.MODEL SMALL

.STACK 100H

.DATA

THK db 0Ah, 0Dh,"ENTER A 16-BIT NUMBER: $"

THK1 db 0Ah, 0Dh,0Ah, 0Dh,"THE NUMBER IS: $"

result dw 0

.CODE

MAIN PROC FAR

MOV AX, @DATA

MOV DS, AX

MOV AX, 0

MOV ES, AX ;<Setting the EXTRA SEGMENT REGISTER to 0.>

lea si,result ;<Loading Effective Address of "result" variable to si>

mov word ptr es:[65\*4], isr0x65 ;<Setting our custom Interrupt Service Routine>

mov word ptr es:[65\*4+2], cs

mov ah, 1 ;<Calling our own Interrupt#65 Service#0x1>

int 65

mov ah, 2 ;<Calling our own Interrupt#65 Service#0x2>

int 65

jmp end1

MAIN ENDP

isr0x65 proc

cmp ah, 1 ;<Condition for calling Service# 01 of Interrupt# 65>

je Serv01

cmp ah, 2 ;<Condition for calling Service# 02 of Interrupt# 65>

je Serv02

jmp end

;<Service 1 prompts the user to enter a 16-bit number, reads in 4 digits.>

Serv01: ;<Defines the beginning of the First Service routine of the ineterrupt# 65.>

mov cx, 4h

lea dx, THK

mov ah, 0x9

int 0x21 ;<Raise interrupt 21h. This call the MS-DOS service to print the "msg" string.>

l1:

mov ah, 1

int 21h

mov ah, 0

sub ax, 0x30

push ax

loop l1

mov bx, 0

mov cx, 0x10

pop ax

add bx, ax

pop ax

mul cx

add bx, ax

pop ax

mov cx, 0x100

mul cx

add bx, ax

pop ax

mov cx, 0x1000

mul cx

add bx, ax

mov [si], bx

jmp end

;<Service 2 prompts the user to enter a 16-bit number, reads in 4 digits.>

Serv02: ;<Defines the beginning of the Second service routine of the ineterrupt# 65.>

mov ax, [si]

mov bx, 10h

mov cx, 0x4

flip:

div bx

push dx

loop flip

mov cx, 0x4

lea dx, THK1

mov ah, 09h

int 21h ;<Raise interrupt 21h. This call the MS-DOS service to print the "msg1" string.>

output:

pop ax

add al, 0x30

mov dl, al

mov ah, 2

int 21h

loop output

end:

iret

isr0x65 endp

end1:

hlt

;Taaha Hussain Khan

;L1F21BSCS0917

;C7