

1.1 目标代码生成

1.1.1. 实验目的

- (1) 了解编译器指令选择和寄存器分配的基本算法;
- (2) 掌握目标代码生成的相关技术和方法, 设计并实现针对 x86/MIPS/RISC-V 的目标代码生成模块;
- (3) 掌握编译器从前端到后端各个模块的工作原理, 目标代码生成模块与其他模块之间的交互过程。

1.1.2. 实验内容

基于 BIT-MiniCC 构建目标代码生成模块, 该模块能够基于中间代码选择合适的目标指令, 进行寄存器分配, 并生成相应平台汇编代码。

如果生成的是 MIPS 或者 RISC-V 汇编, 则要求汇编代码能够在 BIT-MiniCC 集成的 MIPS 或者 RISC-V 模拟器中运行。需要注意的是, config.xml 的最后一个阶段“ncgen”的“skip”属性配置为“false”, "target"属性设置为“mips”、“x86”或者“riscv”中的一个。

如果生成的是 X86 汇编, 则要求使用 X86 汇编器生成 exe 文件并运行。

1.1.3. 实验过程与方法

从 github 重新下载代码, 基于上一次中间代码生成的基础, 完成目标代码生成模块。

虽然指令选择和寄存器分配都存在多种算法, 但是本部分实验最重要的目标是能够生成目标代码, 并在相应的系统平台 (x86) 或者模拟器 (MIPS, RISC-V) 中运行, 走完编译器设计的整个流程。各位同学可以根据中间代码以及前面模块的实现情况, 选择适合于自己, 且能够在时间节点之前完成的算法, 有余力的同学可以选择课堂上讲到的更高级的算法生成更加高效的代码。

由于 RISC-V 模拟器实现的原因, RISC-V 带有乘除法的汇编无法运行。

386 汇编需要配置好 masm32。

对于内置的模块，BIT-MiniCC 提供了 7 个测试程序，**本次实验只要能通过其中任意 2 个测试程序即可拿到本次实验的所有分数**。如有为了完成目标代码生成，对前面模块的修正和提高的内容，也可以适当提高分数。

如果发现代码抄袭，将按照 0 分计算。

关于 MIPS 处理器、RISC-V 处理器及汇编程序编写方法，可以从以下网址下载：

<https://github.com/jiweixing/build-a-compiler-within-30-days>

1.1.4. 实验提交内容

本实验要求提交目标代码生成实现源码（项目 src 下所有文件），以及对应的 config.xml；C/C++ 需提供对应的可执行程序（不需要编译的中间文件），每个人提交一份实验报告。

实验报告放置在 doc 目录下，应包括如下内容：

- 实验目的和内容
- 实现的具体过程和步骤
- 运行效果截图
- 实验心得体会