 Description: https://lh4.googleusercontent.com/bpfGlrXcwL2InVVTEmmoD_RjmplIRjQ_qTei0xro8dNAn1_LWfEzxzwbA6Ph-12qiEAOhlUhNU8lPPq-x6jsbRG0r59XzrcxgbNNAZbAQvOMofzDXY8UgF2dYyKOfh8XmWHIBgv7CYrN-Z6srCJdzCEa-Cl_sPh8A67eEpwJKJBpnFroi2ANV9WYy4jU1nQkymy8v_s

**“Open Ended Lab”**

**COURSE :**

**Computer Architecture And Logic Design**

**SUBMITTED TO :**

**SIR Shoaib**

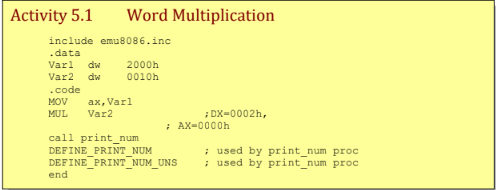
**SUBMITTED BY :**

**Rabia Batool (2021-BSE-064)**

**SECTION :**

**B**

**Activity :5.1**



include emu8086.inc

.data

Var1 dw 2000h

Var2 dw 0010h

.code

MOV ax,Var1

MUL Var2 ;DX=0002h,

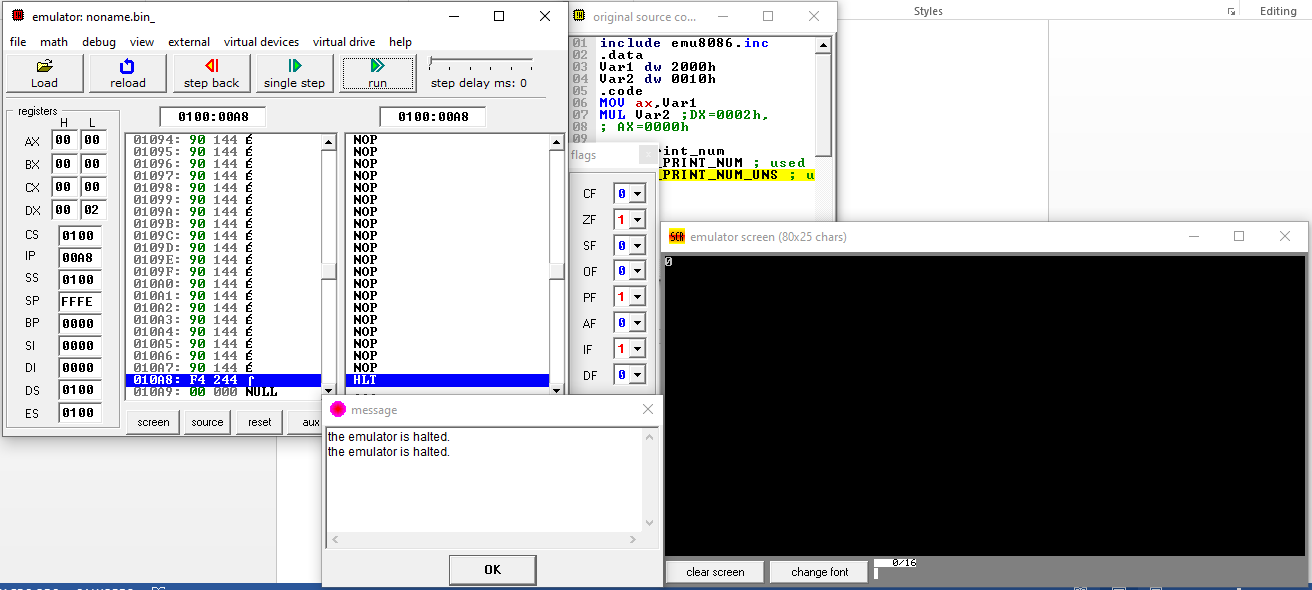
; AX=0000h

call print\_num

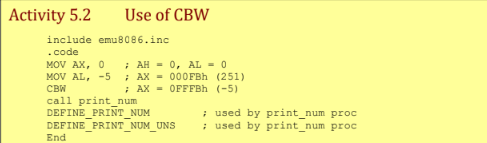
DEFINE\_PRINT\_NUM ; used by print\_num proc

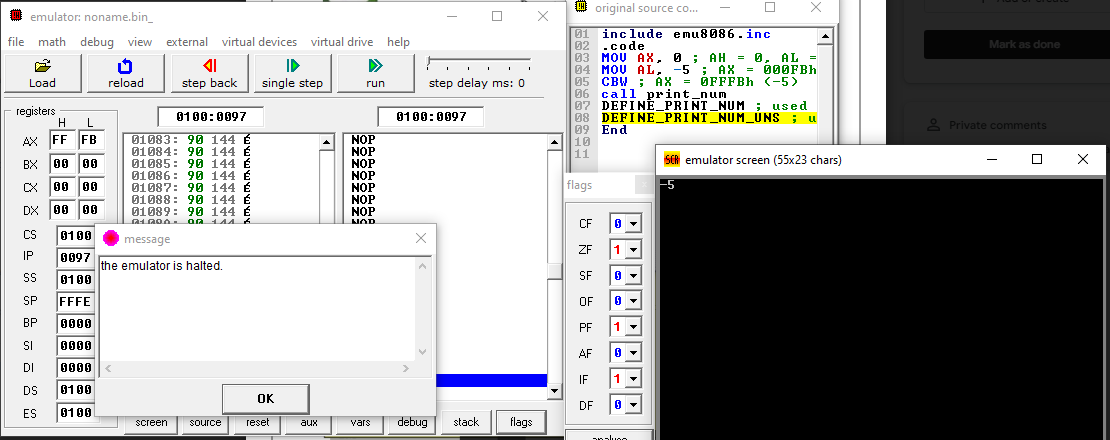
DEFINE\_PRINT\_NUM\_UNS ; used by print\_num proc

End

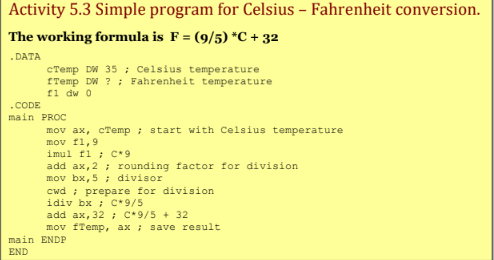


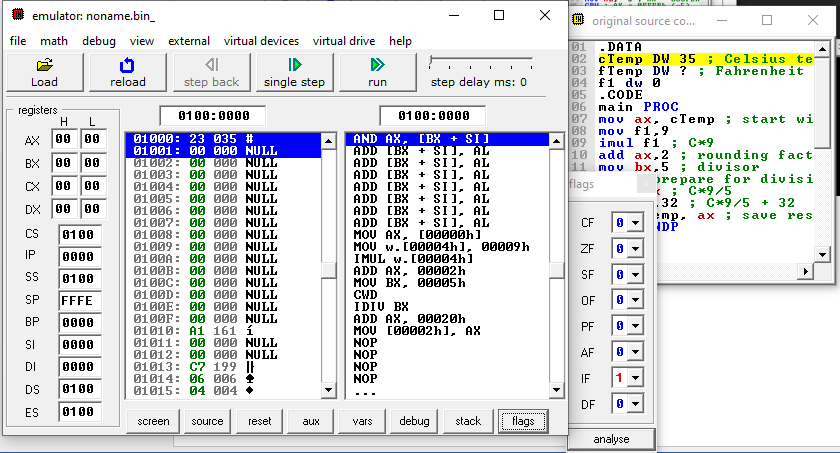
**Activity :5.2**



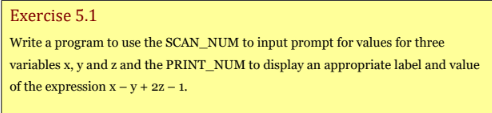


**Activity :5.3**





**Exercise :5.1**



**Code :**

.MODEL SMALL

.STACK 100h

include emu8086.inc ; Include necessary header file for SCAN\_NUM and PRINT\_NUM

.DATA

prompt1 DB "Enter value for x: $"

prompt2 DB "Enter value for y: $"

prompt3 DB "Enter value for z: $"

resultLabel DB "The value of the expression x - y + 2z - 1 is: $"

x DW ?

y DW ?

z DW ?

result DW ?

.CODE

MAIN PROC

MOV AX, @DATA

MOV DS, AX

; Input values for x, y, and z

LEA DX, prompt1

MOV AH, 9

INT 21h

CALL SCAN\_NUM ; Input x into CX (assuming SCAN\_NUM uses CX for output)

MOV x, CX

LEA DX, prompt2

MOV AH, 9

INT 21h

CALL SCAN\_NUM ; Input y into CX

MOV y, CX

LEA DX, prompt3

MOV AH, 9

INT 21h

CALL SCAN\_NUM ; Input z into CX

MOV z, CX

; Calculate the expression

MOV AX, x

SUB AX, y

MOV BX, z

SHL BX, 1 ; Multiply z by 2

ADD AX, BX

SUB AX, 1

MOV result, AX

; Display the result

LEA DX, resultLabel

MOV AH, 9

INT 21h

MOV AX, result ; Load the result into AX for PRINT\_NUM

CALL PRINT\_NUM ; Assuming PRINT\_NUM expects the value in AX

MOV AH, 4Ch

INT 21h

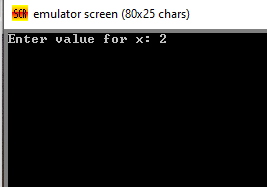
DEFINE\_SCAN\_NUM ; used by scan\_num proc

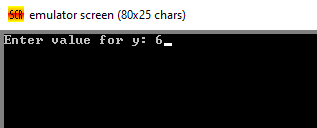
DEFINE\_PRINT\_NUM ; used by print\_num proc

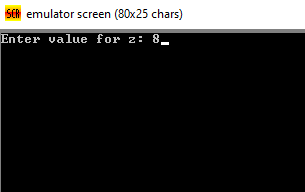
DEFINE\_PRINT\_NUM\_UNS ; used by print\_num proc

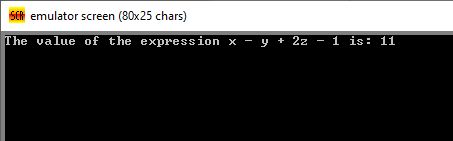
END MAIN

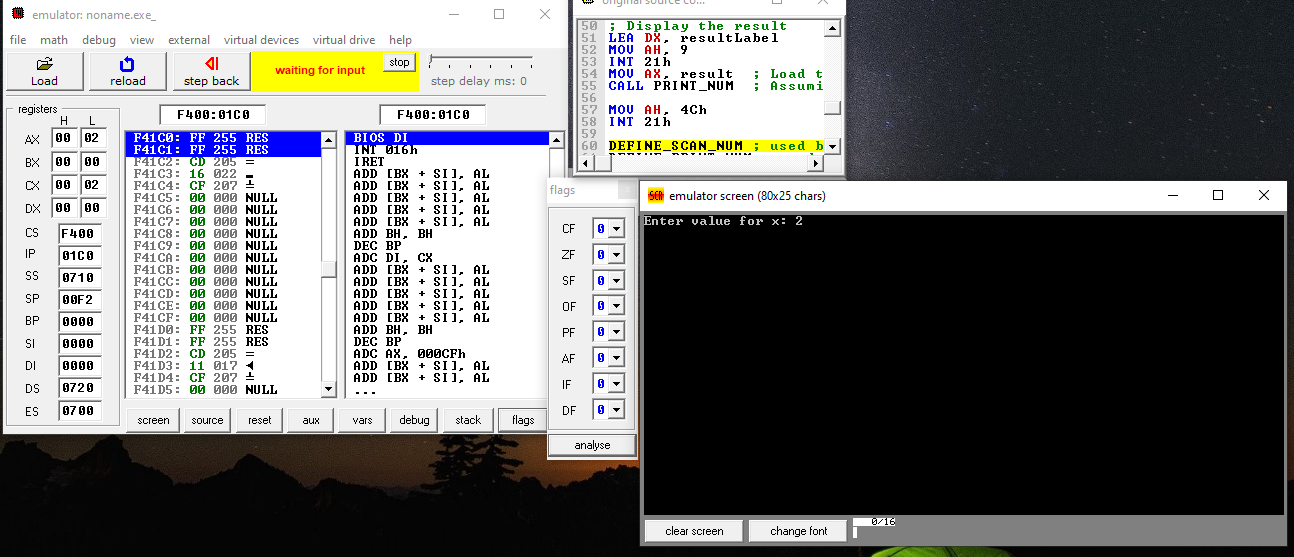
**Output :**

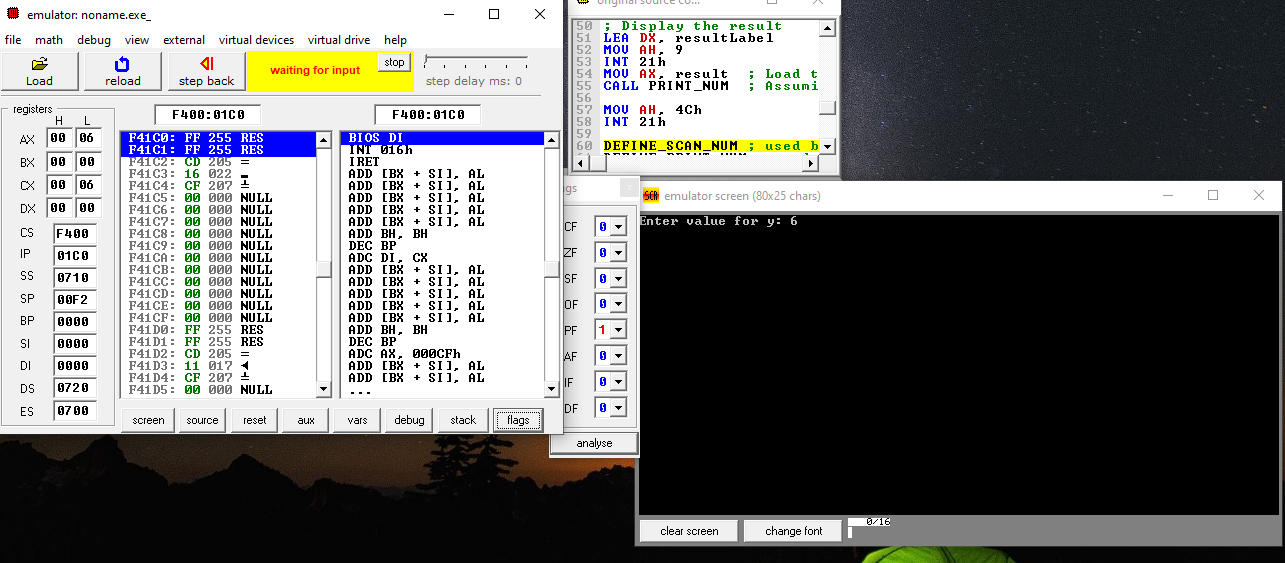


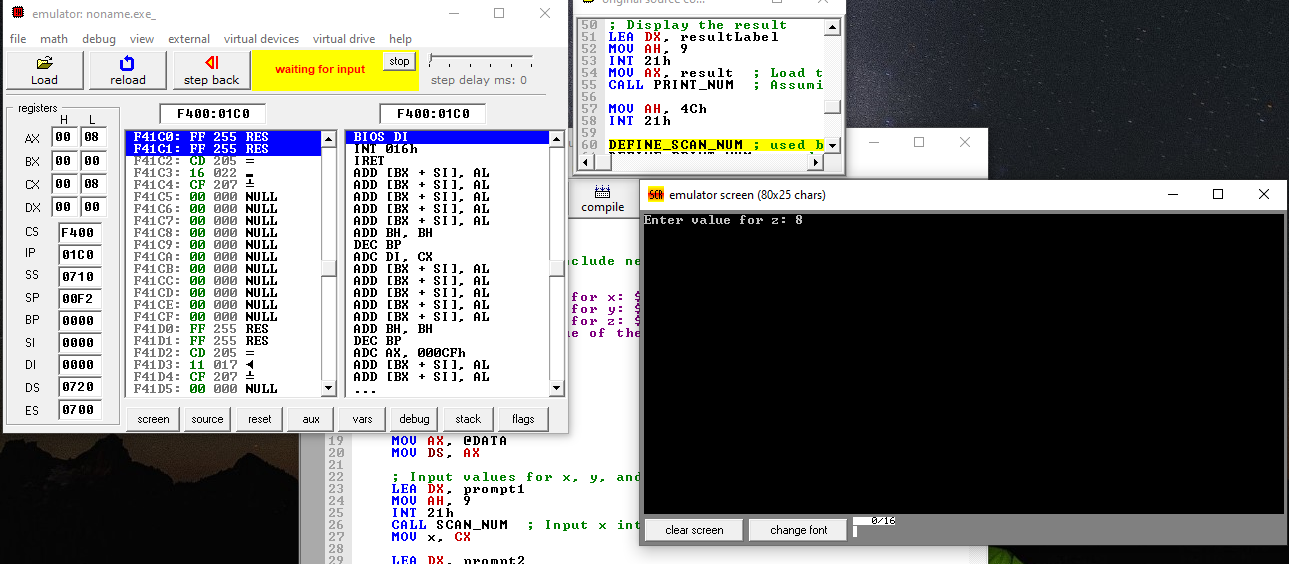


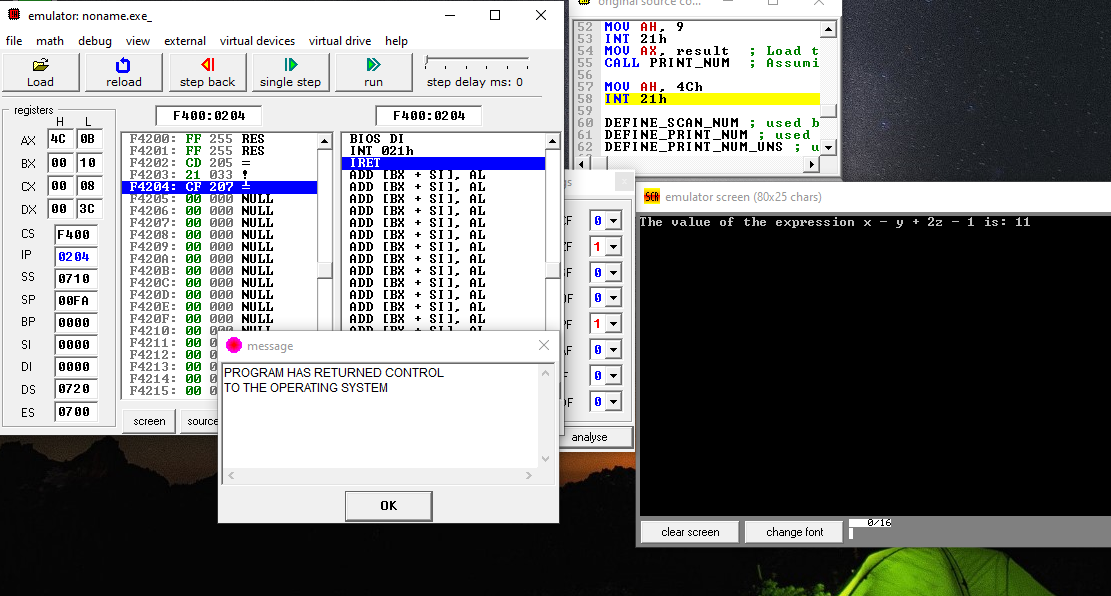




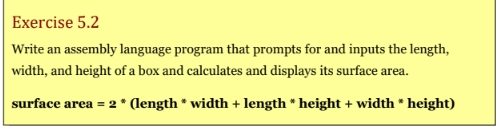








**Exercise :5.2**



**Code :**

.MODEL SMALL

.STACK 100h

INCLUDE emu8086.inc

.DATA

promptLength DB "Enter length of the box: $"

promptWidth DB "Enter width of the box: $"

promptHeight DB "Enter height of the box: $"

resultLabel DB "The surface area of the box is: $"

length DW ?

width DW ?

height DW ?

surfaceArea DW ?

.CODE

MAIN PROC

MOV AX, @DATA

MOV DS, AX

LEA DX, promptLength

MOV AH, 9

INT 21h

CALL SCAN\_NUM

MOV length, CX

LEA DX, promptWidth

MOV AH, 9

INT 21h

CALL SCAN\_NUM

MOV width, CX

LEA DX, promptHeight

MOV AH, 9

INT 21h

CALL SCAN\_NUM

MOV height, CX

; Calculate the surface area

MOV AX, length

MOV BX, width

MUL BX ; AX = length \* width

MOV CX, length

MOV DX, height

MUL DX ; DX:AX = length \* height

ADD AX, DX ; AX = length \* width + length \* height

MOV BX, width

MOV DX, height

MUL DX ; DX:AX = width \* height

ADD AX, DX ; AX = length \* width + length \* height + width \* height

SHL AX, 1 ; AX = 2 \* (length \* width + length \* height + width \* height)

MOV surfaceArea, AX

; Display the result

LEA DX, resultLabel

MOV AH, 9

INT 21h

MOV AX, surfaceArea

CALL PRINT\_NUM

MOV AH, 4Ch

INT 21h

DEFINE\_SCAN\_NUM

DEFINE\_PRINT\_NUM

DEFINE\_PRINT\_NUM\_UNS

END MAIN

MAIN

**Output :**

