

《高级程序设计 II》一大实验

第 2 次作业

2020 年 04 月 20 日

- ✧ 本次作业主要目的是让同学们通过编写代码，基于 LLVM IR 完成数据依赖关系、数据共享关系的构建，并建立可用于后期数据挖掘的 Transaction（仅包含所关注的程序元素及关系）
- ✧ 提交的实验报告为 PDF 文件，文件名格式为“学号-姓名-大实验-作业 2.pdf”，文件内容格式不限，但应该清晰标注学号、姓名与高级程序设计 II 大实验作业 2 等信息。**注意：本次作业开始，大实验以分组方式完成，但每位同学均需提交，请实验报告内应明确包含组员信息。**
- ✧ 作业**截止时间** 2020 年 05 月 06 日晚 23:59 (周三)，**在没有明确通知可缓交的情况下，每晚交一天，扣 3 分；缓交扣分最多 30 分（即本次作业若拖到期末才上交，最多可得 70 分）**
- ✧ 提交至 OBE 系统“作业”板块的“三、大实验作业 2”
- ✧ 其他注意事项：
 - ◆ 使用 clang 生成 LLVM 中间表示的命令如：`clang -emit-llvm -S test.c`，
获得文本文件 `test.ll`

使用说明：

1. 请根据第一次作业的使用说明进行。
2. 请安装 Graphviz, 使用其中的 dot 将 .dot 文件转化为图片, 一般命令为: `dot -Tpng -o x.png x.dot`, 可改变 png 为其他格式如 jpg、svg 等。

自学：

1. 调用 Instruction、BasicBlock、Function 等数据结构中的各个函数, 了解其作用;
2. 请自学 LLVM 中另一种重要的数据结构 Value, 并通过 Instruction 相关函数, 了解并掌握 Value 及其与 Instruction 中各元素的关系与作用。

作业中附件说明：

Tests 目录中给出了一个简单的 C 代码文件 (Test2.c)、以及该文件内的函数对应的数据依赖/共享关系图的输出结果 (dot 文件及对应的 jpg 图片)。文件名中含“flat”的是消除无关程序元素之后的结果; 文件中含“transaction”的是将程序元素规范化然后转化为整数之后的结果; 文件中含“legacy”的是仅使用 LLVM IR 自带的信息建立的数据依赖图。

附件包含一个 .cpp 文件和一个 .h 文件, 是我们提供的参考实现, 里边使用中文注释的地方需要同学们自行实现完成。我们提供了构建数据共享关系的代码, 以及将数据依赖图转化为 dot 文件的代码。同学们可以参考该代码实现自己的代码, 或者将其集成到自己的代码中 (某些重要函数及数据结构需要自己设计完成)。

请自行试验在 analyze.sh 脚本文件执行 clang 时使用“-g”选项与不使用该选项时, 结果的区别; 同时实验待分析代码中存在 memcpy 函数调用时 LLVM IR 有何特异。

作业：

请根据相关资料, 学习利用 LLVM 各种数据结构及方法, 遍历 Module 中的 Function、Function 中的 BasicBlock、BasicBlock 中的 Instruction, 完成构建数据依赖关系、数据共享关系、建立 Transaction 的功能, 针对附件的 TestMe.c (与第一次作业所使用的文件相比有所变化) 进行测试, 并将代码和结果文件打包上传到 OBE 中。

要求：

1. 分小组完成 (可单人成组, 但每组不得超过 4 人)
2. 严禁抄袭代码
3. 在代码中使用注释, 说明代码的功能 (如果在参考实现上完成, 需要明确注明添加的代码与改变的代码)
4. 可根据实验课的讲解, 使用迭代式数据流分析方法实现数据依赖的构建, 也可以探索其他可行的方法, 使用其他方法实现的组, 将给予最高 10% 的奖励分
5. 参考实现中缺失了部分重要数据结构、重要函数的实现, 请自行设计补充完整
6. 如果测试中发现参考实现有错, 可以报给老师, 但我们不再提供改正后的实现代码
7. 可先针对给定的示例文件 Test2.c 中的单个函数进行测试, 通过查看结果判断代码是否正确
8. 在截止日期之前, 将代码与针对 TestMe.c 的分析结果打包上传, 并在实验报告中阐述实现方法 (使用、增加了什么数据结构, 实现的功能使用了什么算法、有什么比较有特点的实现【比如可改进效率的实现方式】等), 同时清晰注明组内成员
9. 提交的分析结果应包含完整的数据依赖/共享图 (如附件中的 Test_func.jpg)、消除无关元素之后的图 (如 Test_func.flat.jpg) 和 Transaction 的图 (如 Test_func.transaction.jpg), 不要求提供仅使用 LLVM IR 中的关系构建的数

据依赖图（如 `Test_func.legacy.jpg`，我们称之为 **legacy** 模式）

10. 如果确实无法完成数据依赖关系的构建，可以尝试完成 **legacy** 模式的数据依赖关系（会有扣分），后期数据挖掘等内容则需要在给出的参考算法基础上做出适当改变，但是注意，数据挖掘、生成规则、检测缺陷时会出现错误（如生成错误的规则）导致进一步扣分