



Lab 2_1

Bonilla Caiza David Alejandro

Universidad de las Fuerzas Armadas ESPE

31073: Computación Digital

Ing. Urbina Gamboa Wilmer Marcelo

08 de diciembre del 2025

Laboratorio

Escriba un programa en ensamblador x86 que realice las siguientes tareas:

- Presente en pantalla el decremento de 99 a 00.

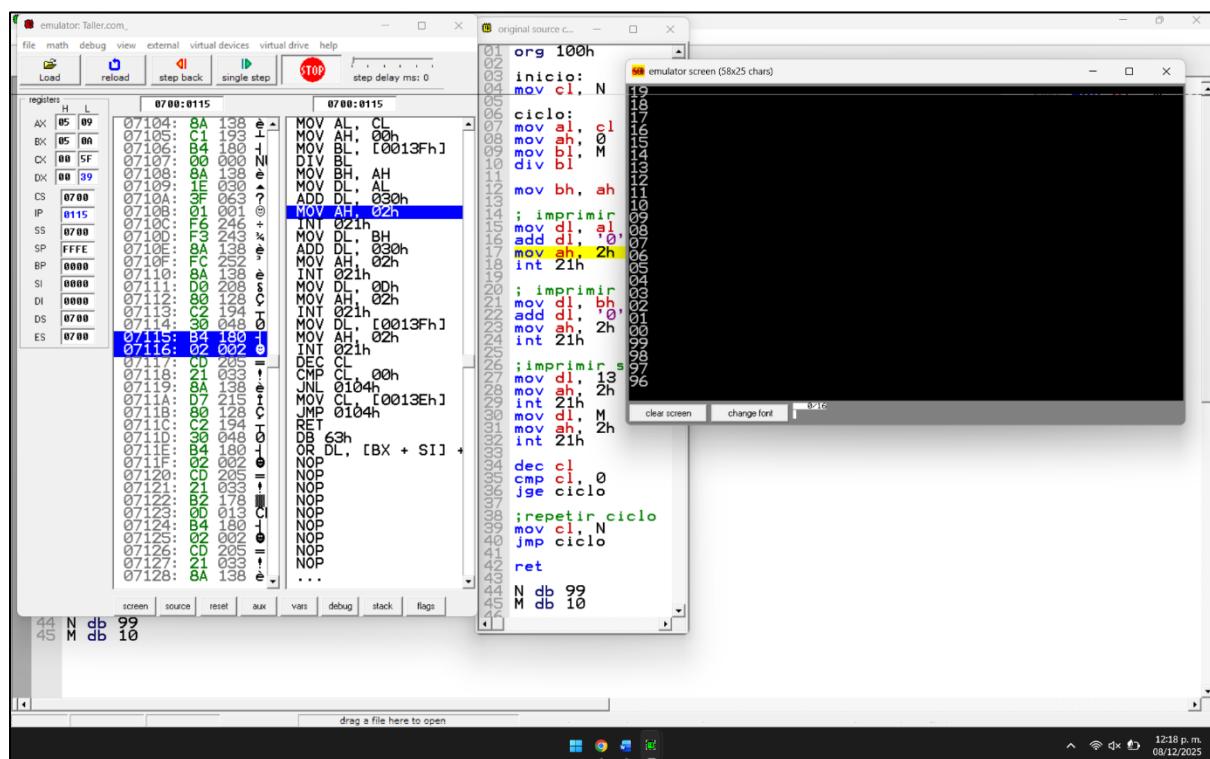
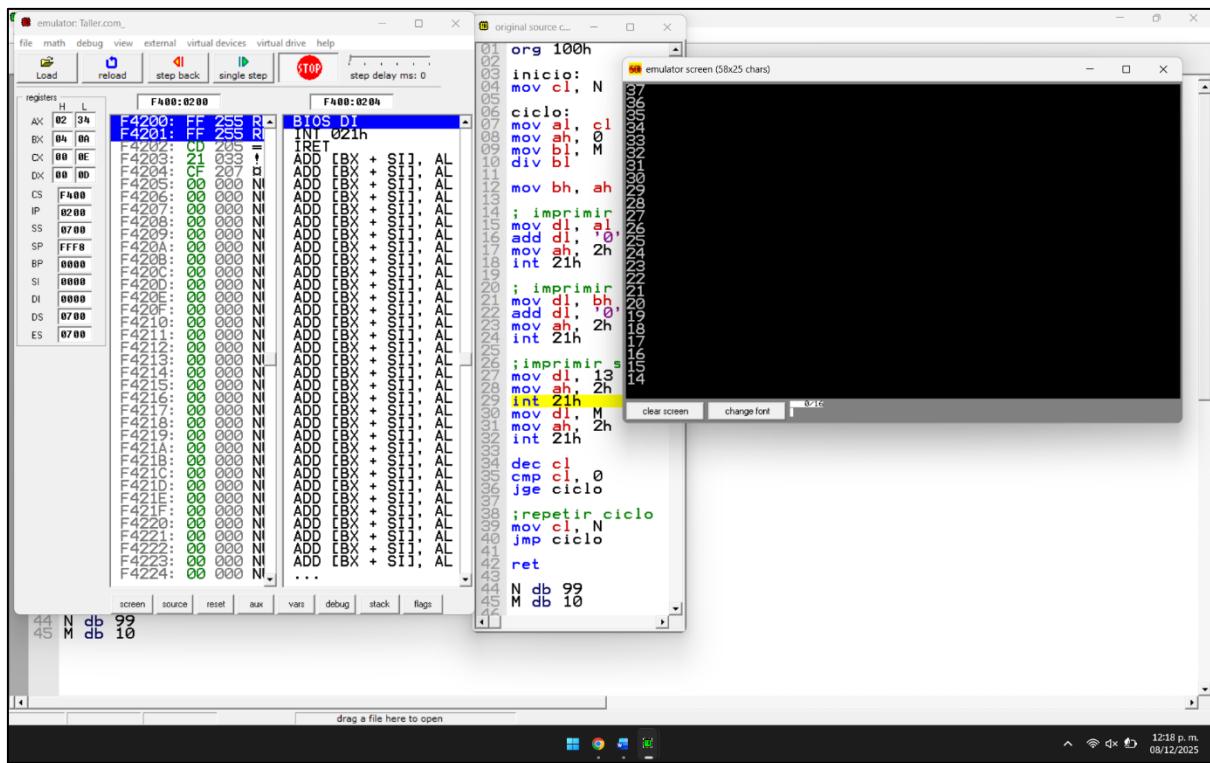
Compilación

The screenshot shows a debugger interface with two windows. The left window displays assembly code for a decrement loop. The right window shows the emulator screen output. The assembly code includes instructions for moving values between registers (AL, AH, BL, BH, CL, DL, DH, BP, SI, DI, DS, ES) and memory, performing arithmetic operations (MOV, ADD, SUB, INT 21h), and printing to the screen (INT 21h). The right window shows the output of the program, which prints the numbers 99, 98, 97, 96, 95, 94, 93, 92, 91, 90, 89, 88, 87, 86, 85, 84, 83, 82, 81, 80, 79, 78, 77, 76, 75, 74, 73, 72, 71, 70, 69, 68, 67, 66, 65, 64, 63, 62, 61, 60, 59, 58, 57, 56, 55, 54, 53, 52, 51, 50, 49, 48, 47, 46, 45, 44, 43, 42, 41, 40, 39, 38, 37, 36, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0. The registers window shows initial values for AX, BX, CX, DX, CS, IP, SS, SP, BP, SI, DI, DS, and ES. The right window also has buttons for 'clear screen' and 'change font'.

```
01 org 100h
02 inicio:
03     mov cl, N
04     ciclo:
05         mov al, cl
06         mov ah, 00h
07         mov bl, M
08         div bl
09         mov bh, ah
10         ; imprimir
11         mov dl, al
12         add dl, 00h
13         mov ah, 2h
14         Int 21h
15         ; imprimir
16         mov dl, bh
17         add dl, 00h
18         mov ah, 2h
19         Int 21h
20         ; imprimir
21         mov dl, 13h
22         int 21h
23         mov dh, M
24         mov ah, 2h
25         int 21h
26         ; imprimir
27         mov dl, 13h
28         int 21h
29         mov ah, 2h
30         int 21h
31         ; imprimir
32         mov dl, bh
33         add dl, 00h
34         mov ah, 2h
35         int 21h
36         ; imprimir
37         mov dl, 13h
38         int 21h
39         mov ah, 2h
40         int 21h
41         ; repetir ciclo
42         mov cl, N
43         jmp ciclo
44         ret
N db 99
M db 10
```

This screenshot is identical to the one above, showing the same assembly code for the decrement loop and the same output on the emulator screen. The assembly code and register values are identical to the first screenshot.

```
01 org 100h
02 inicio:
03     mov cl, N
04     ciclo:
05         mov al, cl
06         mov ah, 00h
07         mov bl, M
08         div bl
09         mov bh, ah
10         ; imprimir
11         mov dl, al
12         add dl, 00h
13         mov ah, 2h
14         Int 21h
15         ; imprimir
16         mov dl, bh
17         add dl, 00h
18         mov ah, 2h
19         Int 21h
20         ; imprimir
21         mov dl, 13h
22         int 21h
23         mov dh, M
24         mov ah, 2h
25         int 21h
26         ; imprimir
27         mov dl, 13h
28         int 21h
29         mov ah, 2h
30         int 21h
31         ; imprimir
32         mov dl, bh
33         add dl, 00h
34         mov ah, 2h
35         int 21h
36         ; imprimir
37         mov dl, 13h
38         int 21h
39         mov ah, 2h
40         int 21h
41         ; repetir ciclo
42         mov cl, N
43         jmp ciclo
44         ret
N db 99
M db 10
```



Código

```
org 100h

inicio:
    mov cl, N

ciclo:
    mov al, cl
    mov ah, 0
    mov bl, M
    div bl

    mov bh, ah

    ; imprimir decena
    mov dl, al
    add dl, '0'
    mov ah, 2h
    int 21h

    ; imprimir unidad
    mov dl, bh
    add dl, '0'
    mov ah, 2h
    int 21h

    ;imprimir salto
    mov dl, 13
    mov ah, 2h
    int 21h
    mov dl, M
    mov ah, 2h
    int 21h

    dec cl
    cmp cl, 0
    jge ciclo

    ;repetir ciclo
    mov cl, N
    jmp ciclo

ret

N db 99
M db 10
```