FIND SMALLEST NUMBER

```
assume cs:code, ds: data
code segment
start : mov ax, data
        mov ds, ax
        mov si, 0000h
        mov cl, 04h
        mov al, [si]
   up:    inc si
        mov bl, [si]
        cmp al, bl
        JC down
        xchg al,bl
 down: dec cl
        JNZ up
        mov [si+1], al
code ends
data segment
num db 12h, 07h, 25h, 18h, 02h
data ends
end
```

FIND LARGEST NUMBER

```
Assume cs:code, ds: data
Code segment
Start : movax, data
        Mov ds, ax
        Movsi, 0000h
        Mov cl, 05h
        Mov al, [si]
  Up:   incsi
        Movbl,[si]
        Cmp al, bl
        JNC down
        Xchgal,bl
 Down: dec cl
        JNZ up
        Mov [si+1], al
Code ends
Data segment
Numdb 03h, 05h, 02h, 06h, 01h, 07h
Data ends
End
```

WAP TO ARRANGE THE NOS IN ASCENDING ORDER

```
assume cs:code,ds:data
code segment
start:mov ax,data
      mov ds,ax
      mov si,0000h
      mov ax,0000h
      mov di,si
      mov cl,05h
prachi:mov ch,cl
       mov al,[si]
up:    inc si
       cmp al,[si]
       jc down
       xchg al,[si]
down: dec ch
      JNZ up
      mov [di],al
      inc di
      mov si,di
      dec cl
     jnz prachi
code ends
data segment
     num db 03h,02h,01h,04h,05h,06h
data ends
end
```

WAP TO ARRANGE THE NOS IN DESCENDING ORDER

```
assume cs:code,ds:data
code segment
start:mov ax,data
      mov ds,ax
      mov si,0000h
      mov ax,0000h
      mov di,si
      mov cl,05h
prachi:mov ch,cl
       mov al,[si]
up:    inc si
       cmp al,[si]
       jnc down
       xchg al,[si]
down: dec ch
      JNZ up
      mov [di],al
      inc di
      mov si,di
      dec cl
     jnz prachi
code ends
data segment
     num db 03h,02h,01h,04h,05h,06h
data ends
end
```

EVEN NUMBER

```
MOV assume cs:code, ds: data
Code segment
Start : mov ax, data
Mov ds, ax
Mov si, 0000h
Mov di, 1000h
Mov ax, 0000h
Mov cl, 0ah
Up: mov al, [si]
Ror al, 01h
Jnc down
inc ah
Rol al, 01h
Mov [di], al
Inc di
Down: inc si
Dec cl
Jnc Up
Mov [si+1],ah
Code ends
Data segment
Num db 01h,02h,03h,04h,05h,06h,07h,08h,09h,0ah
Data
```

ODD NUMBER

```
assume cs:code, ds: data
code segment
start : mov ax, data
mov ds, ax
mov si, 0000h
Mov di, 1000h
mov ax, 0000h
Mov cl, 0ah
Up: mov al, [si]
Ror al, 01h
Jnc down
Inc ah
Rol al, 01h
Mov [di], al
Inc di
Down: inc si
Dec cl
Jnc Up
Mov [si+1],ah
Code ends
Data segment
Num db 01h,02h,03h,04h,05h,06h,07h,08h,09h,0ah
Data ends
End
```

(Space for answers)

Program Code with comments

```
Assume cs: code, ds: data
code segment
mov Ax, data
mov ds, AX
mov SI, 0000H
mov CL, 09 H
mov AX, 0000H
mov AL, [SI]
up: INC SI
mov BL, [SI]
Add AL, BL
JNC down
INC ah
down: dec CL
JNC up
mov [sit I], AX
code ends
data segment
num db 01h, 01h, 01h, 01h, 01h, 01h, dh, 01h, 01h, 01h
data ends.
end.
```

XV Results (Output of the Program)

BLOCK TRANSFER OPERATION

```
assume cs:code,ds:data
code segment
start:mov ax,data
mov ds,ax
mov si,0000h
mov di,1200h
mov cl,0ah
up:mov al,[si]
mov [di],al
inc si
inc di
dec cl
jnz up
code ends
data segment
num db 01h,02h,03h,04h,05h,06h,07h, 08h,09h,0ah
data ends
end
```