

Sesión I

Introducción al lenguaje ensamblador

El objetivo de esta sesión es recordar el lenguaje ensamblador del MIPS y el simulador que utilizaremos para la realización de las prácticas.

Carga y ejecuta el siguiente ejemplo:

```
# programa
.text
.globl main

main:
    la $t0,num
    lw $a0,0($t0)
    la $s0, datos
    addi $a1, $s0, 0

    jal sub1

    addi $a0, $v0, 0
    addi $v0, $0, 1
    syscall

    addi $v0,$0,10
    syscall

sub1:
    addi $t0, $a1, 0
    addi $t1, $0, 0
    addi $t2, $0, 0

loop:
    beq $t1, $a0, salir
    lw $t3, 0($t0)
    add $t2, $t3, $t2
    addi $t0, $t0, 4
    addi $t1, $t1, 1
    j loop

salir:
    sw $t2, 4($t0)
    addi $v0, $t2, 0
    jr $ra

.data
num: .word 3
datos: .word 1, 2, 3, 4, 5

# fin
```

1. ¿Cuántos ciclos tarda en ejecutarse?
2. ¿Qué hace el código?
3. ¿Qué queda almacenado en el registro \$t2 al final de la ejecución? ¿Qué representa?
4. ¿Cuál es la instrucción de llamada a la subrutina? ¿Qué dos acciones realiza esta instrucción?
5. ¿Qué queda almacenado en el registro \$ra después de ejecutar la instrucción jal? Escribe el valor en hexadecimal.
6. ¿Cómo se pasan los parámetros desde el programa principal a la subrutina?
7. ¿Cómo se devuelve el resultado de la subrutina al programa principal?

8. ¿Qué instrucción se utiliza para volver al programa principal cuando la subrutina ha terminado?
9. ¿A partir de qué dirección de memoria se almacena el vector `datos`? ¿Dónde está situado, en el segmento de datos o en el segmento de texto?
10. ¿En qué dirección de memoria se almacena el dato `num`?
11. En el ciclo de reloj 18 ¿qué dato se almacena en `$t3` y en qué posición de memoria está.
12. ¿En que dirección de memoria se almacena el dato que hay en `$t2`? ¿Qué instrucción se utiliza y cómo se calcula la dirección?
13. ¿Qué hace la pseudoinstrucción `la`?
14. ¿En qué dirección de memoria se almacena la instrucción `beq`? ¿Qué hace dicha instrucción?
15. ¿En qué dirección de memoria se almacena la instrucción `j`? ¿Qué hace dicha instrucción?
16. ¿Qué valor tiene el contador de programa en el ciclo de reloj 22? ¿por qué?

Escribe una subrutina que, utilizando las instrucciones de carga para bytes (un caracter se representa con un byte en ASCII), devuelva la longitud de la cadena que acaba con el primer byte a cero. El esquema del programa principal es el siguiente:

```
# programa longitud
.text
.globl main

main:
    la $a0,cadena      # la direccion de la etiqueta cadena en a0

    jal longitud

    addi $a0, $v0, 0
    addi $v0,$0,1      # la llamada para imprimir un entero
    syscall

    addi $v0,$0,10     # la llamada para salir del programa
    syscall

# segmento de datos
.data
cadena: .asciiz 'simula3MS'

# fin
```

1. ¿Cuál es la instrucción que utilizas para cargar un caracter?
2. ¿Cuál es la representación ASCII de “simula3MS”?
3. ¿Cuál es la representación ASCII de la “s”? ¿En qué posición de memoria está?
4. ¿Cuál es la representación ASCII del “3”? ¿En qué posición de memoria está?