

Práctica 4. Operaciones lógicas

1. Escriba un programa en lenguaje ensamblador de MIPS que pida por teclado un número entero n e imprima por pantalla en decimal el número representado en binario natural por los 4 bits de menos peso de la representación en complemento a 2 de n .

Ejemplos:

- Si el número introducido fuera 133 (en binario 0...0010000101) el programa debe imprimir 5 (en binario 0101)
 - Si el número introducido fuera -1 (en binario 1...11111111) el programa debe imprimir 15 (en binario 1111)
2. Repita el ejercicio anterior, pero imprimiendo el número en hexadecimal. Para la impresión emplee solo la función del sistema que imprime caracteres. (`syscall 11`).

Ejemplos: en los casos anteriores el programa debería imprimir 5 y f, respectivamente.

3. Escriba ahora un programa en lenguaje ensamblador de MIPS que pida por teclado un número entero n e imprima por pantalla en hexadecimal el número representado en binario natural por los bits del 7 al 4 de la representación en complemento a 2 de n . Para la impresión emplee solo la función del sistema que imprime caracteres.

Ejemplos:

- Si el número introducido fuera 133 (en binario 0...0010000101) el programa debe imprimir 8 (en binario 1000)
 - Si el número introducido fuera -1 (en binario 1...11111111) el programa debe imprimir f (en binario 1111)
4. Escriba una función con los siguientes parámetros:
 - \$a0: Número a convertir.
 - \$a1: Dirección de la cadena donde quedará el número anterior en hexadecimal codificado en ASCII.

La función debe convertir el número contenido en el registro \$a0 en una cadena codificada en ASCII que exprese el número en hexadecimal. La cadena debe depositarse en la dirección contenida en el registro \$a1. No olvide incluir el terminador de cadena (0). La función no debe imprimir ningún resultado por pantalla, eso lo debe hacer el programa que la llame.

Para hacer la conversión tenga en cuenta que convertir un número binario a hexadecimal supone dividir su expresión en binario en grupos de 4 bits. .

5. **(Entregable en un solo fichero fuente junto con la función del apartado 4)** Empleando la función del apartado 4, escriba ahora un programa que lea un número por teclado en decimal y escriba por pantalla ese mismo número pero expresado en hexadecimal. Para la impresión por pantalla solo debe emplear la función del sistema que imprime cadenas de caracteres (`syscall 4`).

Pruebe el programa con los números 0, 1, -1, 130, 511, -2048, 32 769, 2 147 483 647 y -2 147 483 648.