Universidad Autónoma de Baja California
Facultad de Ingeniería Arquitectura y Diseño

Ingeniería en Software y Tecnologías Emergentes

Materia: Organización de Computadoras

Alumno: Jesus Eduardo Rodríguez Ramírez

Profesor: Jonatan Crespo Ragland

Grupo 932

Trabajo: CÓDIGOS DE ENSAMBLADOR Taller 5

Ensenada, B.C; a 27 de septiembre del 2024

1. **De acuerdo con el siguiente programa en ensamblador, identifica, desarrolla y describe su funcionamiento. (Qué secciones conforman al programa, qué tipo de registros se utilizan y por qué.**

```
section .data msg db 'Imprimir input del teclado: ',0 ;
Mensaje que se mostrará antes de la entrada, //se
cambió por el ‘imput: ‘

section .bss input resb 1 ; Espacio para almacenar el
carácter ingresado
 sum resb 1 ; Espacio para almacenar la suma

section .text
global _start

_start:
; Mostrar mensaje en consola
mov eax, 4
mov ebx, 1
mov ecx, msg ; dirección del mensaje
mov edx, 30 ; longitud del mensaje //se extendió la
longitud del mensaje para que mostrar todo el mensaje
completo que se pide
int 0x80

; Leer un carácter desde el teclado

mov eax, 3
mov ebx, 0
mov ecx, input ; dirección para almacenar la entrada
mov edx, 80 ; leer 1 byte (1 carácter) //se extiende la
longitud de bytes que lee para el mensaje que se pide
en el taller
int 0x80

; Mostrar el carácter ingresado
mov eax, 4 ; syscall número 4 es write (sys_write)
mov ebx, 1 ; descriptor de archivo 1 es stdout
```

mov ecx, input ; dirección del carácter
mov edx, 20 ; longitud del carácter //igual se extendió la longitud del carácter para el mensaje que se pide, pero este es específicamente para el carácter que se ingresa
int 0x80 ; llamada al sistema

; Calcular la suma de los caracteres
mov al, [input]
add al, [input]
mov [sum], al ; almacenar la suma en la variable sum

; Mostrar la suma
mov eax, 4
mov ebx, 1
mov ecx, sum ; dirección de la suma
mov edx, 1 ; longitud de la suma
int 0x80

; Terminar el programa mov eax, 1 xor ebx, ebx ; código de salida 0 int 0x80

2. **Cambia los valores que se están asignando a los registros eax, ebx, int 0x80. Desarrolla qué pasa cuando cambias estos valores y por qué.**

```
12  ; Mostrar mensaje en consola
13  mov eax, 4
14  mov ebx, 1
15  mov ecx, msg ; dirección del mensaje
16  mov edx, 30 ; longitud del mensaje
17  int 0x80
18
19  ; Leer un carácter desde el teclado
20
21  mov eax, 3
22  mov ebx, 0
23  mov ecx, input ; dirección para almacenar la entrada
24  mov edx, 80 ; leer 1 byte (1 carácter)
25  int 0x80
26
27  ; Mostrar el carácter ingresado
28  mov eax, 4 ; syscall número 4 es write (sys_write)
29  mov ebx, 1 ; descriptor de archivo 1 es stdout
30  mov ecx, input ; dirección del carácter
31  mov edx, 20 ; longitud del carácter
32  int 0x80 ; llamada al sistema
```

3. **Cambia el mensaje de salida ‘Input: ’ ‘Imprimir input del teclado: ’.Documenta que pasa al imprimir el resultado. Modifica el programa para que imprima la nueva cadena en su totalidad y explica tus cambios.**

HelloWorld.asm 42t4huvxb

```
1  section .data
2  msg db 'Imprimir input del teclado:  ',0 ; Mensaje que se mostrará antes de la entrada
3
```

al cambiar el ‘input: ‘ por ‘Imprimir input del teclado: ‘, se muestra tal cual el mensaje en la consola seguido de la cadena de caracter dada por el usuario

4. **Modifica el programa para que imprima la siguiente cadena de texto y documenta en tu práctica de taller: Imprimir input del teclado: organización. Para esto debes ingresar un valor de input en el STDIN**