



**Universidad Autónoma de Baja California
Facultad de Ingeniería Arquitectura y Diseño
Organización de computadoras**

Docente:

Crespo Regland Jonatan

Alumno:

Garcia Ceseña Edgar Rafael

GRUPO 932

Ingeniería en Software y Tecnologías Emergentes

TALLER 12



```

1 section .data
2     mensaje1 db "Ingrese el primer numero: ", 0
3     mensaje2 db "Ingrese el segundo numero: ", 0
4     resultado_msg db "El resultado es: ", 0
5     num1 db 0
6     num2 db 0
7
8 section .bss
9     resultado resb 4 ; Espacio para almacenar el resultado
10
11 section .text
12     global _start
13
14 _start:
15     ; Pedir el primer número
16     mov eax, 4 ; syscall: escribir
17     mov ebx, 1 ; archivo: stdout
18     mov ecx, mensaje1 ; mensaje a mostrar
19     mov edx, 24 ; longitud del mensaje
20     int 0x80
21
22     ; Leer el primer número
23     mov eax, 3 ; syscall: leer
24     mov ebx, 0 ; archivo: stdin
25     mov ecx, num1 ; buffer donde se guarda el número
26     mov edx, 1 ; leer un carácter
27     int 0x80
28     sub byte [num1], '0' ; convertir de carácter ASCII a número
29
30     ; Pedir el segundo número
31     mov eax, 4
32     mov ebx, 1
33     mov ecx, mensaje2
34     mov edx, 24
35     int 0x80
36
37     ; Leer el segundo número
38     mov eax, 3
39     mov ebx, 0
40     mov ecx, num2
41     mov edx, 1
42     int 0x80
43     sub byte [num2], '0'
44
45     ; Sumar los números
46     mov al, [num1]
47     add al, [num2]
48     add al, '0' ; convertir de número a carácter ASCII

```

STDIN

54

Output:

Ingrese el primer numeroIngrese el segundo numeroEl resultado es: 9

```

21
22     ; Leer el primer número
23     mov eax, 3 ; syscall: leer
24     mov ebx, 0 ; archivo: stdin
25     mov ecx, num1 ; buffer donde se guarda el número
26     mov edx, 1 ; leer un carácter
27     int 0x80
28     sub byte [num1], '0' ; convertir de carácter ASCII a número
29
30     ; Pedir el segundo número
31     mov eax, 4
32     mov ebx, 1
33     mov ecx, mensaje2
34     mov edx, 24
35     int 0x80
36
37     ; Leer el segundo número
38     mov eax, 3
39     mov ebx, 0
40     mov ecx, num2
41     mov edx, 1
42     int 0x80
43     sub byte [num2], '0'
44
45     ; Sumar los números
46     mov al, [num1]
47     add al, [num2]
48     add al, '0' ; convertir de número a carácter ASCII
49     mov [resultado], al ; guardar resultado en el buffer
50
51     ; Mostrar el resultado
52     mov eax, 4
53     mov ebx, 1
54     mov ecx, resultado_msg
55     mov edx, 18
56     int 0x80
57
58     ; Imprimir el resultado
59     mov eax, 4
60     mov ebx, 1
61     mov ecx, resultado
62     mov edx, 1
63     int 0x80
64
65     ; Salir del programa
66     mov eax, 1
67     xor ebx, ebx
68     int 0x80

```

STDIN

54

Output:

Ingrese el primer numeroIngrese el segundo numeroEl resultado es: 9