

## TALLER #6

```
section .data
    num1      dd 10
    num2      dd 20

    msgMayor  db "num1 es mayor que num2", 10, 0
    msgMenor  db "num1 es menor que num2", 10, 0
    msgIgual  db "num1 es igual a num2", 10, 0

    lenMayor  equ $ - msgMayor
    lenMenor  equ $ - msgMenor
    lenIgual  equ $ - msgIgual

section .text
    global _start

_start:
    ; cargar variables en registros
    mov eax, [num1]
    mov ebx, [num2]

    ; compararlas
    cmp eax, ebx

    ; dependiendo del resultado, saltar
    je esIgual      ; saltar si eax == ebx
    jg esMayor      ; saltar si eax > ebx
    jl esMenor      ; saltar si eax < ebx

; Bloque si num1 > num2
esMayor:
    mov ecx, msgMayor
    mov edx, lenMayor
    jmp imprimir

; Bloque si num1 < num2
```

```

esMenor:
    mov ecx, msgMenor
    mov edx, lenMenor
    jmp imprimir

; -----
; Bloque si num1 == num2
; -----
esIgual:
    mov ecx, msgIgual
    mov edx, lenIgual

; -----
; Rutina para imprimir
; -----
imprimir:
    mov eax, 4          ; syscall write
    mov ebx, 1          ; file descriptor (stdout)
    int 0x80

; terminar programa
salir:
    mov eax, 1          ; syscall exit
    xor ebx, ebx
    int 0x80

```