《汇编语言程序设计实验》

实验四任务

实验名称：实验四 模块化程序设计

1. 实验目的和要求
2. 掌握子程序设计的方法与技巧，熟悉子程序的参数传递方法和调用原理；
3. 掌握宏指令、模块化程序的设计方法;
4. 掌握较大规模程序的合作开发与调试方法；
5. 掌握汇编语言程序与C语言程序混合编程的方法；
6. 熟悉C编译器的基本优化方法;
7. 了解C语言编译器的命名方法，主、子程序之间参数传递的机制。
8. 实验内容

任务1 宏与子程序设计

进一步修改与增强实验三的学生成绩查询程序的功能。程序执行时首先显示一个功能菜单: 然后，根据用户选择的菜单项，完成相应的功能。

1=录入学生姓名和各科考试成绩

2=计算平均分

3=按平均分排序从高到低排序，排序结果仍存放在原缓冲区中

4=输出成绩单

5=程序退出。

要求：

1. 2人一组，一人负责包括菜单显示、程序退出在内的主程序，以及菜单中的功能1和2；另一人负责菜单中的功能3和4。各自汇编自己的模块，然后连接生成一个程序。
2. 录入学生姓名和各科考试成绩时，首先显示录入的是第几个学生的信息，然后分别在提示之后输入姓名和各科成绩。所有学生信息录入完毕后回到菜单显示的位置。姓名及考试成绩的存放、平均分的计算，按照实验三的要求。
3. 按平均分排序从高到低排序，排序结果仍存放在原缓冲区中。
4. 输出成绩单时，依次显示每个学生的姓名、各科成绩，平均成绩。可在排序前调用，也可在排序后调用。
5. 将9号和10号DOS系统功能调用定义成宏指令并调用。使用子程序完成各模块的功能。实现串、数转换时，也应使用子程序来完成。

**上述任务1中，值得思考的问题提示：**

1. 在TD中跟踪到子程序内部有几种方法？在TD中观察子程序调用和返回时堆栈的变化。
2. 注意观察FAR、NEAR类型子程序的RET指令的机器码有何不同？观察FAR类型子程序被调用时堆栈的变化情况。
3. 通过把一个模块拆成多个模块或反之，体会子程序和模块化程序设计的方法，体会模块调用关系图、子程序功能说明、输入/输出说明在程序设计中的作用。
4. 在编程中使用不同的子程序参数传递方法来编写子程序。
5. 观察模块间的参数的传递方法，包括公共符号的定义和外部符号的引用，若符号名不一致或类型不一致会有什么现象发生？
6. 通过TD观察宏指令在执行程序中的替换和扩展，解释宏和子程序的调用有何不同。

任务2：在C语言程序中调用 汇编语言实现的函数

对于任务1的程序进行改造，主控程序、以及输入输出等功能用C语言实现，其他功能用独立的汇编语言子程序的方式实现； 在C语言程序中调用汇编语言子程序。

**要求与提示：**

(1)在不同的C语言开发环境中实现与汇编语言程序的混合编程，其操作方法有可能是不同的。请大家选择自己熟悉的C语言开发环境并查找相关的资料完成本实验。

1. 观察主、子程序之间参数传递的机制，通过堆栈传递参数后堆栈空间回收的方法。
2. 对混合编程形成的执行程序，用调试工具观察由C语言形成的程序代码与由汇编语言形成的程序代码之间的相互关系，包括段、偏移的值，汇编指令访问C的变量时是如何翻译的，等。

选做与思考题：

1. 如何使菜单和成绩单显示得更漂亮一点？
2. EXTRN说明语句放在.386之前或者之后有什么区别？
3. EXTRN说明的变量的段与段寄存器的关联关系（ASSUME伪指令所表达的信息）是否能带入到本模块中？如果不能带入，是否可以通过加段前缀的方法来解决？
4. 请尝试在C语言源程序中不合理地嵌入汇编语言的指令语句，达到破坏C语言程序的正确性的目的。比如，在连续的几条C语言语句中间加入一条修改AX寄存器（或DS等其他寄存器）的汇编指令语句，而AX的内容在此处本不该被修改，这样就可观察到破坏C语言程序正确性的效果（该项实验表明：在C语言程序中，若不考虑上下语句翻译成怎样的机器码而随意嵌入汇编指令语句时，有可能存在出错的风险）。
5. 观察C编译器的优化策略对代码的影响。

**参考文献：**

[1]许向阳，《80X86汇编语言程序设计上机指南》“第十一章 汇编语言程序与C程序的连接”。