实验一 汇编程序上机环境的熟悉

一、实验目的

1、学会使用汇编程序集成开发环境来编辑源文件(*.ASM)、调用 MASM 宏汇编程序对源文件进行汇编，获得目标程序 (*.OBJ)、调用 LINK 连接程序将汇编后的目标文件(*.OBJ)连接成可执行的文件(*. EXE)。

2、学会使用 DEBUG 调试程序把可执行文件装入内存并调试运行，熟悉 DEBUG 命令。

3、上机环境：基于 windows 的可视化集成开发环境。

二、实验内容

1、完成编辑源文件、汇编、连接、运行并调试。

汇编程序对源程序进行翻译，生成扩展名为 OBJ 的目标文件，请注意：汇编程序会对源程序进行语法分析，如果出现严重错误（error），则需要修改错误，直到没有严重错误，才会生成目标文件；

连接程序是将目标程序和库文件进行连接、定位，生成扩展名为 EXE 的可执行文件；

运行程序是看程序的运行结果，运行完毕返回 DOS；调试程序是对可执行文件进行调试运行，验证它的正确性。

2、实验过程

①、安装汇编程序集成开发环境，安装文件：asm20125；

②、在汇编程序集成开发环境下，编辑源文件、汇编、连接、运行。

3、调试程序

①用 DEBUG 调试程序调试：**附：**DEBUG 调试程序各种命令的使用方法

功能	命令格式	使用说明
显示内存单元内容	D 地址	从指定地址开始显示 80H 个字节
修改内存单元内容	E 地址	先显示地址和单元内容等待输入修改的内容
检查和修改寄存器的内容	R	显示全部寄存器和标志位及下条指令单元十六进制数码和反汇编格式
反汇编	U 地址	从指定地址开始反汇编 16 个或 32 个字节
汇编	A 地址	从指定地址直接输入语句并从指定指定汇编装入内存

跟踪	T＝地址	从指定地址开始逐条跟踪指令
运行	G＝地址	无断点，执行正在调试的指令
退出	Q	退出 DEBUG 返回 DOS

附：标志寄存器

在 DEBUG 调试程序下标志寄存器各个标志位的含义：

OV、NV 即溢出标志位 OF=1 或 0，表示运算结果有无溢出。当算术运算的结果超出了带符号数的范围（8 位带符号数的范围是-128——+127，16 位的是-32728——+32767）。

DN、UP 即方向标志位 DF=1 或 0，表示串操作时按地址减或加的方式进行。

EI、DI 即中断标志位 IF=1 或 0，表示 CPU 可否响应可屏蔽中断请求。IF 的状态对不可屏蔽中断及内部中断没有影响。

NG、PL 即符号标志位 SF=1 或 0，表示运算结果是负（最高位为 1）或正（最高位为 0）。

ZR、NZ 即零标志位 ZF=1 或 0，表示运算结果是 0 或非 0。

AC、NA 即辅助进位标志位 AF=1 或 0，在 8 位加减法操作中表示低 4 位向高 4 位有无进借位。

PE、PO 即奇偶标志位 PF=1 或 0，表示逻辑运算结果中 1 的个数是否为偶数。

CY、NC 即进位标志位 CF=1 或 0，表示加减法操作中最高位有无进借位。

三、上机 step by step

注：希望同学们对照上机操作！难点：debug 常用命令的使用。

（1）在汇编程序集成开发环境下：

源程序如下：

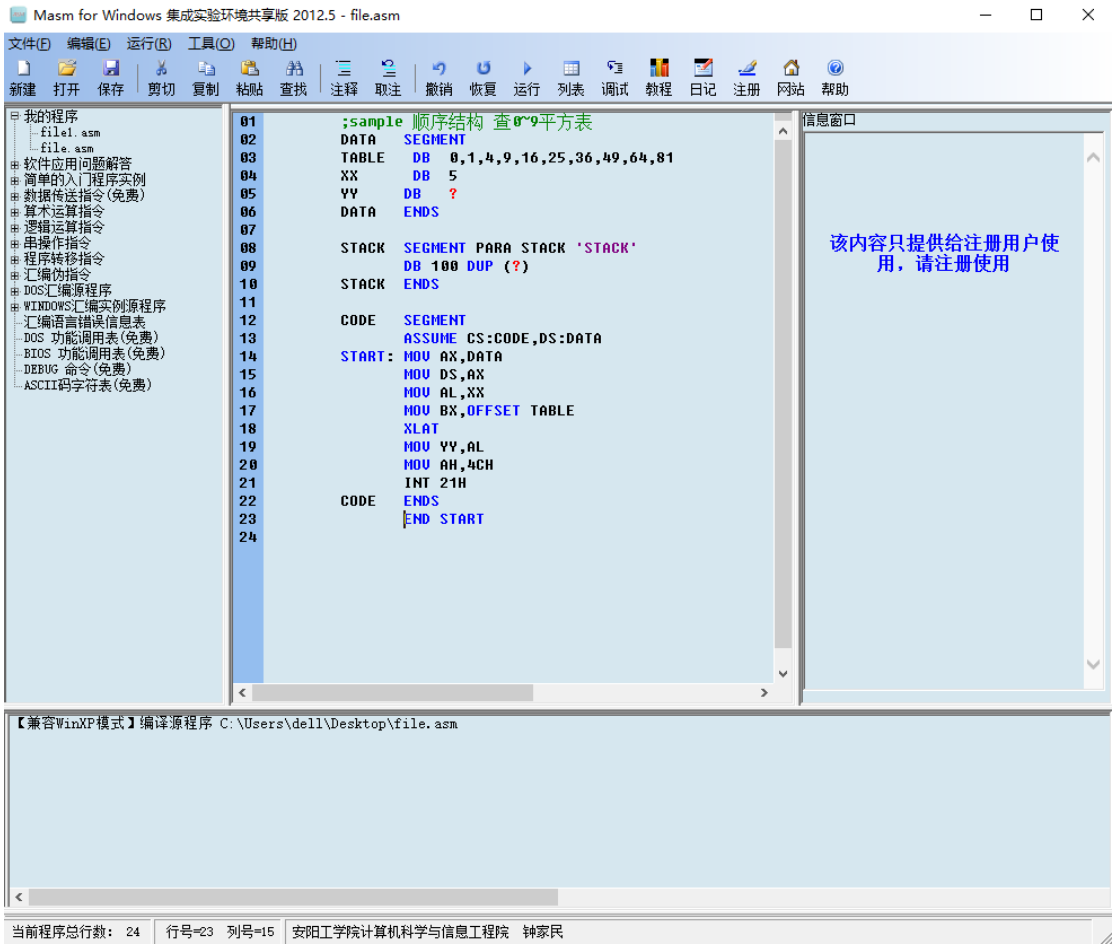
```
;sample  顺序结构  查 0~9 平方表
DATA     SEGMENT          ;数据段定义
TABLE   DB  0,1,4,9,16,25,36,49,64,81      ;平方表定义
XX       DB  5
YY       DB  ?                             ;存放查表结果
DATA     ENDS

STACK    SEGMENT PARA STACK 'STACK'       ;堆栈段定义
         DB 100 DUP (?)
STACK    ENDS

CODE     SEGMENT          ;代码段定义
```

```
        ASSUME CS:CODE,DS:DATA
START: MOV AX,DATA
        MOV DS,AX          ;装载段地址
        MOV AL,XX
        MOV BX,OFFSET TABLE
        XLAT
        MOV YY,AL
        MOV AH,4CH
        INT 21H            ;结束程序并返回 dos
CODE    ENDS
        END START
```

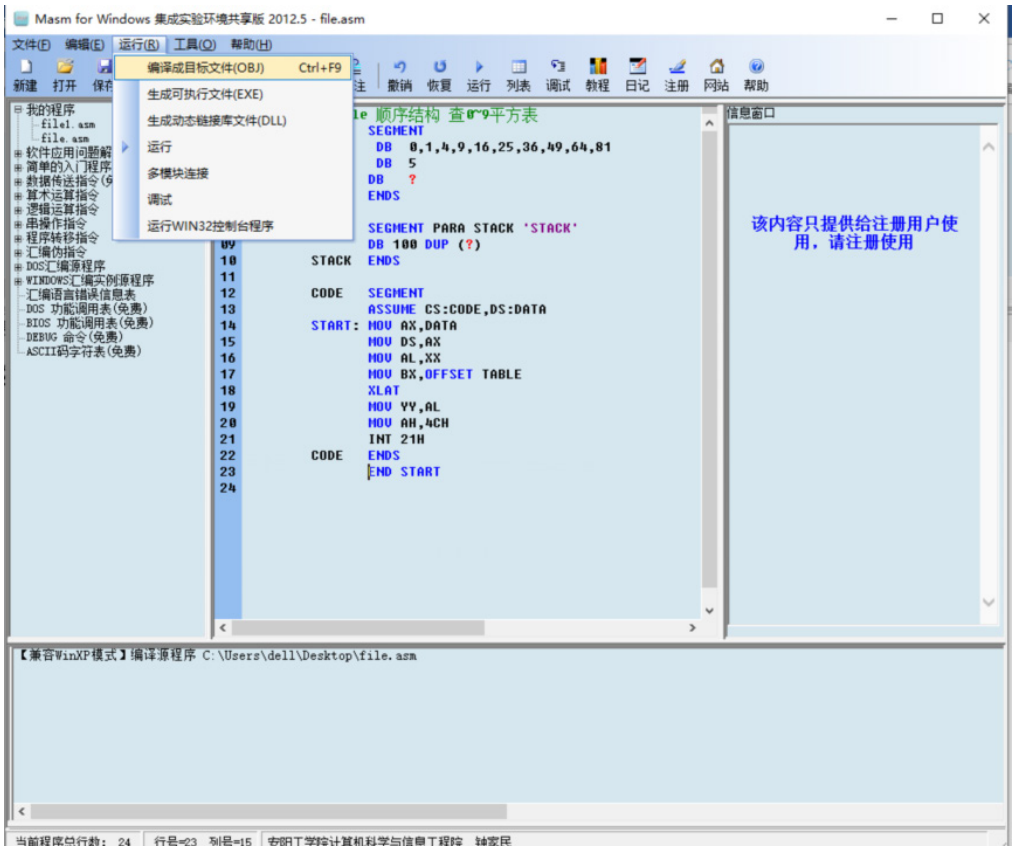
2、在汇编开发环境下编辑源文件，并存盘为 file.asm.



3、编译与链接、运行子菜单

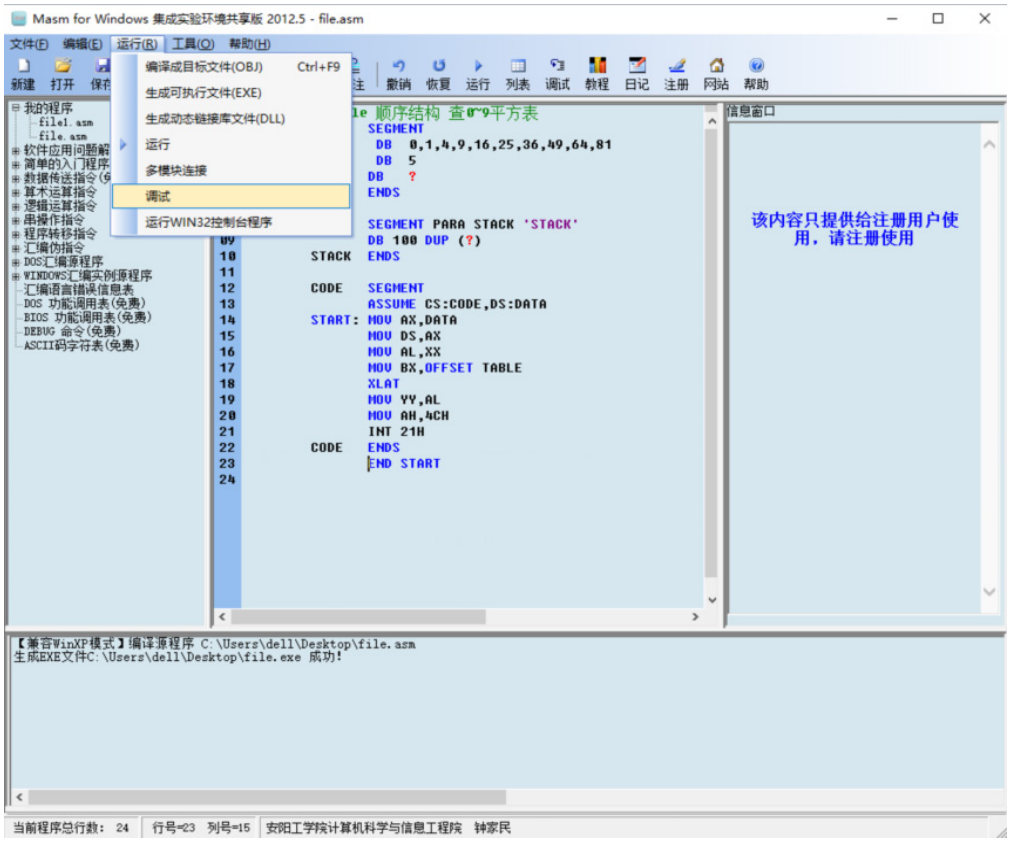
点击菜单：运行，进入子菜单：编译成目标文件 obj、生成可执行文件 exe。

编译：编译成目标文件 obj；链接：生成可执行文件 exe。在下面信息窗口中提示 obj 文件生成和 exe 文件生成。

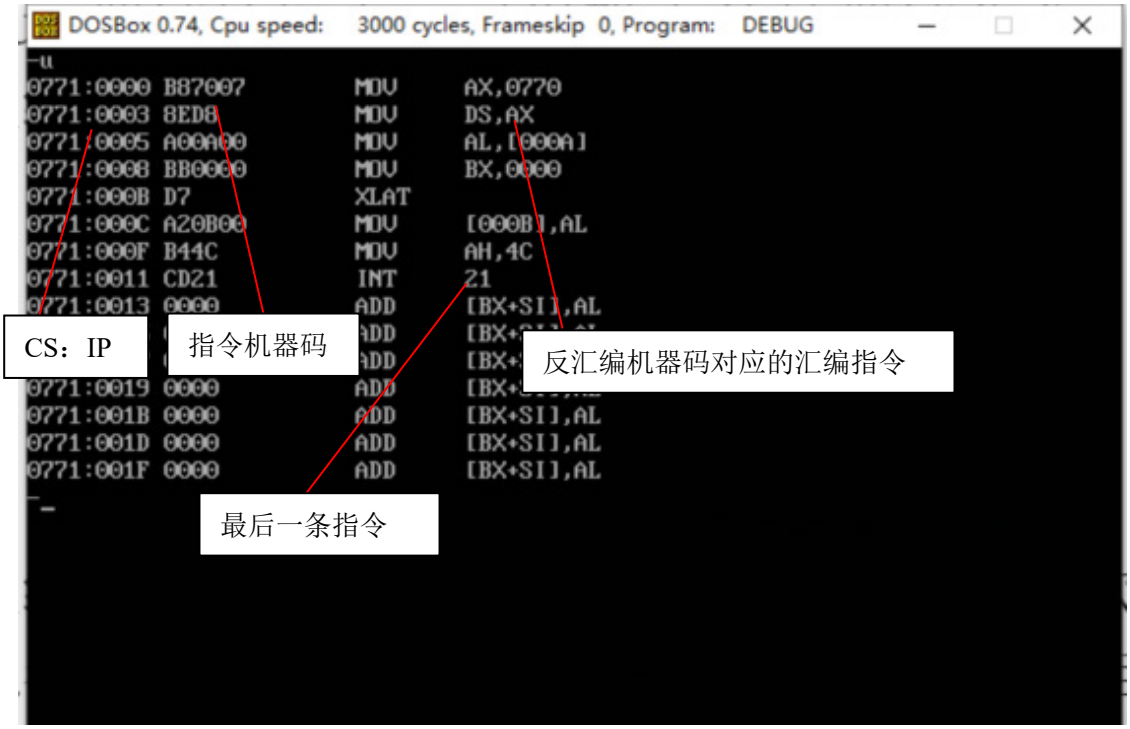


4、调用 debug 调试软件看运行结果（在这里，运行结果没有输出到显示器，而是存放在内存单元中，所以要到内存去看结果）

点击菜单：运行，进入子菜单：调试。



进入 debug 调试环境下，可以输入各种调试命令。
首先反汇编：输入 u



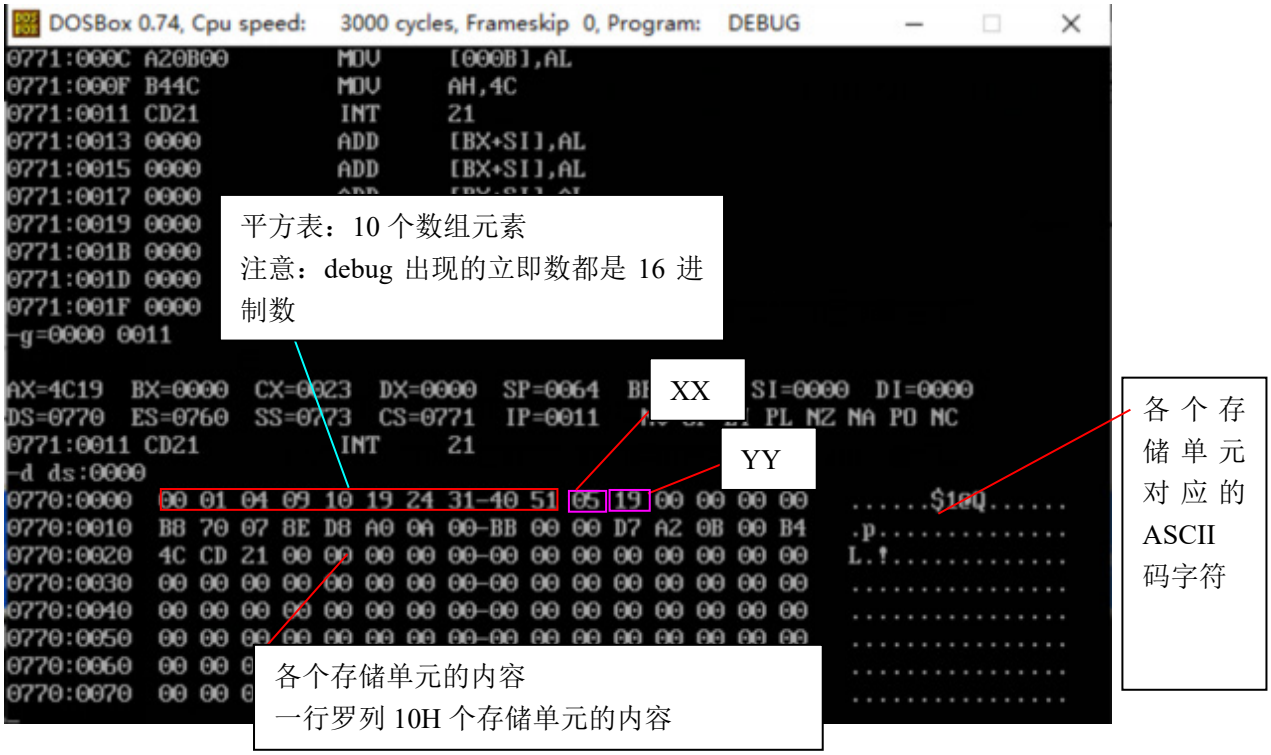
通过反汇编，可以知道各条指令的偏移地址。一次反汇编命令，反汇编 20H 个字节的机器码，得到汇编指令。若程序较长，可以多次 U 命令反汇编，直到看到最后一条指令。

可以看到，就是自己编写的程序，只是符号地址都替换成立即数了。我们程序的最后一条指令是 INT 21。紧接着的下面内存反汇编得到的指令不是我们的程序了。注意：debug 调试环境下默认 16 进制数（H 省略）。

此时，CPU 并没有执行程序，debug 只是把 exe 装载到内存了。
而我们要 CPU 运行程序，在返回 DOS 系统结束程序之前，在数据段看运行结果。
-g=0000　0011↙

上述命令意思是让 CPU 执行程序：从 IP=0000 开始处执行，也就是第一条指令，在 IP=0011 指令处暂停，也就是断点处停止运行。而 IP=0011 处存放的就是最后一条指令：INT 21，最后一条指令就是返回 DOS 系统的。注意：最后一条指令没有执行。

运行结果，输入命令-d ds:0000
察看数据段内存区 80H 个存储单元。



ds:000C 就是 YY 变量的逻辑地址，可以看到存储内容是 19H，运行结果正确！
对于调试，实验要求掌握 DOS 命令符下调用 debug 系统软件。

四、实验题目

1、调试看内存。（1）MAX=？（2）若将 JGE 指令替换为 JAE，MAX=？为什么？（3）给程序写注释。

```
DATA    SEGMENT
BUFF    DB      79H,98H,23H,67H,0A8H
MAX     DB      ?
DATA    ENDS

STACK   SEGMENT PARA STACK 'STACK'
        DB 100 DUP (?)
STACK   ENDS

CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA,SS:STACK
START: MOV AX,DATA
        MOV DS,AX

        MOV     CX,5
        LEA     SI,BUFF
        MOV     AL,[SI]
        DEC     CX
        INC     SI
    LP: CMP     AL,[SI]
        JGE     G1
        MOV     AL,[SI]
    G1: INC     SI
        LOOP    LP
        MOV     MAX,AL

        MOV     AH,4CH
        INT     21H

 CODE   ENDS
        END START
```

2、有两个无符号字节型数组，设数组元素个数相等，编程将数组中的对应元素相加，结果存入另一个内存区。

1) 不考虑进位。

①已知数据段的定义：

```
org    2000h
num1   db     12h,95h,0f1h,0c2h,82h,2h,10h,34h
count equ      $-num1
org    3000h
```

```
num2    db      23h,0dfh,23h,3fh,3ch,0b3h,57h,3h
org    5000h
res     db      count dup(0)
```

②debug 调试：修改 num1 和 num2 数组元素的值，运行看结果。
提示：-e2000
-d2000

2) 考虑进位。
①已知数据段的定义：

```
org    2000h
num1    db     12h,95h,0f1h,0c2h,82h,2h,10h,34h
count equ     $-num1
org    3000h
num2    db      23h,0dfh,023h,3fh,3ch,0b3h,57h,3h
org    5000h
res     dw      count dup(0)
```

②debug 调试：修改 num1 和 num2 数组元素的值，运行看结果。
提示：-e2000
-d2000

3、课题习题上机：已知数据段有个数组，编程求和，平均分。

```
stu  db      60,82,53,92,77,35,69,95,74，88
len  equ     10
res  dw      ?
ave db       ?
```