

总结感想

编译技术实验课程终于要画上一个不那么圆满的句号了。在这门实验课上最大的收获就是“要学会合理安排时间”！不然就像我一样，在有些人开始写MIPS生成的时候，我还在慢悠悠地赶我的符号表与错误处理（代码生成一还没写）。

编译实验的难度实际上并不是很大，一点一点慢慢做还是能啃完的；要敢于花单独的时间去设计架构，思考编译过程中每一步的细节、可能存在的问题，想通了之后就动手写；写完回顾也就发现其实没想象中那么恐怖了。但是编译实验的体验仍然非常糟糕，具体后面再论述。总之，一学期下来，我锻炼了自己的工程设计能力与代码能力，也得到了一个四千行左右的不那么优雅、不那么高效、不那么简洁的编译器设计，同时还收获了理论与实践相结合带来的知识贯通的爽快感。

接下来是本人对编译课程的一点拙见，还请老师、助教们见谅。那就是竞速排序，众所周知在本课程的竞速排序机制下，你的分数不完全取决于你的付出，更要看其他人的付出程度，这极大地增加了同学之间不必要的竞争，为了获取一个相对较高的分数而功利性地针对数据点进行优化，而不是静下心来仔细学习编译优化的原理，写出健壮的、通用的优化算法；或者说，大家都尽了自己最大的努力，最后却依然得到同样的分数。并且竞速排序在整个实验中的占比过大，LLVM中间代码的实现难度并不小，从LLVM IR到MIPS生成的难度也不大，而做前者只能得到最高85分，后者需要卷出天际才能得到相对较高的分数。因此我个人认为，应当适当调整分数配比，提高中间代码生成的分数占比至90甚至95分，将其余分数分给代码优化；并且代码优化的给分不再采用竞速制，而是设置有梯度的“挑战点”，完成对应点的任务就能得到固定的分数。

最后，还请老师、助教们审阅我的建议，并对不正确、不合理之处提出批评，谢谢！