**SANA MEHMOOD**

**56331**

**CODE 1:**

org 100h

.model small

.stack 100h

.data

msg1 DB 10,13, "Enter value of A:$"

msg2 DB 10,13, "Enter value of B:$"

msg3 DB 10,13, "A is greater than B:$"

msg4 DB 10,13, "A is less than B:$"

msg5 DB 10,13, "A is equal to B:$"

.code

main proc

mov ax, @data

mov ds, ax

mov dx, offset msg1

mov ah, 9

int 21h

mov ah, 1

int 21h

mov cl, al

mov dx, offset msg2

mov ah, 9

int 21h

mov ah, 1

int 21h

mov dl, al

cmp cl, dl

jg label\_greater

jl label\_less

mov dx, offset msg5

mov ah, 9

int 21h

jmp end\_program

label\_greater:

mov dx, offset msg3

mov ah, 9

int 21h

jmp end\_program

label\_less:

mov dx, offset msg4

mov ah, 9

int 21h

end\_program:

mov ah, 4Ch

int 21h

main endp

end main

**CODE 2:**

org 100h

.model small

.stack 100h

.data

msg1 DB 10,13, "Enter your lab mid marks (0-9): $"

msg2 DB 10,13, "Need hard work.$"

msg3 DB 10,13, "Satisfactory.$"

msg4 DB 10,13, "Invalid input! Please enter a digit between 0 and 9.$"

.code

main proc

mov ax, @data

mov ds, ax

mov dx, offset msg1

mov ah, 9

int 21h

mov ah, 1

int 21h

sub al, 30h

cmp al, 9

jg invalid\_input

cmp al, 0

jl invalid\_input

cmp al, 5

jl hard\_work

mov dx, offset msg3

mov ah, 9

int 21h

jmp end\_program

hard\_work:

mov dx, offset msg2

mov ah, 9

int 21h

jmp end\_program

invalid\_input:

mov dx, offset msg4

mov ah, 9

int 21h

end\_program:

mov ah, 4Ch

int 21h

main endp

end main

**CODE 3:**

org 100h

.model small

.stack 100h

.data

msg1 DB 10,13, "Enter a one-digit number (0-9): $"

msg2 DB 10,13, "The number is zero.$"

msg3 DB 10,13, "The number is positive.$"

msg4 DB 10,13, "Invalid input! Please enter a digit between 0 and 9.$"

.code

main proc

mov ax, @data

mov ds, ax

mov dx, offset msg1

mov ah, 9

int 21h

mov ah, 1

int 21h

sub al, 30h

cmp al, 9

jg invalid\_input

cmp al, 0

jl invalid\_input

cmp al, 0

je is\_zero

mov dx, offset msg3

mov ah, 9

int 21h

jmp end\_program

is\_zero:

mov dx, offset msg2

mov ah, 9

int 21h

jmp end\_program

invalid\_input:

mov dx, offset msg4

mov ah, 9

int 21h

end\_program:

mov ah, 4Ch

int 21h

main endp

end main