

## PRÁCTICA 5

Ing. y Esp. Rodolfo Guadalupe Alcántara Rosales



NOMBRE DE LA PRÁCTICA	Instrucción de comparación CMP			No.	UNIDAD 2
ASIGNATUR A:	LENGUAJE INTERFAZ	CARR ERA:	ISIC	PLAN:	ISIC-2010- 204

Nombre: Vanesa Hernández Martínez

Grupo: 3501

Objetivo: Realizar un programa que compara las edades de dos personas en ensamblador.

 Realiza un programa en ensamblador que compare las edades de dos personas e indique cual es la mayor. Deberá imprimirse los nombres de las persona e indicar la mayor.

Anexa la captura de pantalla del código y la corrida del programa:

```
mensajeNombrel db "Ingre
mensajeEdad1 db 10, 13,
mensajeEdad2 db 10, 13,
mensajeEdad2 db 10, 13,
mensajeMayor db 10, 13,
mensajeMayor db 10, 13,
        ax, @data
ds, ax
         dx, mensajeNombrel
                  nombre1
                 mensajeEdad1
           dx, mensajeNombre2
         ah, 0Ah
dx, nombre2
```



## PRÁCTICA 5

Ing. y Esp. Rodolfo Guadalupe Alcántara Rosales



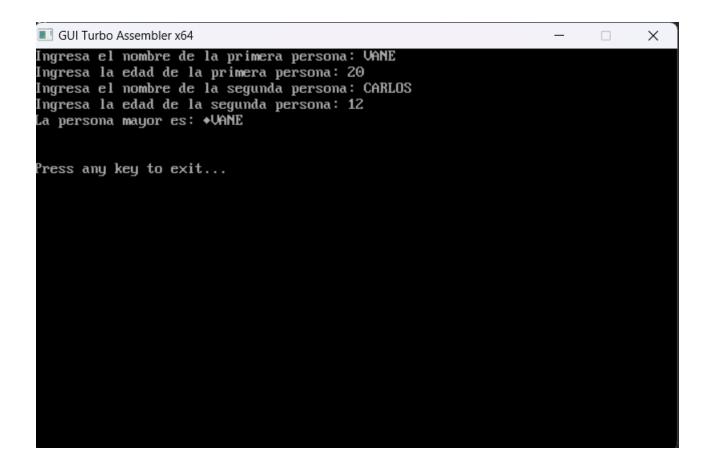
```
lea dx, mensajeNombre2
int 21h
mov ah, 0Ah
lea dx, nombre2
int 21h
       mov ah, 09h
lea dx, mensajeEdad2
int 21h
       ; Leer el primer d?gito de la edad
mov ah, 01h
int 21h
sub al, 30h
mov bl, al
      ; Leer el segundo d?gito de la edad
mov ah, 01h
int 21h
sub al, 30h
mov bh, al
      ; Calcular edad2 = (primer_digito * 10) + segundo_digito
mov ah, 0
mul cl
add al, bh
mov edad2, al
       mov al, edad1
cmp al, edad2
jg primeraMayor
jl segundaMayor
primeraMayor:
       mov ah, 09h
lea dx, mensajeMayor
       int 21h
lea dx, nombre1+1 ; +1 para saltar la longitud del string
mov ah, 09h
       mov ah,
int 21h
jmp fin
segundaMayor:
       mov ah, 09h
lea dx, mensajeMayor
       int 21h
lea dx, nombre2+1 ; +1 para saltar la longitud del string
mov ah, 09h
int 21h
fin:
       ; Terminar el programa mov ax, 4C00h int 21h
main endp
```



## PRÁCTICA 5

Ing. y Esp. Rodolfo Guadalupe Alcántara Rosales





## Conclusión

Este programa demuestra la capacidad del lenguaje ensamblador para realizar operaciones lógicas y manipulaciones de datos con eficiencia a bajo nivel. Aunque el código en ensamblador requiere una mayor comprensión del hardware y la gestión manual de registros, el resultado final resalta la versatilidad de este lenguaje para resolver problemas prácticos, como la comparación de datos y la generación de salidas personalizadas como la identificación de la persona de mayor edad.