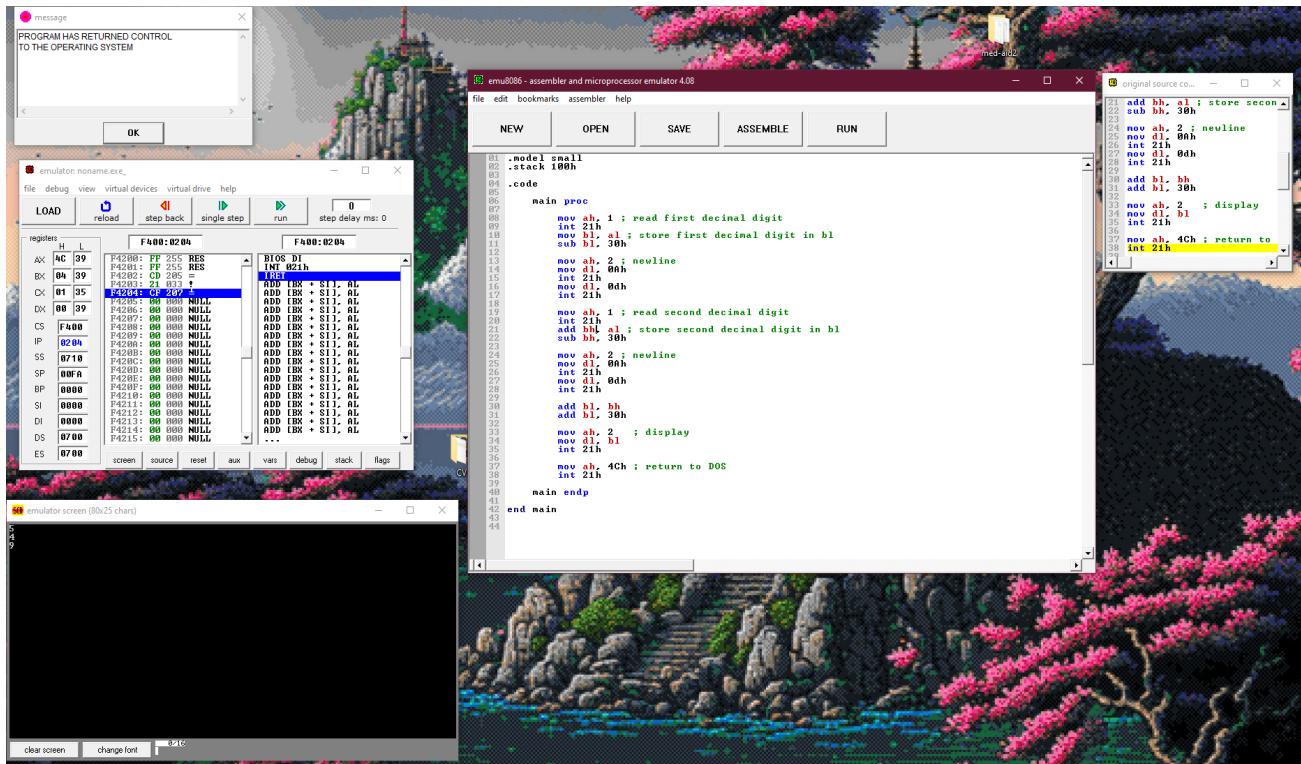


TASK-1

First we had to take input of the first number. We took the input from the console and stored it in "bl". Same way we took the second decimal input and before and also after that we had to print a newline.



Then we had to add them. So using the add command we add bl with bh and stored it in bl. After the summation we converted it to decimal and displayed it. Finally ended the program by returning.

Input 1: 5

Input 2: 4

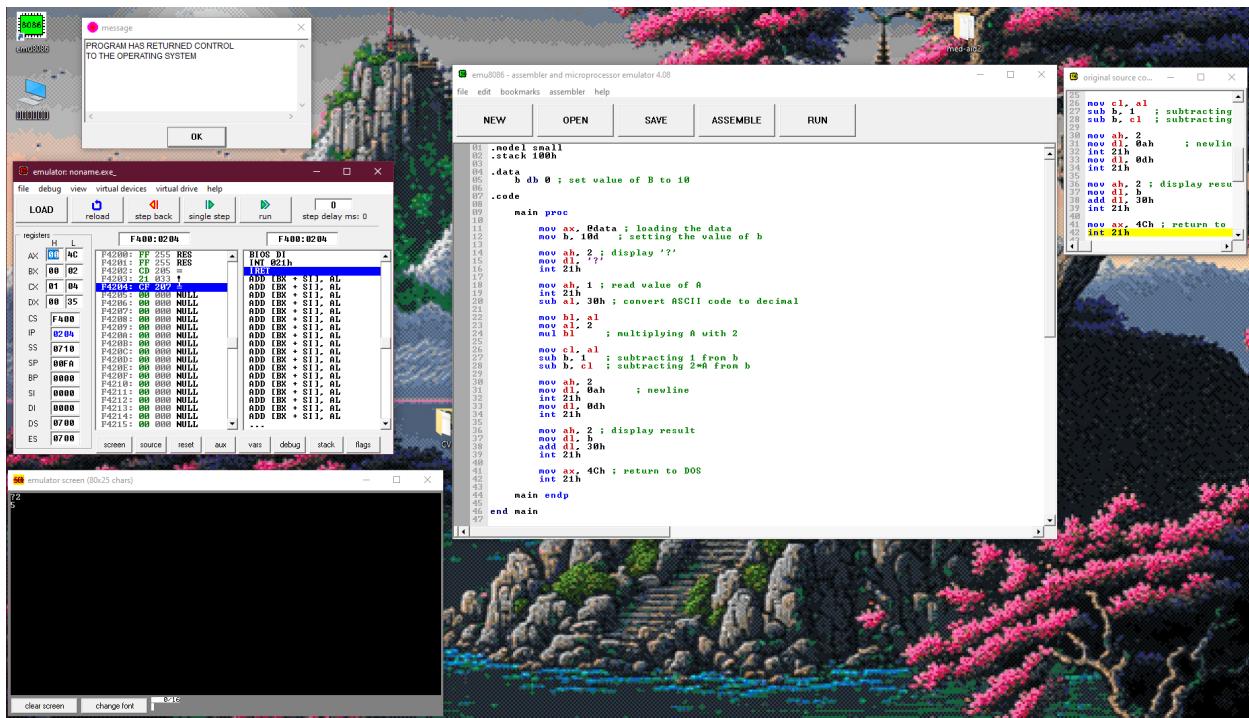
Sum: 9

TASK-2

In this task, we had to calculate an expression. The value of B is fixed. So we assigned a databyte named "b" in the data segment.

Then load that data segment. Displaying the "?" mark to the console and then took the input of another variable "a". Then did the multiplication, a with 2.

Then we subtracted the value from b and also subtracted 1 from the final value.



Then printed a new line and displayed the value. Terminated the program by exiting.

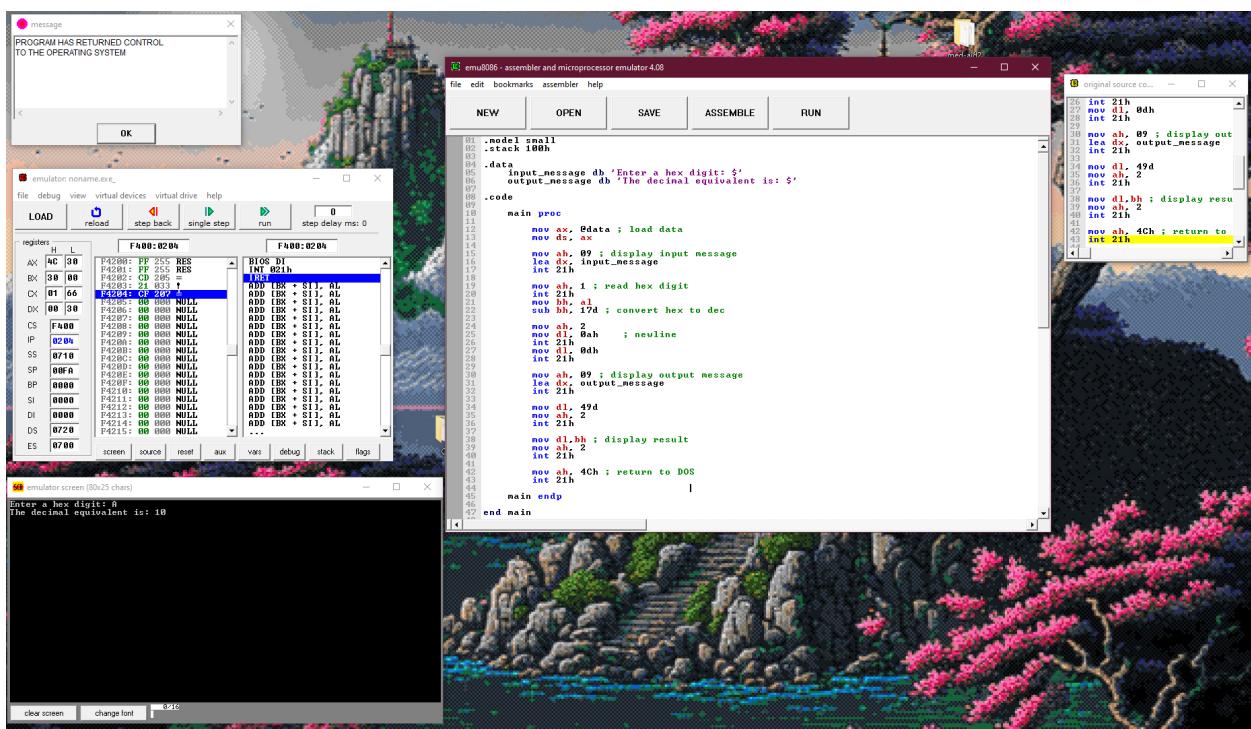
TASK-3

In task 3 it's easy. We had to take a Hexadecimal value as input and convert it to a decimal value.

So to show the input and output message to the console we loaded the message to db.

After that we took the hexa value as input. Stored it in bh and converted to decimal by subtracting 17d from it.

Then printed a newline. Displayed the output message. And displayed the converted decimal value of the input.

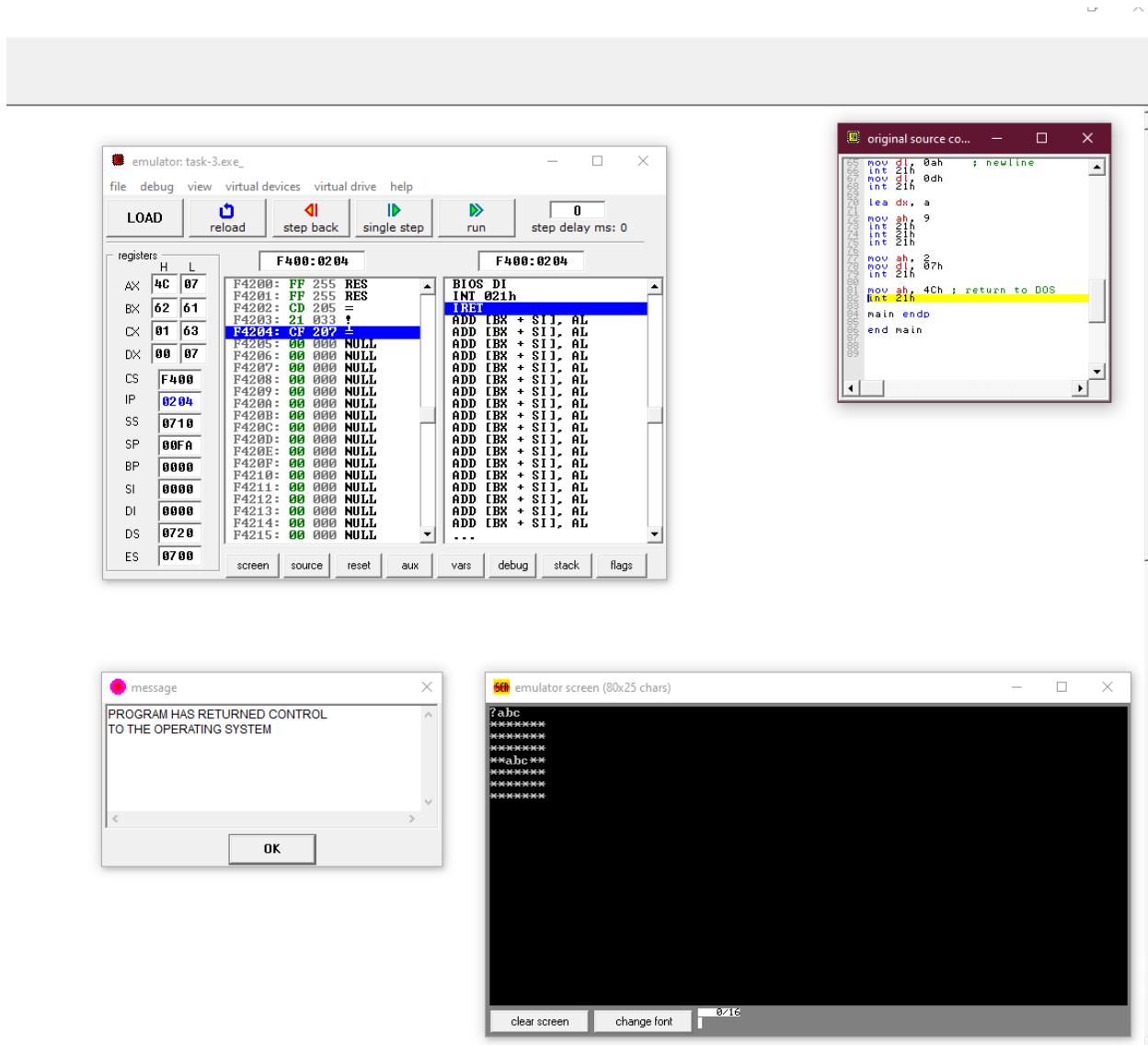


TASK-4

As we had to avoid looping. So what we did here was, we stored some repetitive pattern in data segment like “*****” and “**”.

Then we had to take input of a,b and c. So I wrote the code for that. Then printed the first pattern 3 times. Then printed the second pattern with a,b,c two times. Then again printed the first pattern 3 more times.

Finally we had to make a beep sound and terminate the program.



```
model small
.stack 100h
.data
    a db '*****', 10, 13, '$'
    mid db '**$'
.code
main proc
    mov ax, @data
    mov ds, ax
    mov ah, 2
    mov dl, '?'
    int 21h
    mov ah, 1
    int 21h
    mov bl, al
    mov ah, 1
    int 21h
    mov bh, al
    mov ah, 1
    int 21h
    mov cl, al
    mov ah, 2
    mov dl, 0ah      ; newline
    int 21h
    mov dl, 0dh
    int 21h
    lea dx, a
    mov ah, 9
    int 21h
    int 21h
    int 21h
    mov ah, 9
    lea dx, mid
    int 21h
    mov ah, 2
    mov dl, bl
    int 21h
    mov ah, 2
    mov dl, bh
    int 21h
    mov ah, 2
    mov dl, cl
    int 21h
    mov ah, 9
    lea dx, mid
    int 21h
    mov ah, 2
    mov dl, 0ah      ; newline
    int 21h
    mov dl, 0dh
    int 21h
    lea dx, a
    mov ah, 9
    int 21h
    int 21h
    int 21h
    mov ah, 2
    mov dl, 07h
    int 21h
    mov ah, 4Ch ; return to DOS
    int 21h
main endp
end main
```