# 

# 《汇编语言》

# 实验指导书

计算机学院网络工程系

2020.10.5

# 实验一 汇编语言上机环境及基本操作

## **一、实验目的及要求**

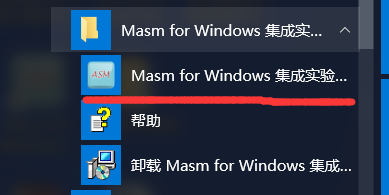
1．学习及掌握汇编语言源程序的书写格式和要求，明确程序中各段的功能和相互之间的关系。

2．熟练掌握在计算机上建立、汇编、连接、调试及运行程序的方法。

## 二、实验内容

### 1．汇编语言源程序的建立

**(1) 进入汇编语言集成开发环境，如图1所示。**

**图1**

**（2）或双击桌面的 “Masm for Windows 集成实验环境 2015”快捷文件图标，进入汇编语言集成开发环境。**

**（3）本例中给出的程序是要求编写3+5=8的程序，并把8存入AL或AX中。设定源程序的文件名为ABC.ASM 。(可用任何一个种文本编辑器编写此程序，存盘)**

DATAS SEGMENT

;此处输入数据段代码

DATAS ENDS

STACKS SEGMENT

;此处输入堆栈段代码

STACKS ENDS

CODES SEGMENT

ASSUME CS:CODES,DS:DATAS,SS:STACKS

START:

MOV AX,DATAS

MOV DS,AX

;======================================

MOV AX ,3 ;自己写的代码

Mov BX ,5 ;自己写的代码

ADD AX ,BX ;自己写的代码(只有3条)

;此处输入代码段代码

;======================================

MOV AH,4CH

INT 21H

CODES ENDS

END START

程序输入完毕后一定要将源程序文件存入盘中，以便进行汇编及连接，也可以再次调出源程序进行修改。

### **2． 将源程序文件汇编成目标程序文件，如图2所示**

**一般情况下，汇编程序MASM.exe的主要功能有以下3点：**

（1）检查源程序中存在的语法错误，并给出错误信息。

（2）源程序经汇编后没有错误，则产生目标程序文件，扩展名为 .OBJ。

（3）若程序中使用了宏指令，则汇编程序将展开宏指令。

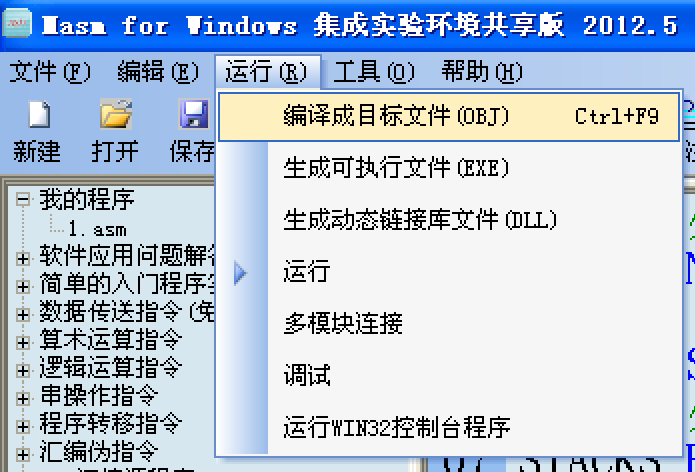


图2

### **3．用链接程序link.exe生成可执行程序文件，本例中给出的程序是要求编写3+5=8的程序，并把8存入AL或AX中。设定源程序的文件名为ABC.ASM 。**

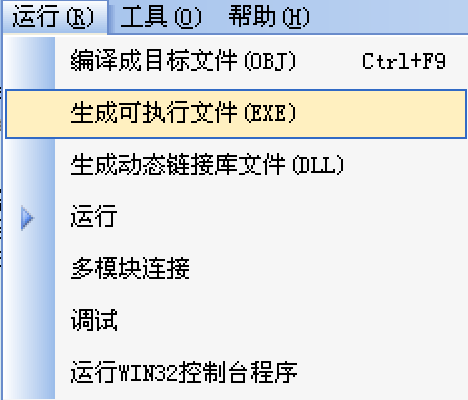
经汇编以后产生的目标程序文件（扩展名为.OBJ文件）并不是可执行程序文件，必须经过链接以后，才能生成可执行文件（即扩展名为.EXE）如图3 所示。

图3

链接以后，便可以产生可执行程序文件（如ABC.EXE文件）。

### **4．程序的执行，如图4所示**

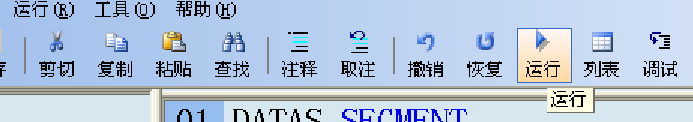


图4

本程序当中没有编写输出部分代码，所以在屏幕上看不到程序执行的结果。我们可以采用调试程序DEBUG来进行检查，相关内容请参见实验二。

# 实验二 DEBUG调试程序的应用

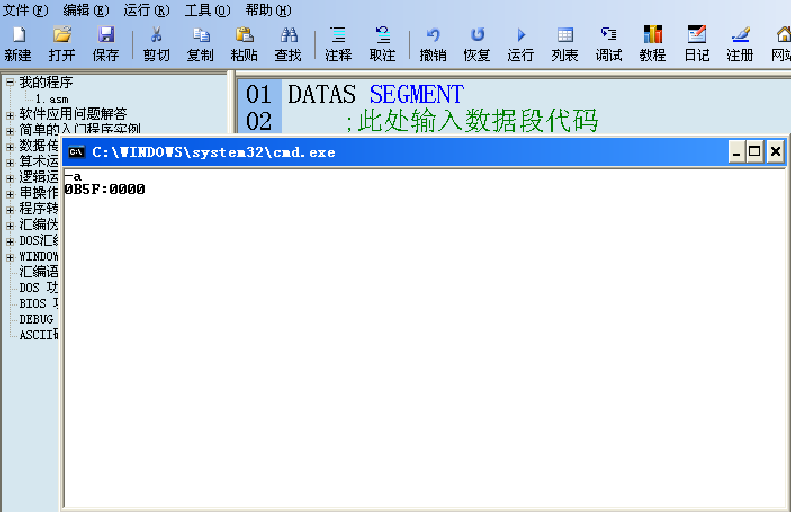
**一、实验目的**

DEBUG.EXE程序是专门为分析、研制和开发汇编语言程序而设计的一种调试工具，具有跟踪程序执行、观察中间运行结果、显示和修改寄存器或存储单元内容等多种功能。它能使程序设计人员或用户触及到机器内部，因此可以说它是80X86CPU的心灵窗口，也是我们学习汇编语言必须掌握的调试工具。

**二、实验内容**

### **1．在DEBUG环境下建立和汇编程序**

在DEBUG环境下用户可以直接建立汇编语言源程序，并可以进行编辑、修改和调试。



## **二、常用DEBUG命令**

DEBUG命令是在命令提示符“\_”下由键盘键入的。每条命令以单个字母的命令符开头，然后是命令的操作参数，操作参数与操作参数之间，用空格或逗号隔开，操作参数与命令符之间用空格隔开，命令的结束符是回车键（ENTER）。命令及参数的输入可以是大小写的结合。Crtl+Break键可中止命令的执行。Crtl+NumLock键可暂停屏幕卷动，按任一键继续。所用的操作数均为十六进制数，不必写H。

### 1．汇编命令A

格式： （1）A <段寄存器名>：<偏移地址>

（2）A <段地址>：<偏移地址>

（3）A <偏移地址>

（4）A

功能：汇编命令是将用户输入的汇编语言指令，汇编为可执行的机器指令。键人该命令后显示段地址和偏移地址并等待用户从键盘逐条键入汇编语言指令。每当输入一行语句后按“ENTER”键，输入的语句有效。若输入的语句中有错，DEBUG会显示“^ Error”，要求用户重新输入，直到显示下一地址时用户直接键入回车键返回到提示符“\_”。

其中（1）用指定段寄存器内容作段地址；（3）用CS内容作段地址；（4）以CS**:**100作地址。以后命令中提及的各种‘地址’形式，均指（1）、（2）、（3）中A命令后的地址形式。

### 2．显示内存单元命令D

格式： （1）D <地址>

（2）D <地址范围>

（3）D

其中（1）以CS为段寄存器；（3）显示CS**:**100为起始地址的一片内存单元内容。

功能：该命令将显示一片内存单元的内容，左边显示行首字节的段地址：偏移地址，中间是以十六进制形式显示的指定范围的内存单元内容，右边是与十六进制数相对应字节的ASCII码字符，对不可见字符以‘·’代替。

### 3．修改内存单元命令E

格式： （1）E <地址><单元内容>

（2）E <地址><单元内容表>

其中<单元内容>是一个十六进制数，或是用引号’（或”）括起来的字符串；<单元内容表>是以逗号分隔的十六进制数，或是用’或”括起来的字符串，或者是二者的组合。

功能：（1）将指定内容写入指定单元后显示下一地址，以代替原来内容。可连续键入修改内容，直至新地址出现后键入回车<ENTER>为止；（2）将<单元内容表>逐一写入由<地址>开始的一片单元中，该功能可以将由指定地址开始的连续内存单元中的内容，修改为单元内容表中的内容。

例如：\_E DS**:**30 F8,AB,”AB”

该命令执行后，从DS**:**30到DS**:**33的连续4个存储单元底内容将被修改为F8H，ABH，41H，42H。

### 4．填充内存命令F（不用练习）

格式：F <范围><单元内容表>

功能：将单元内容表中的值逐个填入指定范围，单元内容表中内容用完后重复使用。

例如：F 05BC**:**200 L 10 B2,‘XYZ’,3C

该命令将由地址05BC**:**200开始的10H（16）个存储单元顺序填充“B2，58，59，5A，3C，B2，58，59，5A，3C，B2，58，59，5A，3C，B2”

### 5．连续执行命令G（不用练习）

格式： （1）G

（2）G**=**<地址>

（3）G=<地址>，<断点>

其中（2）、（3）中的“=”是不可缺省的。

功能：

* 默认程序从CS**:**IP开始执行。
* 程序从当前的指定偏移地址开始执行。
* 从指定地址开始执行，到断点自动停止并显示当前所有寄存器、状态标志位的内容和下一条要执行的指令。DEBUG调试程序最多允许设置10个断点。
* G=0，3 表示从0地址开始执行到断点地址3（ip=3）自动停止并显示当前所有寄存器、状态标志位的内容和下一条要执行的指令。

### 6．跟踪命令T

格式：T [=<地址>][<条数>]

功能：如果键入T命令后直接按“ENTER”键，则默认从CS**:**IP开始执行程序，且每执行一条指令后要停下来，显示所有寄存器、状态标志位的内容和下一条要执行的指令。用户也可以指定程序开始执行的起始地址。<条数>的缺省值是一条，也可以由<条数>指定执行若干条命令后停下来。

例如：T

该命令执行当前指令并显示所有寄存器、状态标志位的内容和下一条要执行的指令。

例如：T 10

该命令从当前指令开始执行10H条指令后停下来，所有寄存器、状态标志位的内容和下一条要执行的指令。

例如：T =0 10

该命令从0地址开始开始执行10H条指令后停下来，显示所有寄存器、状态标志位的内容和下一条要执行的指令。

### 7．反汇编命令U

格式： （1）U <地址>

（2）U <地址范围>

功能：反汇编命令是将机器指令翻译成符号形式的汇编语言指令。该命令将指定范围内的代码以汇编语句形式显示，同时显示地址及代码。注意，反汇编时一定确认指令的起始地址后再作，否则将得不到正确结果。地址及范围的缺省值是上次U指令后下一地址的值。这样可以连续反汇编。

U 0,9 显示地址0-9 范围的指令

### 8．执行过程命令P

格式：P

功能：执行一条指令或一个过程（子程序），然后显示各寄存器的状态。

说明：该命令主要用于调试程序。它与跟踪命令T的作用类似，T命令是跟踪一条或多条指令，而P命令是执行一条指令（包括带重复前缀的数据串操作指令）或一个完整的过程（子程序）。

例如：设有如下指令序列

：

MOV AH，02H

INT 21H

：

当要执行INT 21H这条指令时，若用T命令进行跟踪，则进入INT 21H程序（DOS功能调用）后，需要数十次T命令才能返回当前程序；若采用P命令进行调试，则只执行INT 21H一条指令，执行完立即返回，给调试者的感觉好象是执行了一条普通指令一样。因此，当读者以后遇到CALL、INT n指令或带重复前缀的数据串操作指令时，若不想观察相应过程的详细执行过程，就可以用P命令。

### 9．显示命令R

格式： （1）R

（2）R <寄存器名>

功能：显示当前所有寄存器内容、状态标志及将要执行的下一指令的地址（即CS**:**IP）、机器指令代码及汇编语句形式。其中对状态标志寄存器FLAG以状态标志位的形式显示，详见表2-1。

|  |  |  |
| --- | --- | --- |
| 表2-1 状态标志显示形式 | | |
| 状态标志位 | 状态 | 显示形式 |
| 溢出标志OF | 有/无 | OV/NV |
| 方向标志DF | 减/增 | DN/UP |
| 中断标志IF | 开/关 | EI/DI |
| 符号标志SF | 负/正 | NG/PL |
| 零标志ZF | 零/非零 | ZR/NZ |
| 奇偶标志PF | 偶/奇 | PE/PO |
| 进位标志CF | 有/无 | CY/NC |
| 辅助进位标志AF | 有/无 | AC/NA |

键入该命令后将显示指定寄存器名及其内容，“：”后可以键入修改内容。键入修改内容后按“ENTER”键有效。若不需修改原来内容，直接按“ENTER”即可。

例如：R AX

### 10．结束DEBUG返回DOS命令Q

格式：Q

功能：返回DOS提示符下。

**四、重新调试实验一的程序ABC.ASM ，验证最终AX寄存器的值是否正确。**

# 实验三 简单算术运算程序设计

## **一、实验目的**

1. 掌握数据传送和算术运算指令的用法。

2．熟悉在PC机上建立、汇编、连接和调试程序的方法。

3．理解计算机中除法溢出的含义。

## **二、实验内容**

1．设W、X、Y、Z均为8位带符号数，要求完成计算表达式**W=X+Y-Z。**

**编程思路（仅供参考）：**

该题目要求掌握运算中带符号数和无符号数运算的区别。本题目只考虑有符号数情况

**参考程序清单:ABC.ASM**

DATA SEGMENT

X DB 1 ;x 单元存放数字1，以下类推

Y DB 4

Z DB 3

W DB ?,? ;预留2个单元，存放运算结果

DATA ENDS

CODE SEGMENT

ASSUME CS:CODE,DS:DATA

START:MOV AX , DATA

MOV DS ,AX

;==========================

**MOV AL , X**

**ADD AL , Y ;x+y**

**SUB AL ,Z ;x+y-z**

**MOV W ,AL 存结果到W单元**

;==========================

;==========================输出结果

**MOV AH ,2 ;调用DOS中断INT 21H功能的2号子功能**

**ADD AL,30h ;个位数字变成数字字符，**

**Mov dl,al**

**INT 21H**  ;此3条语句输出数字的ASCII码

;==========================

MOV AH,4CH

INT 21H

CODE ENDS

END START

**思考：**

1. **如果运算结果大于等于10，显示的结果正确吗？如何解决？**
2. **如果运算结果小于0，显示的结果正确吗？如何解决？**

# 实验四 汇编语言的输入输出

## **一、实验目的**

1. 掌握数据传送和算术运算指令的用法。

2．熟悉字符的输入输出。

3．理解计算机中除法溢出的含义。

二、实验内容

改写**实验三**ABC.ASM，分以下6个问题进行编程

1. 改写实验三ABC.ASM，把其中的X,Y,Z 都置空,代码如下

**X DB ?**

**Y DB ?**

**Z DB ?**

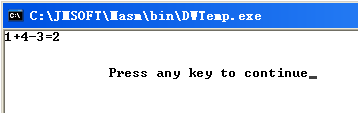
2. 改为从键盘输入X，Y，Z的值，分别为1,4,3，输入单个字符的代码为：

**MOV AH , 1**

**INT 21H**

**;调用INT 21H中的1号功能，输入单个字符，输入的字符的ASCII码值存放在AL寄存器中，分3次输入，然后再进行加、减运算，看看运算结果是否正确？如果不正确，如何解决？**

1. **如果想让输入和输出的结果如下图所示，又该如何解决？**

****

1. **如果想让输入和输出的结果如下图所示，又该如何解决？**

****

1. **在此基础上，把ABC.ASM程序改为下面的表达式，写出代码：**

**W=((X+Y)\*2-Z)/2 ;X,Y,Z,W均为8位二进制数值**

**6．如果X,Y,Z,W均为16位二进制数值，那么如何编写下面的表达式的代码？**

**W=((X\*Y)+5-Z\*5)/2**

# 实验五 分支结构程序设计

**一、实验目的**

1．掌握分支程序的结构及相关指令。

2．掌握分支程序的设计，调试方法。

**二、实验内容**

1． 在BUF和BUF+1、BUF+2单元分别放有一个无符号字节型数，编程序将其中最大数存入MAX单元，并在屏幕上显示（要求画出流程图）。

2． 要求同上，只是比较的数为有符号数。

3．将1000H单元开始的10个字节数，采用奇偶校验，将奇数在前偶数在后仍存回原数据区。本实验要求在DEBUG调试状态下进行,包括汇编程序、运行程序、检查结果。

4．把从键盘输入的大写字母转换成小写字母程序，按ESC键结束输入。

算法分析：

由ASCII码字符表可知，英文大、小写的26个字母的值的差为20H，因此欲将从键盘上输入的大写字母转换为相应的小写字母，只需将从键盘输入的大写字母的ASCII编码值加上20H即可。

5．已知两整数变量A和B，试编制完成下列功能的程序：

①若两数中一个是奇数，则将奇数送入A—BUF单元中，偶数送入B—BUF单元中。

②若两个数均为奇数，则两数分别加1，并存入原变量中。

③若两个数均为偶数，则两变量不变。

算法分析：

设义、B两变量预先存放在数据段的A—BUF和B—BUF的字节单元中，其值分别为X和Y。解答这个题目的关键是如何判断一个数是奇数或是偶数，这只要判断一个数的最低位是0或是1就可以了。若是0，则为偶数，若为1，则为奇数，可以采用TEST指令和条件转移指令JZ来实现，

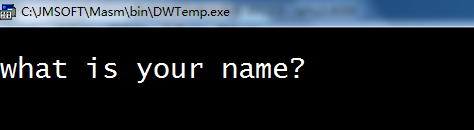
5．在内存中以BUF单元开始缓冲区中连续存放着10个学生的分数，试编制程序统计其中90一100分、80一89分、60一79分及60分以下者各有多少人，并把结果分别存放在S9、S8、S7和S6中。

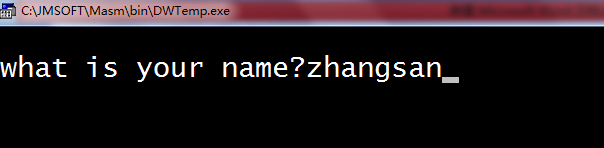
6．　统计一个有10人的班中，男、女生的人数各是多少。将统计的男生人数存入变量MANK中，女生人数存入变量WOMAN中。

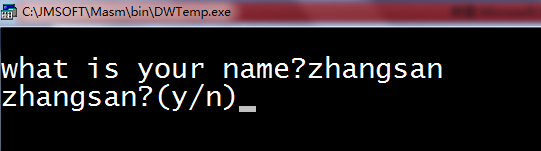
算法分析：

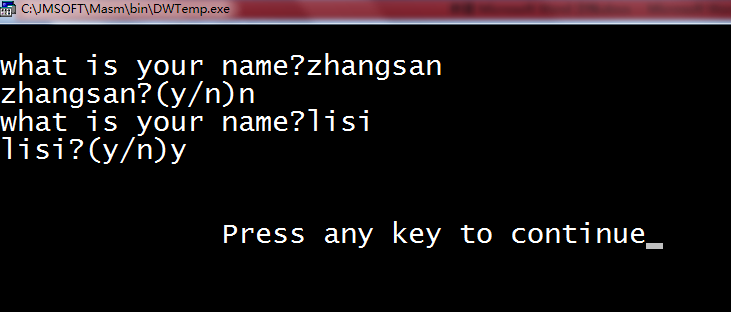
男、女生的标志可用“0”、“1”字符来表示，并从键盘输入此标志。根据题意，总人数已知，在此范围下，对输入标志进行分类求和显然可用循环结构来完成。每循环一次使相应人数增1，而循环计数值减1，直至该计数值为0。可用CX作为循环计数值，用LOOP指令对循环条件进行判定。　

7.编程，要求：（1）用9号功能显示一串字符，如”INPUT YOUR NAME:” （2）在串的尾部输入你的英文名字，至少3个字符，按回车键后，（3）在下一行，重新显示你输入的名字，（4）名字后面，继续显示串“(y/n)?”（5） 输入为字母”y”结束程序，输入字母”n”回到（1）重新输入。









# 实验六 循环结构程序设计

## 一、实验目的

1．掌握循环结构程序设计的方法；

2．掌握数据块传送程序设计的方法；

3．掌握循环指令的应用。

## 二、实验内容

1. 编写程序实现将数据段STR1单元开始的20个字符移到STR2开始的存储区。假设STR1+20>STR2，可以用串传送指令，也可以用MOV传送指令。

编程思路：

该题目要求掌握单循环程序设计方法。为了实现指定功能，应从以下几个方面考虑：

（1）数据串操作的起始位置从高地址还是从低地址开始。

（2）确定使用串传送指令，还是用MOV指令。

1. 确定循环程序的结构。
2. 编制程序计算S=1+2+3+4+……+N直到和大于500为止，并将结果由屏幕上显示出来（N的值和最终的和的值）。
3. 3000H内存单元开始的100个字节存储单元全部清0。 本实验要求在DEBUG调试状态下进行,包括汇编程序、运行程序、检查结果。
4. 编程：输入一行字符，分别统计出其中英文字母、数字和其他字符的个数，并显示各种统计结果。
5. 动态从键盘输入20个二位无符号数，找出其中最大数并显示。
6. 动态从键盘输入20个二位无符号数，从小到大排序并显示。
7. 设数据区有两个字节串，串1的长度为5，串2的长度为10。编程： 若串2的最后5个字符和串1相同，则置FLAG单元为“Y”**，否则置为“N”。**
8. **编写打印如下图形程序**

0

1 0

2 1 0

3 2 1 0

4 3 2 1 0

5 4 3 2 1 0

6 5 4 3 2 1 0

7 6 5 4 3 2 1 0

8 7 6 5 4 3 2 1 0

9 8 7 6 5 4 3 2 1 0

**实验七 子程序设计**

## 一、实验目的

1．掌握子程序的定义和调用及参数传递的方法。

2． 掌握子程序、子程序的嵌套、递归子程序的结构。

3． 掌握子程序的设计、编制及调试。

## 二、实验内容

* 1. 对缓冲区中的一组字符的ASCII码加偶校验。要求画出流程图，编写程序。

算法分析：每个字符的ASCII码为7位，占一个字节，其最高位为校验位。偶校验的含义为：校验位与7位字符位中的1的个数之和为偶数，这可用于字符正确与否的校验。例如：字符为1011011，其中1的个数为5，那么校验位填1，这样1的个数为6，即为偶校验。偶校验位的填入可作为子程序来设计，缓冲区的首址一般由主程序提供，可假设缓冲区中的字符个数由缓冲区中的第一个字节给出。

* 1. 用子程序调用实现在屏幕上显示10行相同的字符，要求显示字符串功能用子程序调用方法实现。
  2. 试编制计算N! （N≥0）的程序。N!=N\*（N-1）\*（N-2）\*……\*1，其递归定义如下： 0!=1，

**N!=N\*（N-1）! （N＞1）**

* 1. 用子程序结构编写程序：从键盘输入一个二位十进制数的月份数（01～12），然后显示相应的英文缩写名。
     1. 编写一个子程序嵌套结构的程序模块，分别从键盘输入姓名及8个字符的电话号码，并以一定的格式显示出来。

**具体要求：**

**主程序TELIST中包括以下功能**

1. 显示提示符INPUT\_NAME:;
2. 调用子程序INPUT\_NAME输入姓名；
3. 显示提示符INPUT TELEPHONE NUMBER:;
4. 调用子程序Input\_PHONE输入电话号码；
5. 调用子程序PRINTLINE显示姓名及电话号码。

**子程序INPUT\_NAME中包括以下功能**

1. 调用键盘输入子程序GETCHAR，把输入的姓名存放在INBUF缓冲区；
2. 把INBUF中的姓名移入输出行OUTNAME。

**子程序Input\_PHONE中包括以下功能**

1. 调用键盘输入子程序GETCHAR，把输入的8位号码存放在INBUF缓冲区；
2. 把INBUF中的号码移入输出行OUTPHONE。

**执行子程序PRINTLINE输出效果如下：**

显示姓名及电话号码，格式为：

NAME TEL

ＸＸＸ　　　　　ＸＸＸ

**实验八 显示程序设计**

## **一、实验目的**

1． 掌握软件延时程序的编写方法。

2． 熟悉系统功能调用的方法。

3． 掌握宏定义与宏调用的方法。

## **二、实验内容**

1． 编写在屏幕上每隔一秒连续显示字符0－9的程序。

　编程提示：

　为了使输出的字符之间有空格，在每一个循环中输出一个0－9的字符和一个空格。为了输出0－9，置寄存器BL初值为0，每循环一次使其增量。为了保证输出是十进制，增量后用DAA调整指令。由于输出为一位十进制数，用AND 0FH指令屏蔽高4位。

1. 用系统功能调用将‘HELLO’由键盘输入，再从屏幕输出‘\*\*HELLO\*\*’。
2. 编程实现：每隔5.5S从键盘读一个字符，并将字符的ASCII码存入缓冲区，共读50个字符，利用INT 16H的0功能读取字符，返回值在AL中。
3. 利用自编中断INT 60H 实现在屏幕上显示10行相同字符。显示功能用INT 60H实现。

**实验九　综合实验**

1. **实验题目：**

显示学生名次表

1. **实验目的：**

进行汇编语言程序设计应用的综合性训练。综合运用汇编语言循环程序和子程序编程方法，巩固提高汇编语言程序设计能力应用水平。

1. **实验内容**

**实验内容：** 编制一程序，要求接收从键盘输入的一个班的学生的学号、姓名、成绩，对学生的成绩进行排序，最后把学号、姓名、学生成绩、名次显示出来。

**实验要求：**1.**必须画流程图**。2.**本程序涉及汇编语言的多重循环、子程序及宏**，其中 成绩输入、计算学生名次、显示学生名次都分别用子程序，也可用宏处理。**3.要求用菜单选择**。具体要求：

**输入1**是数据录入并输出（包括姓名、学号、成绩）；

**输入2**是排序后的数据输出（包括姓名、学号、成绩、名次）；

**输入3是可以按学号查找某学生，并输出其成绩，选择修改或不改成绩（选做）**

1. **考核标准：**

**1 程序算法设计正确，流程图准确规范**

**2 源程序结构合理、注释清晰、可读性好**

**3 实验运行结果正确，达到实验要求**

**4 实验报告格式规范、内容充实、文字流畅**

**5 实验分析有自己的见解、有思考、有结论**

模拟试题一

1. **单选题**

1．有一16个字单元的数据区，它的起始地址为4000：1000H，问这个数据区首字、末

字单元的物理地址各是（ ）。   
A. 首字单元的物理地址为5000H， 末字单元的物理地址为 501EH。

B. 首字单元的物理地址为41000H，末字单元的物理地址为 41032H。

C. 首字单元的物理地址为41000H， 末字单元的物理地址为 4101EH。

D. 首字单元的物理地址为41000H， 末字单元的物理地址为 41016H。

2．在机器数中，（ ）对零的表示形式是唯一的。   
A. 补码 B.原码 C. 反码 D. 没有正确答案

3. 将十进制数58转换成压缩的BCD码的结果是（ ）。

A．0508H B. 58H C. 85H D. 5080H

4. 将十进制数 -1 转换成8位二进制数的补码是（ ）。

A． 11H B. 01H C. FFFFH D. 0FFH

5. 执行指令POP CX后，堆栈指针SP自动（ ）。

A．-2 B. +1 C. -1 D. + 2

6. 在标志寄存器中表示运算有进/借位的标志位是（ ）。

A． AF B. OF C. SF D. CF

7. 表示下一条要执行指令的物理地址 ( )。  
 A.　 DS:[SI] B. ES:[DI] C. CS:IP D. IP   
8. 如果80X86 CPU的地址线为20根，则它的内存寻址空间大小为（ ）。

A． 1K B. 1 M C. 16 M D. 32 M

9. 设物理地址（10FF0H）=70H，（10FF1H）=80H，（10FF2H）=90H，则从地址10FF1H单元中取出一个字的内容是（ ）。

A．80H B. 8090H C. 7080H D. 9080H

10. 串操作指令中，目的操作数的段地址一定在（ ）寄存器中

A. ES B. SS C. DS D. CS

11. 在8086标志寄存器中，ZF=1表示（ ）。

A． 结果有进位 B. 运算结果不为0 C. 运算结果为0 D. 结果溢出

12. 在8086计算机的内存分段管理中，最小段的大小为（ ）。

A．256B B. 64K C. 128K D. 16B

13. 指令MOV AX， TAB[SI]，源操作数的寻址方式是（ ）。

A．基址变址寻址 B. 相对寻址 C. 间接寻址 D. 直接寻址

14. 指令MOV BX，TAB+5 ，源操作数的寻址方式是（ ）。

A．直接寻址 B. 间接寻址 C. 立即数寻址 D. 相对寻址

15. 若(BX)=1234H，(SI)=1800H，执行MOV [SI], BX指令后，下面（ ）是正确的。

A．1800单元的值为12H B. SI 寄存器的值为1234H

C. 1801单元的值为34H D. 1801H单元的值为12H

16. 将8位有符号数B6H扩展为字的结果是（ ）。

A．11B6H B. FFB6H C. 00B6H D. 10B6H

17. 一条指令中，目的操作数不允许使用的寻址方式是（ ）。

A．寄存器寻址 B. 直接寻址 C.立即寻址 D. 基址变址寻址

18. X已经定义为字节型，可以用（ ）改变为字型。

A．WORD PTR X B. LABEL X

C. BYTE PTR X D. OFFSET X

1. 下列哪个指令可以把字节扩展为字（ ）。

A． CBW B. CWD C. CLD D. CTD

20. Z DW 1，5， $+7，其中地址计数器 $ 所在存储单元的值为（ ）。

A．3 B. 11 C. 4 D. 7

21. 当一个带符号数大于FFH时程序转移，满足条件的是（ ）。

A．负数和0 B. 负数

C. 正数和0 D. 大于255的数

22. 在执行条件转移指令中，不能形成条件转移的指令是（ ）。

A．AND B. MOV C. CMP D. TEST

23. 汇编语言中存储单元属性不能是（ ）。

A. 双字型 B．字节型 C. 字型 D. 字符CHAR型

24.用指令MOV　AX，OFFSET　BUFF指令，可得到存储单元BUFF的（ ）。

A．偏移地址 B. 段基址 C. 物理地址 D. 类型

25.若BL=84H，CF=1，则ADC BL，90H执行后，BL=（ ）。

A. 174H B．15H C. 115H D. 175H

26.带符号数除以2操作可用（ ）指令实现。

A．SHL AL, 1 B. SAR AL, 1 C. RCR AL, 1 D. RCL AL, 1

27. 下面哪个标志位为1 时允许产生中断（ ）。

A．DF=1 B. IF=0 C. TF=1 D. IF=1

28.获得TABLE单元偏移地址的汇编指令为（ ）。

A. LEA BX，TABLE B．MOV BX，TABLE

C. MOV BX，SEG TABLE D. LDS BX，TABLE

29. 下列描述错误的是（ ）。

A．LOOP指令中CX为循环控制计数器

B. LOOPE指令能够循环的条件是CX≠0且ZF=0

C. LOOPNE指令能够循环的条件是CX≠0且ZF=0

D. LOOPE指令能够循环的条件是CX≠0且ZF=1

30. 在子程序调用过程中，断点指的是（ ）。

A．CALL指令本身 B. 子程序名

C. CALL的下一条指令 D. CALL的下一条指令的地址

1. CALL指令和JMP指令的区别是（ ）。

A．CALL指令使程序转移 B. 子程序名可以是标号

C. CALL指令可以段间调用 D. CALL指令将断点地址保存

1. 用CALL指令调用子程序时，从子程序返回到主程序（ ）。

A．只能用RET指令 B. 只能用JMP指令

C. 用INT 21H D. 既可用RET又可用JMP

1. 宏定义时，是通过（ ）实现参数传递的。

A．寄存器 B.哑元和实元

C.堆栈 D. 存储单元

1. 宏库可以保存多个宏，在程序中用（ ）伪指令打开宏库。

A．MACRO B. SEGMENT C. PROC D. INCLUDE

1. 有关宏的作用，下列说法不正确的是（ ）。

A．宏定义体中不可以有标号 B. 宏调用时不用保存断点

C. 宏可以被多次调用 D. 宏展开是在汇编过程中完成的

1. 在串传送指令中，串的传送方向是由（ ）标志位决定的。

A．DF B. CF C. ZF D. OF

1. 在多重循环程序中，从外循环再次进入内循环时，内循环的计数值（ ）。

A．不必考虑 B. 置1 C. 置0 D.重新赋值

38. 用户程序中的数据段的段地址，用（ ）指令给出。

A． MOV AX，CODE B. MOV AX，DATA

MOV CS，AX MOV DS，AX

C. MOV CS，DATA D. MOV CS，CODE

MOV DS，CS MOV DS，CS

1. 执行CLD指令后，每执行一次串传送指令MOVSD，源串和目标串的**（ ）**。
2. 偏移地址+2 B. 偏移地址-2 C. 偏移地址-4 D. 偏移地址+4

40. 关于字节乘法指令错误的说法是**（ ）**。

A．被乘数隐含在AL中 B. 乘积是字型

C. 被乘数隐含在AX中 D. 乘数和被乘数都是字节型

**二、简答题**  
**1. 80X86微机的中断系统主要分几类，各是什么？**

**2. TEST 和 AND 指令有区别吗，为什么？**

**3．写出求1+2+3…..+100 的累加和的关键代码，不要求输入和输出**

**4. 简述REP MOVSB指令都完成了哪些功能。**

**5. 简述CMP 指令和sub指令的区别。**

**6. 写出将从键盘输入的小写字母转换为大写字母并输出的关键代码。**

**7.宏和子程序的优缺点**

**8．PC/XT机采用向量中断方式处理级外中断，中断类型号依次为08H~0FH。在RAM 中断向量表的 0：34H单元开始依次存放34H，F2H，01H和F0H四个字节，问该向量对应的中断类型号是多少？中断程序入口地址是多少？**

**9．用相对寻址方式，将存储单元X中的第5个字取出（为源操作数）后，与AX相减，结果送Y字单元。（写出关键代码，不要求输入和输出）**

**10 . 实模式下存储器分段后，最大的段和最小的段各是多少，有多少根地址线，可寻址的地址空间是多少？**

**三、程序分析题**  
1． **TAB1　DW 　1234H   
 TAB2 　DW　5678H   
 　 ┊   
 PUSH 　TAB1   
 PUSH　 TAB2   
　　POP　 AX   
　　POP　 BX** **试回答：①上述程序段执行后（AH）＝＿＿＿  
 ②设执行前SP＝2A00H，执行POP AX 指令后，SP＝＿＿＿＿＿**   
**2．程序段如下：**   
 **TAB　 DW　 38, 48, 58, 68, 78**

**NUM DW 3  
………**

**MOV　 BX , OFFSET TAB   
ADD BX , NUM  
MOV AX , [BX]**   
**请回答程序段运行后：（AX）＝＿＿＿＿＿＿**

**3. 假设下面程序的附加段和数据段已加载，分析下面程序段完成什么功能？**

**LEA SI , BUF1**

**LEA DI , BUF2+5**

**MOV CX , 5**

**REPE CMPSB**

**JE P2**

**MOV DL , 'N'**

**JMP E2**

**P2: MOV DL , 'Y'**

**E2: mov ah,4ch**

**Int 21h**

**4. 现有下列程序段 ，问下述程序段运行后， （AL）＝ ＿＿＿＿** **MOV CL , 2**

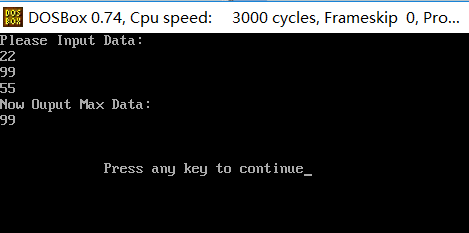
**MOV AL , 0A8H**

**SAR AL , 1**

**ROL AL , CL**

**四、判断下列语句是否合法，正确的打√， 错误的打× 。**

1. **MUL 10 （ ）**
2. **ADD BL , 266 （ ）**
3. **AND [BX] , AX （ ）**
4. **MOV CS，AX （ ）**
5. **IN AL, 62H （ ）**
6. **PUSH BL （ ）**
7. **LEA BX ,TAB ( ）**
8. **RCR AX , 4 ( ）**
9. **XCHG [BX], BUF （ ）**
10. **CMP CX, AX （ ）**

**五、程序改错（不得改变算法）：**

**程序完成的功能是从键盘输入3个2位10进制数并顺序存入内存单元，找出其中的最大值并输出最大值。下面代码有10处错误，划线位置都是有错误之处，请在相应位置修改或写出正确代码，每处错误只有一行代码。**

**程序正确输入和输出界面如右图所示：**

**DATAS SEGMENT**

**buf1 db "Please Input Data:",13,10,"$"**

**buf2 db "Now Ouput Max Data:",13,10,"$"**

**buf3 db 13,10 ;-------------(1)---------------------**

**X DB 3 (?) ;预留3个单元 ; ----(2)---------------------**

**Y DB ?**

**Z DB 10**

**max db 0**

**DATAS ENDS**

**dsps macro buf ;定义宏，输出串**

**mov ah,9**

**mov dx, buf ; --------------(3)-------------------**

**int 21h**

**; --------------(4)-------------------**

**CODES SEGMENT**

**ASSUME CS:CODES,DS:DATAS**

**START:**

**MOV AX,DATAS**

**MOV DS,AX**

**BEG:dsps buf1 ;调用宏，显示串**

**MOV CH,3 ;--------------(5)------------------**

**MOV BX,0**

**LP1:MOV AH,1 ;输入3个2位10进制数**

**INT 21H**

**SUB AL,30H**

**MOV AH,0**

**MUL Z**

**MOV Y,AL**

**MOV AH,1**

**INT 21H**

**SUB AL,30H**

**add al,y**

**MOV X[BX],AL ;存储数据到内存**

**ADD BX,2 ;----------------(6)--------------------**

**dsps buf3 ;回车换行**

**LOOP LP1**

**call max1 ;调用求最大值子程序**

**dsps buf2 ;显示一个串，然后输出最大值**

**CBW**

**mov al,max**

**idiv 10 ;----------------(7)------------------**

**mov bx,ax**

**add bx,3030h**

**mov ah,2**

**mov dl,bl**

**int 21h**

**mov ah,2**

**mov dl,bh**

**int 21h**

**MOV AH,4CH ;结束程序**

**INT 21H**

**max1 proc ;求最大值子程序**

**MOV CX,3**

**mov bx,0**

**lp2:mov al,x[bx]**

**cmp al,max**

**jge lp3 ;----------------(8)--------------------**

**mov max,al**

**lp3:inc bx**

**loop lp2**

**\_ \_\_\_\_\_ ;----------------(9)--------------------**

**max1 endp**

**CODES ENDS**

**END BEG ;----------------(10)--------------------**