**##Program to transfer a block of data without overlap from array 1 to array 2( 5 elements )**

.data

Array1 db 1,2,3,4,5

array2 db 10 dup(0)

.code

mov ax,@data ;Initialize the data segment

mov ds,ax

lea si,array1 ;Load offset address of array1 in si

lea di,array2 ;Load offset address of array2 in di

mov cx,05 ;load cx with count

up:mov al,[si] ;Move the 1st element of array1 to al

mov [di],al ;Move to array2

inc si ;Increment si

inc di ;Increment di

dec cx ;decrement the counter

jnz up ;Repeat till cx becomes zero

mov ah,4ch int 21h

end

## **To find the Largest/smallest number in an array of 16 bit numbers**

.data

array dw 1111h,2222h,8567h,4589h,5555h

res dw 02 dup(0)

.code

mov ax,@data

mov ds,ax ;Initialize Data segment

lea bx, array ;Load BX with EA of array

mov cx,05h ;cx=number of elements in the array

mov ax,[bx] ;move the 1st word of the array to ax

up: cmp ax,[bx+2] ;compare the 1st and 2nd elementjnc below ;if 1st number>2nd number compare 1st with ;3rd number and so on.;

Jc below ;for smallest number

mov ax,[bx+2]below:

inc bx

inc bx

dec cx

jnz up ;if cx≠0, repeat

mov res,ax ;store the largest number in location res

mov ah,4ch ;terminate the program

int 21h

end

## **Program to find the factorial of a given number(no<=8)**

.model small

.data

no dw 08h

res dw 02 dup(0)

.code

mov ax,@data ;initialize DS

mov ds,ax

mov ax,no ;move the number to ax and bx

mov bx,no

cmp ax,02h ;compare with 02h

jz down ;if no=2,factorial=2,jump to down

up:

dec bx ;else decrement bx

mul bx ;multiply ax and bx

cmp bx,01h ;compare bx with 01

jz down ;if 0, store the result

jmp up

down:

mov res,dx

mov res+2,ax

mov ah,4ch ;end of the program

int 21h

end

**## Program to check whether the given data byte is positive or negative**

.model small

.data

n1 db 09H

n2 db 0ah,0dh,"the data is positive $"

n3 db 0ah,0dh,"the data is negative $"

.code

mov ax,@data ;Initialize data segment

mov ds,ax

clc ;Clear carry

mov al,n1

and al,10h ;Check the MSB

jz d1 ;If 0, display the no is positive

mov dx,offset n3

mov ah,09h int 21h

jmp end1

d1: mov dx,offset n2 ;Else display the no is negative

mov ah,09h int 21h

end1: mov ah,4ch int 21h

end

##**Program to convert a hex number to ascii number**

.model small

.data

num db 35h;

num db 0AAh

res db 2 dup(0)

.code

Mov ax,@data

Mov ds,ax ; Initialize data segment

Mov al,num ;move the number to al

And al,0f0h ;mask the lower nibble of the digit

Mov cl,04h

Rol al,cl ;bring the digit to LSB position

Cmp al,0ah ;if the number is < 0ah,jump to D1

Jc d1

Add al,07h ;if the number is>0ah, add 37h

D1:

add al,30h

Lea si,res ;store the result in memory

Mov [si],al

Mov al,num

And al,0fh ;mask the upper nibble

Cmp al,0ah ;compare with 0ah

Jc d2 ;if the number is<0ah, jump to D2

Add al,07h ;if the number is >0ah, add 37h

D2:

add al,30h

Mov [si+1],al ;store the result in memory

Mov ah,4ch ;end of the program

Int 21h

End