# 自己工作结果的评价

1. 我的程序模块仿照冯诺依曼体系结构（运算器、控制器、存储器、输入设备和输出设备）分为：alu.c、control.c、Struct.c、io.c四个主要模块。这使得我的程序流程清晰、可读性好。
2. 使用Visual Studio 2019 集成开发环境进行开发，代码整齐、格式统一。
3. 使用不同的数据类型模拟不同的寄存器和内存单元（使用unsigned short类型模拟程序计数器ip，使用short类型（双字节）模拟通用数寄存器，使用char类型（单字节）模拟内存单元），仿照真实PC机，这简化了处理流程，也是代码更好理解。对于相同大小的不同数据类型（例如unsigned short 和short），其在CPU上的二进制运算是统一的，只是其解释方式的不同导致了其表现的不同。因此，使用对应字节大小的数据类型模拟该对象即可。（例如，使用unsigned 或者 signed 模拟寄存器并没有本质区别）。

# 存在的问题

1. 在编写第一版程序之前没系统地做概要设计。仅凭“直觉”，在一晚上将单核版代码打出。这一版代码，复用性低（甚至没有编写“2个byte类型转short类型”、“4个byte类型转int类型”的转换函数），且存在漏洞。虽然没有全局变量，但每个函数都要传递memory和Register的地址，参数过多。并且数据类型处理逻辑比较混乱，调试难度很大。这些原因使我考虑打第二版程序。

参考张艳梅老师去年的课件，对第一版进行了优化（我发现，张老师建议的模块划分方法以及寄存器和内存的数据结构的选取，都和我的第一版程序相似度很高），完成了第二版单核版程序。由于使用了部分第一版代码，其中一些隐藏比较深的错误也被沿袭下来。其中一个BUG：输出内存时使用无符号输出，这个BUG花了我半天的时间去调试和发现。DEBUG效率低。

1. 我发现自己调试代码的能力还需提高，测试方面的能力有所欠缺。
2. 不会精简作图，做出的算法流程图内容杂糅，可读性差。
3. 编写文档相关的能力不足。包括写作能力、概括能力、专业术语的使用、N-S图制图工具的使用、Word的使用，综合能力存在一定的欠缺。

# 启发

1. 先做概要设计，再打代码。
2. 对程序员来说，代码能力很重要，但调试能力也很重要。
3. 多方面提升自身能力，例如：写作能力、概括能力、专业术语的使用、N-S图制图工具的使用、Word的使用。

# 建议

1. 所给指令集理解上存在歧义。因为我代码早早完成，这使得我成为班级同学遇到问题时的询问对象。向我咨询的人都提及了指令集的相关问题。包括我自己在打代码时也被指令集指令所困扰。如果下一届大作业还是这个题目的话，请老师一定要写个指令集指令解释文档，给出更详细的解释。
2. 大二《数据结构课程设计》、《计算机网络课程设计》均需要团队协作完成，如果大一计导大作业也是团队协作的话，会对大二这类课程起到很好的铺垫作用。
3. 如果明年是团队协作大作业，可以选一个更有挑战性的、拆分性更好的题目，避免一个同学完成几乎所有工作的情况。