

TALLER #12

```
%macro PRINT_NUM 1
; Simula la llamada a la función que convierte y escribe el número.
mov eax, %1      ; Mueve el resultado a EAX (o al registro de retorno esperado)
; Llamadas reales para impresión irían aquí (sys_write, conversiones, etc.)
%endmacro

; SECCIÓN DE DATOS INICIALIZADOS (.data)**
section .data
; Definición de la ESTRUCTURA DE DATOS (Mi_Estructura)
; Contiene los 3 números a sumar
Mi_Estructura:
    num1 dd 5      ; Primer número (Double Word = 4 bytes)
    num2 dd 10     ; Segundo número
    num3 dd 15     ; Tercer número
msg db "La suma de los elementos es: ", 0

section .text
global _start

_start:
; 1. Inicializar la suma
mov ebx, 0          ; EBX será nuestro acumulador de suma.

; 2. Cargar y sumar el primer número (num1)
mov eax, [Mi_Estructura + 0] ; Cargar num1 (offset 0)
add ebx, eax         ; EBX = 0 + 5 = 5

; 3. Cargar y sumar el segundo número (num2)
; num2 está en el offset 4 (Mi_Estructura + 4), ya que num1 ocupa 4 bytes
mov eax, [Mi_Estructura + 4] ; Cargar num2 (offset 4)
add ebx, eax         ; EBX = 5 + 10 = 15

; 4. Cargar y sumar el tercer número (num3)
mov eax, [Mi_Estructura + 8] ; Cargar num3 (offset 8)
add ebx, eax         ; EBX = 15 + 15 = 30

PRINT_NUM ebx
```