**COAL LAB 13**

**Task 2:**

.model small

.stack 100h

.data

num1 db 0

num2 db 0

result db 0

msg1 db 'Enter first number (0-9): $'

msg2 db 'Enter second number (0-9): $'

msgAdd db 'The sum is: $'

msgSub db 'The difference is: $'

msgMul db 'The product is: $'

msgDiv db 'The quotient is: $'

newline db 0Dh, 0Ah, '$'

.code

main proc

mov ax, @data

mov ds, ax

mov dx, offset msg1

call display\_message

call read\_number

mov num1, al

mov dx, offset msg2

call display\_message

call read\_number

mov num2, al

call add\_numbers

mov dx, offset msgAdd

call display\_message

call display\_result

mov dx, offset newline

call display\_message

call subtract\_numbers

mov dx, offset msgSub

call display\_message

call display\_result

mov dx, offset newline

call display\_message

call multiply\_numbers

mov dx, offset msgMul

call display\_message

call display\_result

mov dx, offset newline

call display\_message

call divide\_numbers

mov dx, offset msgDiv

call display\_message

call display\_result

mov dx, offset newline

call display\_message

mov ax, 4C00h

int 21h

main endp

display\_message proc

mov ah, 09h

int 21h

ret

display\_message endp

read\_number proc

mov ah, 01h

int 21h

sub al, '0'

ret

read\_number endp

display\_result proc

mov al, result

add al, '0'

mov dl, al

mov ah, 02h

int 21h

ret

display\_result endp

add\_numbers proc

mov al, num1

add al, num2

mov result, al

ret

add\_numbers endp

subtract\_numbers proc

mov al, num1

sub al, num2

mov result, al

ret

subtract\_numbers endp

multiply\_numbers proc

mov al, num1

mov bl, num2

mul bl

mov result, al

ret

multiply\_numbers endp

divide\_numbers proc

mov al, num1

mov bl, num2

xor ah, ah

div bl

mov result, al

ret

divide\_numbers endp

end main

**Task 1:**

.model small

.stack 100h

.data

msg1 db 'Hello! My name is Malaika.$'

msg2 db 'I am a student of BS Computer Science.$'

.code

main proc

mov ax, @data

mov ds, ax

call introduce

mov ax, 4C00h

int 21h

main endp

introduce proc

mov dx, offset msg1

mov ah, 09h

int 21h

mov dx, offset msg2

mov ah, 09h

int 21h

ret

introduce endp

end main

**Task 3:**

.model small

.stack 100h

.data

m1 db 'Enter value for X: $'

m2 db 'Enter value for Y: $'

m3 db 'Enter value for Z: $'

m4 db 'Result: $'

.code

main proc

mov ax, @data

mov ds, ax

mov dx, offset m1

mov ah, 9

int 21h

call input

mov bx, ax

call space

mov dx, offset m2

mov ah, 9

int 21h

call input

add bx, ax

call space

mov dx, offset m3

mov ah, 9

int 21h

call input

sub bx, ax

call space

dec bx

mov ax, bx

call display

mov ah, 4ch

int 21h

main endp

input proc

xor ax, ax

mov ah, 1

int 21h

sub al, '0'

mov ah, 0

ret

input endp

display proc

mov dx, offset m4

mov ah, 9

int 21h

add ax, '0'

mov dl, al

mov ah, 2

int 21h

mov dl, 10

mov ah, 2

int 21h

mov dl, 13

mov ah, 2

int 21h

ret

display endp

space proc

mov dl, 10

mov ah, 2

int 21h

mov dl, 13

mov ah, 2

int 21h

ret

space endp

end main

**Task 4:**

.model small

.stack 100h

.data

m1 db 'Enter first value: $'

m2 db 'Enter second value: $'

m3 db 'Enter third value: $'

m4 db 'Result: $'

.code

main proc

mov ax, @data

mov ds, ax

mov dx, offset m1

mov ah, 9

int 21h

call input

mov bx, ax

call space

mov dx, offset m2

mov ah, 9

int 21h

call input

mul bx

call space

mov dx, offset m3

mov ah, 9

int 21h

call input

mul ax

call space

call display

mov ah, 4ch

int 21h

main endp

input proc

xor ax, ax

mov ah, 1

int 21h

sub al, '0'

mov ah, 0

ret

input endp

space proc

mov dl, 10

mov ah, 2

int 21h

mov dl, 13

mov ah, 2

int 21h

ret

space endp

display proc

mov dx, offset m4

mov ah, 9

int 21h

mov dl, 10

mov ah, 2

int 21h

mov dl, 13

mov ah, 2

int 21h

ret

display endp

end main