《汇编语言程序设计实验》题目汇总

1. 实验的安排与要求
2. 简单程序设计，DOS功能调用，上机操作与TD的使用

课时安排：1.5学时讲解上机环境、工具使用；1学时自行练习；1.5学时交互检查一个实验操作点。

|  |  |
| --- | --- |
| 实验内容 | 将教材中关于数字运算、标志寄存器的改变、变量与寻址方式等知识点涉及的作业题改编成程序，用TD调试工具验证；利用DOS功能调用编写简单程序，记录实验结果并与预计的结果比较。 |
| 类型 | 验证性、设计型实验 |
| 实验目的与要求 | 1. 理解数、符号、寻址方式等在计算机内的表现形式； 2. 掌握汇编源程序编辑工具、汇编程序、连接程序、调试工具的使用； 3. 理解指令执行与标志位改变之间的关系； 4. 熟悉常用的DOS功能调用。 |
| 任务 | 1. 《80X86汇编语言程序设计》教材中 P31的 1.14题。（TD与指令、标志位）   要求：(1) 直接在TD中输入指令，完成两个数的求和和求差的功能（可以省略实验报告中的程序框图）。求和/差后的结果放在(AH)中  (2) 请事先指出执行指令后(AH)、标志位 SF、OF、CF、ZF的内容。  (3) 记录上机执行后与（2）中对应的内容。  (4)求差运算中，若将A、B视为有符号数，且A>B, 标志位有何特点？  若将A、B视为无符号数，且A>B, 标志位又有何特点？   1. 《80X86汇编语言程序设计》教材中 P45的 2.3题，2.4题。(编程过程，观察数据段，寻址方式) 2. 在2.3题的标号LOPA前加上提示语句“Press any key to begin!”,并实现按键之后才执行后续程序的功能。 3. 多种方法在TD数据窗口找到数据段。在TD代码窗口观察机器指令程序在内存中的存放形式，理解机器码与汇编语句对应关系，尤其注意操作数寻址方式的形式；观察连续存放的二进制串在解析成汇编语句时，从不同位置开始解析，结果是不同的，理解 IP/EIP指明指令起始位置的重要性。 |

1. 分支程序、循环程序的设计

课时安排：0.5学时讲解注意事项；2学时自行练习；1.5学时交互检查一个实验操作点。

|  |  |
| --- | --- |
| 实验内容 | 分别设计分支、循环程序，实现字符串比较、进制转换、成绩统计等功能。 |
| 类型 | 设计型实验 |
| 实验目的与要求 | 1. 熟悉分支、循环程序的结构及控制方法，掌握分支、循环程序的调试方法； 2. 加深对转移指令及一些常用的汇编指令的理解； 3. 加深对常用DOS功能调用指令的理解，进一步熟悉TD的使用。 |
| 任务 | 在以BUF为首址的字节数据存储区中，存放着n个学生的课程成绩表，每个学生的相关信息包括：姓名（占10个字节，结束符为数值0），语文成绩（1个字节），数学成绩（1个字节），英语成绩（一个字节），平均成绩（1个字节）。提示输入学生姓名，然后在成绩表中寻找该学生；若未找到，就显示不存在，要求重新输入姓名；若找到，则将语文成绩乘以权重2、英语成绩除以权重2后与数学一起计算该生的平均成绩（百分制）并保存起来（给出公式(A\*2+B+C/2）/3.5)。然后判断该成绩的范围，大于等于90分，则显示“A”,大于等于80分，显示“B”, 大于等于70分，显示“C” , 大于等于60分，显示“D” , 小于60分，则显示“F”。回到提示输入姓名处。若此时输入字符q，则程序退出。  (思考题选做：（1）若要显示平均成绩如何做？（2）若所有成绩项采用字类型定义，会有哪些不同之处？)  避免溢出。 |

1. 程序执行时间与代码长度优化

课时安排：0.5学时讲解优化实例；2学时自行练习；1.5学时交互检查一个实验操作点。

|  |  |
| --- | --- |
| 实验内容 | 将实验二实现的程序用C语言编写，然后将C语言程序反汇编，比较两者的代码长度，分析反汇编后的程序在代码级的优化方法；探讨对实验二中实现的汇编语言程序进一步优化的方法。 |
| 类型 | 综合性实验 |
| 实验目的与要求 | 1. 熟悉指令的特点、代码优化的基本方法； 2. 理解高级语言程序与汇编语言程序之间的对应关系。 |
| 任务 | 1) 将实验二实现的程序用C语言编写，给出具体的对比分析报告。  2）汇编程序的优化。将实验二改成首先计算所有学生的平均成绩，再查询到指定学生的成绩后显示结果。为了便于体现时间优化的效果，将n取可能的最大数（初始化值都可以是一样的），在计算n个平均值的循环体外增加一个64k或更大的循环体。  时间的优化重点从复杂运算指令的替换、减少指令在循环体中执行的次数等角度开展，可以考虑增加程序运行时间计时功能。代码长度的优化重点从不必要指令的消除，通过调整算法进行操作合并，长字节指令的替换等方面开展。  优化不要求面面俱到，重点突出基于指令集特点的优化。 |

1. 子程序的设计与调用，模块化程序设计

课时安排：0.5学时讲解注意事项；2学时自行练习；1.5学时交互检查一个实验操作点。

|  |  |
| --- | --- |
| 实验内容 | 利用格式化的输入输出、进制转换、排序等算法和技术，编制复杂的模块化程序并跟踪调试，充分发挥宏、子程序的作用，认真观察堆栈的变化。 |
| 类型 | 设计型实验 |
| 实验目的与要求 | 1. 掌握子程序设计的技巧方法，熟悉子程序的参数传递方法和调用原理； 2. 掌握宏指令、模块化程序的设计方法; 3. 掌握较大规模程序的合作开发与调试方法。 |
| 任务 | 1. 显示一个功能菜单，选择1=输入姓名和考试成绩，2=计算总分与平均分，3=排序，4=输出排序后的成绩单（从高到低的顺序，以用户指定的进制形式）。 2. 2人一组，一人负责包括菜单显示的主程序，以及功能1和2；另一人负责功能3和4。各自调试自己的模块，然后合并后联合调试。观察段的分配。   进一步明确模块的组成，如果奇数，允许3人一组。可以由老师按照名单分组。 |

1. 与C语言混合编程

课时安排：0.5学时讲解注意事项；2学时自行练习；1.5学时交互检查一个实验操作点。

|  |  |
| --- | --- |
| 实验内容 | 用C语言程序调用汇编语言编写的子程序，用汇编语言的程序调用C的函数，在C语言程序中嵌入汇编语言指令语句；观察在给C语言编译器设定不同的优化策略时，机器代码的差异。 |
| 类型 | 设计型实验 |
| 实验目的与要求 | 1. 掌握汇编语言程序与C语言程序混合编程的方法； 2. 熟悉C编译器的基本优化方法; 3. 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。 |
| 任务 | 对于实验四的程序进行改造，选择其中两个功能用C语言实现，并在C程序中使用ASM 前缀的方式嵌入汇编指令语句，观察可能出现的寄存器冲突。观察段的分配。使用学生学习C语言时的C编译器（要确认其混合编程的语法）。指定用C编写的模块，只要求用C调用汇编函数。 |

1. 中断处理程序设计

课时安排：0.5学时讲解注意事项；2学时自行练习；1.5学时交互检查一个实验操作点。

|  |  |
| --- | --- |
| 实验内容 | 利用调试工具观察端口、中断矢量表等信息；通过访问实时时钟芯片、调用BIOS功能、修改时钟中断处理程序等，实现一个实时时钟显示的中断处理程序；也可以通过修改键盘中断重新定义键盘。 |
| 类型 | 设计型实验 |
| 实验目的与要求 | 1. 掌握中断矢量表的概念； 2. 熟悉I/O访问，BIOS功能调用方法; 3. 掌握实方式下中断处理程序的编制与调试方法。 |
| 任务 | * 1. 用三种方式获取中断类型码10H对应的中断处理程序的入口地址。   ① 直接运行调试工具（TD.EXE），观察中断矢量表中的信息；  ② 编写程序，用 DOS系统功能调用方式获取，观察相应的出口参数与① 看到的结果是否相同 （使用TD观看即可）。  ③ 编写程序，直接读取相应内存单元，观察读到的数据与 ① 看到的结果是否相同 （使用TD观看即可）。   * 1. 编写一个中断服务程序并驻留内存，要求在程序返回DOS操作系统后，键盘的按键A变成了按键B。   2. 通过读写实时时钟内的信息，计算程序段执行的时间。用到以前某个程序中。 |

1. 跟踪与反汇编，反跟踪程序设计

课时安排：0.5学时讲解方法实例；2学时自行练习；1.5学时交互检查一个实验操作点。

|  |  |
| --- | --- |
| 实验内容 | 对未知密码的执行文件进行反汇编，阅读和跟踪执行，暴力破解或找出密码；为已知密码的源程序添加反跟踪代码。 |
| 类型 | 综合性、设计型实验 |
| 实验目的与要求 | 1. 熟悉跟踪与反跟踪的技术； 2. 提升对计算机系统的理解与分析能力。 |
| 实验考核 | 1. 在实验二的基础上，增加查询前输入密码的功能，密码存放分为明文存放、密文存放两种情况，加密方法自选。各科成绩及平均分也分明文和密文存放两种情况（简单的加密方法可以在源程序数据段定义时用数值运算符及其组合实现）。可以采用计时、中断矢量检查、堆栈检查、间接寻址等方式反跟踪。 2. 两人一组，每人实现一类加密与反跟踪方法，把程序交给对方解密。 |

1. WIN32编程

课时安排：0.5学时讲解注意事项；2学时自行练习；1.5学时交互检查一个实验操作点。

|  |  |
| --- | --- |
| 实验内容 | 将前几次实验中的某一题改成基于窗口的WIN32程序，观察WIN32执行程序代码的特点和执行流程。 |
| 类型 | 设计型实验 |
| 实验目的与要求 | 1. 熟悉WIN32程序的设计和调试方法； 2. 熟悉宏汇编语言中INVOKE、结构变量、简化段定义等功能； 3. 进一步理解机器语言、汇编语言、高级语言之间以及实方式、保护方式之间的一些关系。 |
| 任务 | 参考实验四改成WIN32程序。去掉串输入操作。使用INVOKE、结构变量、简化段定义。 |