元操作符，包括结果操作数、两个操作数。
* ifgoto 用于表示条件跳转指令，包括条件判断的操作数和跳转的目标地址等信息。
* dec 用于表示分配空间的操作，包括操作数和所需的空间大小。



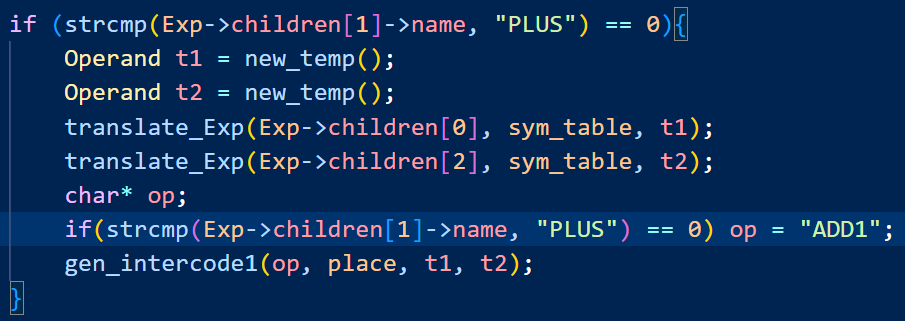
然后在基于InterCode结构体的基础上实现了InterCode双向链表InterCodes，为的是存储遍历和访问InterCode节点方便。

## 翻译过程

以translate\_Exp中的Exp1 PLUS Exp2为例。

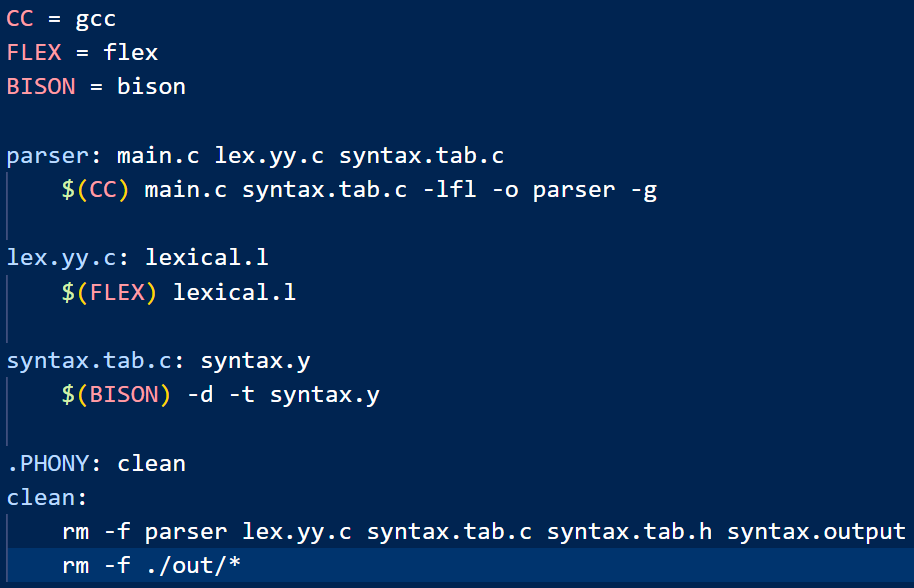
1. 一开始通过translate\_Program函数来递归翻译中间代码，在一级一级的调用中进入translate\_Exp函数，于是比较Exp节点的第二个子节点是否为PLUS，如果是，则进入该分支。
2. 首先通过new\_temp函数生成两个中间变量t1和t2，然后递归调用translate\_Exp函数获取计算t1和t2的中间代码，最后通过gen\_intercode1函数来生成当前节点的中间代码，并将上述中间代码存放到InterCodes双向链表中。
3. 这样我们就完成了这个节点上的中间代码生成。
4. 最后当所有中间代码都存储到InterCodes链表中后，调用intergenerate函数将链表中的内容翻译为中间代码，并将中间代码输出到out文件中。

这样就完成了中间代码的翻译过程。



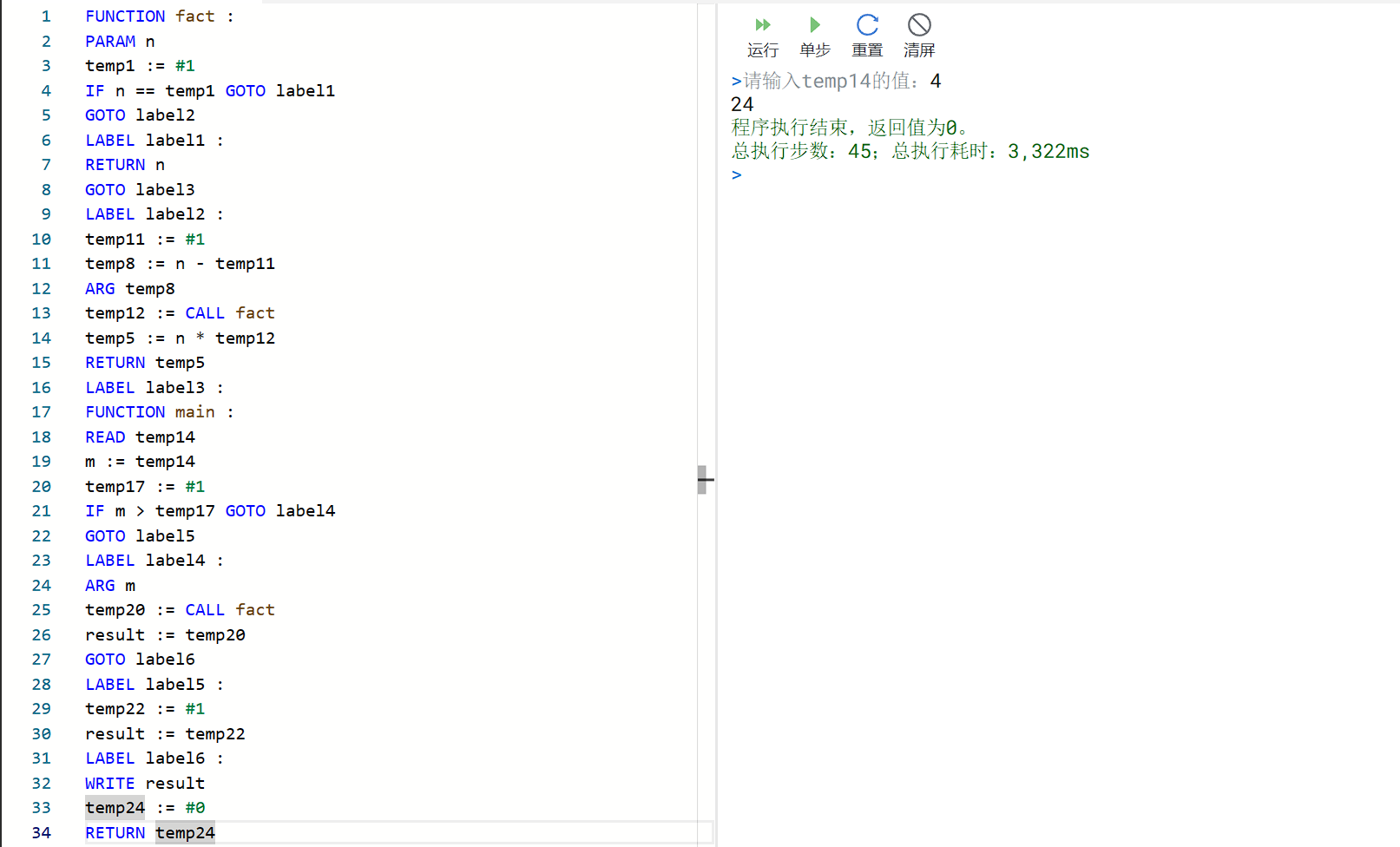
## 编译过程和测试过程

本实验采用makefile进行编译，采用test.sh脚本来测试结果，如下图所示。



# 实验结果

运行test.sh得到实验结果，并通过IR虚拟机来验证结果正确性，如下图所示。



# 实验总结

1. 了解了中间代码生成的全过程，学习了如何生成简单的中间代码。
2. 增强了编程能力、分析问题的能力和动手能力；
3. 对中间代码的生成有了更深层次的认识。