

Práctica 4. Conversión de datos II: hexadecimal a binario

En esta práctica haremos la conversión contraria a la efectuada en la práctica anterior: convertiremos una cadena que contenga un número hexadecimal a binario para que se pueda depositar en un registro y así hacer operaciones con él.

1. Debe escribir una función con un solo parámetro:

- \$a0: Dirección de la cadena donde está el número expresado en hexadecimal.

La función debe convertir el número escrito en hexadecimal y codificado en ASCII situado en la cadena cuya dirección se encuentra en el registro \$a0 a binario y devolver su valor en el registro \$v0.

2. Para comprobar la función escrita en el apartado anterior:

- Escriba un programa que defina en memoria una cadena con un número hexadecimal y la convierta a binario mediante la función anterior, después debe imprimir por pantalla ese mismo número expresado en decimal utilizando la función del sistema para imprimir números enteros en decimal.
- Pruebe el programa con las cadenas "ff", "00ffff", "7ba ", "ffffff", "7ffffff" y "80000000". Comente en el diario los resultados obtenidos.
- Modifique el programa del apartado 2.a para que la cadena que contiene el número en hexadecimal se pida por teclado ¿Funciona todo correctamente? ¿Qué se puede hacer al respecto?
- Empleando la función del apartado 1 escriba ahora un programa que pida por teclado un patrón de bits expresado en hexadecimal, interprete ese patrón como un número representado en complemento a dos e imprima por pantalla en decimal la división entera resultante de interpretar ese dividir ese número por 16. Emplee la operación de desplazamiento para efectuar la división.

3. Un problema que tiene este tipo de conversión es que la cadena de entrada puede ser errónea porque sea demasiado larga, de forma que el número en binario no quepa en el registro. También puede ocurrir que contenga caracteres que no correspondan a dígitos hexadecimales.

- Modifique ahora la función escrita en el apartado 1 para que devuelva el resultado en el registro \$v1 y en el \$v0 devuelva un código de error, con los siguientes valores:
 - 0: Todo es correcto.
 - 1: Cadena demasiado larga.
 - 2: Carácter incorrecto.
 - etc.

Observe que este tipo de funciones nunca deben, por sí mismas, imprimir mensajes de error por pantalla, sino que deben pasar el código de error al programa principal para que sea él quien imprima los mensajes.

- Incluya en la función las instrucciones necesarias para que admita dígitos hexadecimales alfabéticos tanto con mayúsculas como en minúsculas.
- Reescriba el programa del apartado 2.d, empleando la función del apartado 3.b, de forma que admita el número hexadecimal tanto con mayúsculas como con minúsculas y muestre mensajes de error por pantalla si la cadena introducida no es correcta
- Pruebe ese programa introduciendo las cadenas "FF", "FfFf", "ffffffg", "Affffff", "80000000", "FFffffff0" y "Affffffg". Comente en el diario los resultados obtenidos

4.

- Empleando la función escrita en el apartado 3.b y también la función confeccionada en la práctica anterior, escriba un programa que solicite por teclado un patrón de bits expresado en hexadecimal, lo interprete como un entero representado en complemento a 2, e imprima por pantalla, también en hexadecimal, la división entera resultante de dividir ese número por 8. Emplee la operación de desplazamiento para efectuar la división. El programa debe emitir un mensaje de error si la cadena introducida por teclado no representa un número hexadecimal válido y en ese caso debe pedir el número de nuevo. Para la entrada y salida debe emplear solo funciones que trabajen con cadenas de caracteres. Después, empleando la función del sistema para imprimir números reales, imprima el resultado de interpretar el patrón de bits como un número real representado según la norma IEEE-754 de simple precisión.

- Pruebe el programa anterior con las cadenas "c4033500", "00001Be0" y otros cuatro patrones más y compruebe si los campos de la representación en punto flotante son correctos. Escriba los resultados en el diario.

El fichero entregable de esta práctica contendrá, al menos, el programa fuente correspondiente a la función solicitada en el ejercicio 3.b, el de la función citada correspondiente a la práctica anterior y el programa del ejercicio 4.a.