

INTRODUCCIÓN A LOS COMPUTADORES

PRÁCTICA 3.1

EJERCICIO 1:

1. Carga en el **Simula3MS** el código *P31a.s* del CampusVirtual:

```
.data

.text
.globl main

main:

    addi $t0, $0, 5
    addi $t1, $0, -33
    add $t2, $t0, $t1

    addi $v0, $0, 10
    syscall
```

2. A continuación pulsa el botón **Ensamblar**. Si el código es sintácticamente correcto se activará el botón **Ejecutar** y se puede avanzar a la simulación del programa, si no, es necesario corregir el código teniendo en cuenta los errores.
3. Es importante examinar la pantalla de ejecución e identificar las áreas que la componen, de entre la que destacamos las siguientes:
 - El segmento de programa, donde se muestra la instrucción que se ejecuta en cada momento, incluyendo su código máquina y la dirección de memoria donde se encuentra
 - El segmento de datos (Memoria).
4. Es posible ejecutar el programa hasta el final pulsando el botón **Ejecutar**, pero es más instructivo hacerlo paso a paso utilizando el botón **Paso siguiente**. El botón **Paso anterior** permite deshacer el trabajo de una instrucción. Para volver al editor pulsar **Volver editor**.
5. Observa el *Segmento de texto*. La primera instrucción es *addi \$t0, \$0, 5*. Esa instrucción pondrá el valor $\$0 + 5$ en el registro *\$t0*, es decir, hará que $\$t0 = 5$. Busca el registro *\$t0* en el área de registros generales. Observa cuál es su valor. Pulsa *Paso siguiente*. Comprueba si ha cambiado el valor de *\$t0* en el área de registros. La primera instrucción está ahora resaltada. Es la última instrucción que hemos ejecutado.
6. La segunda instrucción es *addi \$t1, \$0, -33*, que hará que $\$t1 = -33$. Observa el valor de *\$t1*. Pulsa *Paso siguiente*. Vuelve a **comprobar** el valor de *\$t1*.
7. La tercera instrucción es *add \$t2, \$t0, \$t1*, que hará que $\$t2 = \$t0 + \$t1$. **Comprueba** que realmente se calcula esa suma.

8. Las 2 últimas instrucciones sirven para terminar el programa. Vuelve a ejecutar el programa de un tirón (botón *Ejecutar*) y/o paso a paso hasta entender completamente su funcionamiento.
9. Cambia el código para ejecutar la suma de $-512 + 1024$. **Comprueba** que realmente se calcula esa suma.
10. Indica la dirección de memoria en la que se encuentra la instrucción *addi \$t1, \$0, -33*. Indica la codificación binaria de esta instrucción. Identifica los campos dentro del formato al que pertenezca esta instrucción.
11. Indica la dirección de memoria en la que se encuentra la instrucción *add \$t2, \$t0, \$t1*. Indica la codificación binaria de esta instrucción. Identifica los campos dentro del formato al que pertenezca esta instrucción.

EJERCICIO 2:

Carga en el **Simula3MS** el código *P31b.s* del CampusVirtual:

```
.data
numeros: .word 8, -7, 10, -12
resultado: .word 0

.text
.globl main

main:

    la $a0, numeros
    lw $t1, 0($a0)
    lw $t0, 4($a0)

    add $t2, $t0, $t1

    la $a1, resultado
    sw $t2, 0($a1)

    addi $v0, $0, 10
    syscall
```

Carga el programa en el simulador y realiza las comprobaciones que se piden:

1. Comprueba en qué formato y en qué posiciones se almacenan los números originales en memoria. Anota sus direcciones y su representación en el computador. Fíjate que está en complemento a 2.
2. Con respecto a la instrucción *lw \$t1, 0(\$a0)*:
 - a) Indica la dirección en la que está
 - b) Anota la codificación binaria de la instrucción
 - c) Indica el formato y el valor de cada campo
 - d) Indica el registro fuente
 - e) Indica el registro destino
 - f) Anota la dirección que lee de memoria
3. Con respecto a la instrucción *lw \$t0, 4(\$a0)*:
 - a) Indica la dirección en la que está
 - b) Anota la codificación binaria de la instrucción
 - c) Indica el formato y el valor de cada campo
 - d) Indica el registro fuente
 - e) Indica el registro destino
 - f) Anota la dirección que lee de memoria
4. Con respecto a la instrucción *add \$t2, \$t0, \$t1* indica:
 - a) la dirección de memoria en la que se encuentra
 - b) la codificación binaria de esta instrucción

- c) los campos dentro del formato al que pertenezca esta instrucción.
- d) Indica los registros fuentes
- e) Indica el registro destino
- f) Anota la dirección que lee de memoria

Indica la dirección de memoria en la que se encuentra la instrucción *add \$t2, \$t0, \$t1*. Indica la codificación binaria de esta instrucción. Identifica los campos dentro del formato al que pertenezca esta instrucción.

5. Con respecto a la instrucción *sw \$t2, 0(\$a1)*:

- a) Indica la dirección en la que está
- b) Anota la codificación binaria de la instrucción
- c) Indica el formato y el valor de cada campo
- d) Indica los registros fuentes
- e) Anota la dirección que escribe de memoria

6. Cambia el código para sume 10 y -12 . Hay dos opciones:

- a) Cambia los campo de desplazamiento de *lw \$t1, 0(\$a0)* y *lw \$t0, 4(\$a0)*.
- b) Añade una *addi* para incrementar *\$a0*.