

**Introducción a la Arquitectura de Sistemas**  
**Guía de Trabajos Prácticos Nro 4**  
**Programación en lenguaje ensamblador MIPS**

1. Dados los valores A, B y C almacenados en las posiciones 0x2002, 0x2006 y 0x200E respectivamente, escriba una subrutina que realice la siguiente operación:  $A + B = C$

2. Implemente

```
i := 0
```

```
while i < 8 then
```

```
    begin
```

```
        c_out(i);
```

```
        i := i + 1;
```

```
    end;
```

3. Escribir una subrutina que comience en la posición 100 h de la memoria que determine si el valor almacenado en \$T5 es negativo. En caso de serlo, que lo reemplace por su valor absoluto.

4. Escribir una subrutina que calcule un promedio entero entre los valores de los registros \$T5 y \$T6 y almacene el resultado en \$T7.

5. Cómo se escriben en lenguaje máquina estas instrucciones en lenguaje ensamblador de MIPS?

```
add $t0, $s0, $s1
```

```
lw $t0, 0x20($t7)
```

```
addi $s0, $0, -10
```

Como se ven las instrucciones en lenguaje máquina en hexadecimal?

6. Escribir un programa que lea 15 valores del puerto de entrada 1Fh y que arme con ellos un arreglo que comience en la posición 1000h de la memoria. Agregar un 0 como último elemento.

7. Escribir un programa que recorra el arreglo que comienza en la posición 1020 h de la memoria y transforme todos los elementos negativos en positivos. Utilice la subrutina escrita en el ejercicio 3. Se asume que el arreglo finaliza cuando se encuentra el primer elemento con valor 0.

8. Escribir un programa que calcule el promedio elemento por elemento de los arreglos que comienzan en la posición 1000 h y 1020 h de la memoria respectivamente, y el resultado lo almacene en un nuevo arreglo que comienza en la posición 1040 h de la memoria. Todos los arreglos deben tener 0 como último valor, y el proceso debe concluir cuando se encuentre un 0 en cualquiera de los dos arreglos de entrada.

9. Se tiene en memoria un conjunto con 0Fh datos que se puede considerar como una tabla. El primer dato (inicio de la tabla) se encuentra en la posición 00C0h. Hacer un programa que vaya leyendo los datos de la tabla, cuente cuantos números positivos,

negativos y ceros hay en el conjunto. Cuando llegue al final, debe mostrar por OP1, primero cuántos números negativos había, a continuación cuantos ceros y por ultimo cuantos positivos había. Cargar el programa a partir de la posición 0010h.

**10. SOLO PARA INGENIERIA...** Escribir un programa que permita recorrer una lista vinculada, sumando todos los elementos de la misma y enviando al final el resultado por el puerto de salida 14. Cada nodo de la lista está formado por dos elementos, ocupando un total de 4 bytes de memoria. El primer elemento es un número entero de 2 bytes que corresponde al valor del nodo. El segundo elemento es un puntero de 2 bytes que indica la posición en la memoria del siguiente nodo de la lista. El último nodo tendrá un puntero nulo (el puntero indica 0). Asumir que el primer nodo de la lista se encuentra en la posición 2000 de la memoria.

**11. SOLO PARA INGENIERIA...** Describa el comportamiento del siguiente volcado de la memoria de programa:

address	instruction
00	35E0000A
04	35A00000
08	35800000
0C	018D7828
10	118F001C
14	21AD0001
18	11CF0004
1C	21CE0001
20	01AD0020
24	20000000