《计算机系统基础实验》任务书

# **实验四 模块化程序设计**

# 实验目的与要求

1. 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；
2. 掌握模块化程序的设计方法;
3. 掌握汇编语言程序与C语言程序混合编程的方法；
4. 理解模块之间的信息传递与组装的基本方法；

# **实验内容**

（上机实验环境说明：本次实验使用使用Linux环境，需要使用gdb、objdump、readelf工具。)

## 任务3.1 采用子程序、多模块等编程技术设计实现一个较为完整的计算机系统运行状态的监测系统。

主要功能为(以任务1.4为基础)：

1. 程序执行后，提示输入用户名（用户名请定义成自己的姓名拼音）和密码，如果不正确，提示出错信息后再次重新输入，最多三次出错机会，三次都出错时程序退出。若用户名和密码都正确，则继续后续处理。
2. 在接下来的处理中，先对N组状态信息分别计算f，并进行分组复制；
3. 然后将MIDF存储区的各组数据在屏幕上显示出来。
4. 最后，按R键重新从“（2）”处执行，按Q键退出程序。

要求：

1. 计算f、复制每组的数据、显示MIDF存储区的各组数据等功能均分别用子程序实现。
2. 功能（1）（3）（4）用 C语言程序实现。
3. 通过实验操作，观察：子程序与主程序之间是如何传递信息的？刚进入堆栈时，堆栈栈顶及之下存放了一些什么信息？执行 CALL 指令及RET指令，CPU完成了哪些操作？若执行RET前把栈顶的数值改掉，那么RET执行后程序返回到何处？子程序中的局部变量的存储空间在什么位置？如何确定局部变量的地址？访问局部变量时的地址表达式有何特点？
4. 将汇编源代码至少分解到两个不同的源文件中。
5. 进一步可观察的内容：

观察模块间的参数的传递方法，包括全局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？

* 进度提示：

本实验使用一次课内上机，4学时完成。

# **提示**

1. C语言与汇编语言的混合编程有两种形式：一种就是C语言的模块与汇编语言的模块组合到一个工程里，编译连接成一个程序；另一种就是在C语言程序中直接嵌入汇编语言指令语句。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（尤其是在几行计算公式对应的C语句的中间）加入一条修改EAX寄存器（或EBP、DS等寄存器）的汇编指令语句，若EAX的内容在此处本不该被修改，则可观察到破坏了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）。
2. 通过调试混合编程的程序，体会与纯粹汇编语言编写的程序的调试过程的差异。
3. 通过本次实验，希望大家明白：不同的编程语言是可以协同解决一个问题的，而且可以利用不同语言的特点来更好地解决问题；利用汇编语言的知识，能够更好地理解高级语言的内部处理原理与策略，为编写更好的C语言程序、用好C编译器提供支持。

可参考《Linux的汇编语言程序设计》，第十三章。