

```

mov dx,5678h
mov ax,1234h
mov cl,4
shl dx,cl
mov bx,ax
and bx,0f000h
mov cl,12
shr bx,cl
add dl,bl
mov cl,4
shl ax,cl

```

```

mov dx,5678h
mov ax,1234h
mov cl,4
shr ax,cl
mov bx,dx
and bx,000fh
mov cl,12
shl bx,cl
add ax,bx
mov cl,4
    shr dx,cl

```

```

da_1 db 8
da_2 db 9
da_3 db 10
input db 1
jsj_10 dw ?
mov al,input
mul da_1
add al,da_2
sub al,da_3
mov jsj_10,ax

```

```

mov ah,1 MOV ES,AX
cld
mov si,offset STRING1
mov di,offset STRING2
mov cx,count
repz cmpsb;相等就重复下去
jz equal
mov dx,offset mess2
mov ah,9
int 21h
    MOV AH,4CH

```

```

INT 21H
equal:
mov dx,offset mess1
mov ah,9
INT 21H

```

```

mov ah,1
int 21h
mov DL,AL
.IF AL=='Y'
MOV DL,'6'
MOV AH,2
INT 21H
.ELSEIF AL=='N'
MOV DL,'8'
MOV AH,2
INT 21H
.ENDIF
int 21h
mov DL,AL
OR DL,00100000B
mov ah,2
int 21h

```

```

mov STRING1,'H'
mov STRING2,'E'
MOV AL,STRING1
.IF AL==STRING2
MOV TAG,0
.ELSE
MOV TAG,1
.ENDIF

```

```

MOV CX,COUNT
a:
MOV SI,CX
.IF STRING[SI]=='&'
MOV STRING[SI],' '
.ENDIF
    LOOP A

```

```

MOV ES,AX
cld
mov cx,9
mov di,offset STUDENT_ADDR
mov al,'-';-在 DI-1 ZF=1

```

```
repne scasb
```

```

MOV ES,AX
std
mov cx,9
mov di,offset STUDENT_ADDR+9
mov al,'-';-在 DI+1 ZF=1
repne scasb
MOV ES,AX
mov cx,132
lea bx,PRINT_LINE
a:
mov bx,cx
dec bx
    mov PRINT_LINE[bx],' '
    loop a

```

```

mov cx,5
xor ax,ax
mov bx,0
a:
add ax,array[bx]
add bx,2
loop a

```

```

mov cx,7
a:
mov bx,cx
dec bx
mov dl,aa_1[bx]
add dl,2
push dx
mov bb_1[bx],dl
loop a
mov cx,7
b:
pop dx
add dx,30h
mov ah,2
int 21h
loop b

```

```

xor ax,ax
mov cx,len
a:
mov bx,cx

```

```

dec bx                                loop a                                INT 21H
.if string1[bx]>=61h    &&    cmp ax,0                                t:
string1[bx]<=7ah        jnz t                                mov cl,0
mov ax,1                mov cl,1
.endif                    MOV AH,4CH

```

```

a=Number(prompt("请输入行数"));
a=a>10?10:a;
b=Math.pow(a,2);
for(var i=1;i<=b;i++)
{
    document.write("#");
    if(i%a==0) document.write("<br>");
}
var count=0;
for(i=2;i<=100;i++)
{
    tag=0;
for(j=i-1;j>1;j--)
    if(i%j==0) tag=1;
    if(tag==0) {
        count=count+1;
        document.write(i+'\\t');
        if(count%5==0) document.write("<br>");
    }
}
for(var i=100;i<=999;i++)
{
    t=i.toString()
    var
a=Number(t[0]),b=Number(t[1]),c=Number(t[2]);
if(Math.pow(a,3)+Math.pow(b,3)+Math.pow(c,3)==i)
    document.write(i+' ');
}
switch (new Date().getDay()) {
case 0:
    day = "sunday";
    break;
case 1:
    day = "monday";
    break;
case 6:
    day = "saturday";
}
alert(day)

```

1.一个双字数据存放在DX和AX寄存器中,DX为高字,AX为低字。编程实现把这个双字左移四个二进制位,低位补0。

2.假设数据段定义如下,一个双字数据存放在DX和AX寄存器中,编程实现把这个双字右移四个二进制位,高位补0。

3.求表达式 $Y=A*X+B-C$ 的值,已知A、B、C分别为存放在DA_1、DA_2、DA_3字节单元的无符号数,它们的值分别是:8、9、10,X存放在INPUT字节单元,把结果Y放入JSJ_10字单元。

4.从键盘输入一个大写字母,用小写字母显示出来。

5.编写汇编程序,实现如下功能:从键盘接收一个小写字母,然后找出它的后续字符并显示。

6.编写汇编程序,实现如下功能:从键盘接收一个数字,然后找出它的后续数字并显示。

7.编写汇编程序,实现如下功能:在首地址为STRING1的存储区中以字节方式存储字符H,在首地址为STRING2的存储区中以字节方式存储字符E,比较两个字符是否相等,相等给字节变量TAG送数据0,不相等给字节变量TAG送数据1。

8.假设数据段定义如下:

```
CONAME DB 'SPACE EXPLORERS INC.'
```

```
PRLINE DB 20 DUP ('')
```

用串指令,从右到左把CONAME中的字符串传送到PRLINE。

9.在数据段中有一个字符串STRING,其长度为17,请编写程序,要求把他们传送到附加段中的一个缓冲区STRING2。

10.假设数据段中数据定义如下:

```
STRING DB 'Visual&C++'
```

```
COUNT EQU $-STRING
```

编写程序段,把字符串STRING中的'&'字符用空格符代替。

11.假设数据段中数据定义如下:

```
STUDENT_NAME DB 30 DUP (?)
```

```
STUDENT_ADDR DB 9 DUP (?)
```

```
PRINT_LINE DB 132 DUP (?)
```

编写程序,在STUDENT_ADDR中查找第一个'-'。

12.假设数据段中数据定义如下:

```
STUDENT_NAME DB 30 DUP (?)
```

```
STUDENT_ADDR DB 9 DUP (?)
PRINT_LINE DB 132 DUP (?)
```

编写程序，在 STUDENT_ADDR 中查找最后一个 '-'。

13.假设数据段中数据定义如下：

```
STUDENT_NAME DB 30 DUP (?)
STUDENT_ADDR DB 9 DUP (?)
PRINT_LINE DB 132 DUP (?)
```

编写程序，用空格符清除 PRINT_LINE 域。

14.假设数据段定义如下：

```
DATA SEGMENT
STRING1 DB 'Visual C++ '
STRING2 DB 'Visual Basic'
COUNT EQU $-STRING2 ;计算串长度
MESS1 DB 'MATCH!',13,10,'$'
MESS2 DB 'NO MATCH!',13,10,'$'
DATA ENDS
```

试编写一程序，要求比较两个字符串 STRING1 和 STRING2 所含字符是否相同，若相同则显示'MATCH'，若不相同则显示'NO MATCH'。

15.编写汇编程序，实现如下功能：从键盘接收一个字符，如果输入字符为 Y，则显示字符 6，如果输入字符为 N，则显示字符 8。

16.编写汇编程序，实现如下功能：在首地址为 ARRAY 的存储区中依次存储 5 个字数据 23, 36, 2, 10, 7, 求出该数组的内容之和 (不考虑溢出)，并把结果存入字变量 TOTAL 中。

17.假设数据段定义如下：

```
DATAS SEGMENT
X DD 12345678H
Y DD 87654321H
DATAS ENDS
```

假设 X 和 Y 都为无符号数，请设计一个程序，如果 $X*2 > Y$ ，则使 (AX) = 1 后退出，如果 $X*2 \leq Y$ ，则使 (AX) = 2 后退出。

18.假设数据段定义如下，把首地址为 AA_1 的数组中每个数加 2 后再存入到数据段中 BB_1 数组开始的标号地址中去，并显示出 BB_1 数组中每个数 (之间用空格分开)

```
DATA SEGMENT
AA_1 DB 1,3,5,7,2,4,6
BB_1 DB 7 DUP(?)
COUNT DW 7
DATA ENDS
```

19.假设数据段定义如下，请编写程序判断字符串 STRING 是否存在小写字母，如果有，把 CL 的第 0 位置 0，否则置 1。

```
DATAS SEGMENT
```

```
STRING1 DB '123879587,/=9-0ADWUIRUWR','$'
LEN EQU $-STRING1
```

```
DATAS ENDS
```

20.假设数据段定义如下，DATA X 和 DATA Y 均为带符号数，从键盘上接收加 (+)、减 (-)、乘 (*) 符号，然后对 DATA X 和 DATA Y 完成相应运算，结果存入 RLT 单元。

```
DATAS SEGMENT
DATA X DW 10
DATA Y DB 2
RLT DW ?
DATAS ENDS
```

21.假设数据段定义如下，编写一个子程序，实现求一个无符号数的三次方的功能。并调用该子程序，求数据段中数据的三次方。

```
DATAS SEGMENT
DATA DB 3
DATAS ENDS
```

22.假设数据段定义如下，内存自 BUF 单元开始的缓冲区连续存放着 10 个学生的英文分数，编程序统计其中 90~100, 60~89, 60 以下者各有多少人?并把 90-100 分人数结果存放在 P1 单元中、60-89 分人数结果存放在 P2 单元中、60 分以下人数结果存放在 P3 单元中 (P1、P2、P3 为数据段的单元)。

```
DATA SEGMENT
BUF DB
70H,58H,50H,91H,99H,62H,75H,82H,74H,60H
ORG 0020H
P1 DB 0
ORG 0030H
P2 DB 0
ORG 0040H
P3 DB 0
ORG 0050H
COUNT DW 10
DATA ENDS
```

23.假设数据段定义如下：

```
DATA SEGMENT
ARRAY DW 1,2,-3,-4,0,0,5,6,-7,-8,-9,10
COUNT EQU $-ARRAY
DATA ENDS
```

要求编写程序，测试其中正数、0 及负数的个数。正数的个数放在 DI 中，0 的个数放在 SI 中，负数的个数放在 AX 中。