

实验报告

181850236 张子辰 (基础班) 181850236@smail.nju.edu.cn

1. 程序运行和本地环境:

进入"/Lab/Code"目录,直接在终端运行 make 命令,即可编译生成 parser 文件,运行 parser *.cmm *.s 后,即可得到虚拟机使用的 s 文件。

我的本地运行环境为 Ubuntu amd64 18.04, flex 2.6.4, bison 3.0.4, gcc 7.5.0。

2. 程序功能

本次实验中我的主要代码在 mips.c/h 文件中,它依赖于 L3 中实现的中间代码,因此我修改了 L3,将中间代码生成和中间代码输出拆解成了两个函数。

同时我修复了 L3 中无法直接将函数写在函数参数中的问题。

我使用了 LRU 方法进行寄存器分配,我将 8-25 号寄存器作为分配对象,并且在函数调用前后分别进行全部的存储和恢复。

在实验中我发现,必须先将帧指针下移创建空间后才能在这部分空间写入数据,不然是无效的。在最初的实现中,我本来是现将需要存储的寄存器存进去,再根据需要空间的多少来移动 \$sp, 发现这个问题后,一方面我修改成使用 \$fp 访问数据,同时简单地将寄存器全部存储或取出。

因为时间比较紧,而且可能是因为 L3 的实现中使用了太多奇技淫巧,所以这次提交的代码很不完善,它可以处理简单的函数调用,但一旦使用数组就会产生错误。我认为有可能是产生在临时变量的使用上,我对于临时变量在被溢出时应该存放的位置有些疑惑,并且根据临时变量找到数组在栈中的位置比较困难。

本次实验我只通过了讲义中给出的两个样例,对于以前用到的测试库中的样例,因为数组的问题一直没有解决,就先搁置了。