

# Numeración - Mascaras

## Sistemas de numeración

1. Armar un código en C para mostrar por pantalla los diferentes tamaños de los tipos: char, short, int, long, float y double.
2. ¿Qué significa la palabra unsigned en lenguaje C? ¿Cómo se usa? ¿Modifica el tamaño que ocupa el tipo de datos?
3. ¿Que números distintos se pueden representar con...?
  - a. 8 bits sin signo
  - b. 32 bits sin signo
  - c. 8 bits con signo
  - d. 32 bits con signo
  - e. N bits con signo y sin signo
  - f. 2 dígitos hexadecimales sin signo
  - g. 8 dígitos hexadecimales sin signo
  - h. N dígitos hexadecimales sin signo
  - i. Con 8 bits usando codificación BCD sin signo
4. Implemente un código que convierta un número decimal ingresado por el usuario a binario, octal y hexadecimal.
5. Implemente un código que convierta un número decimal ingresado por el usuario a su representación en BCD y viceversa.

## Mascaras

6. Realice un código que, tomando un uint16 como entrada, devuelva los 8 bits mas significativos.  
Ej: 1001 0101 1110 0100 → 1001 0101
7. Realice un código que, tomando un uint16 como entrada defina si el numero es o no par.
  - a) utilizando el operador %
  - b) sin utilizar el operador %
8. Se tiene un sensor que devuelve los datos en un unit16 de la siguiente manera: temperatura (8 bits) - humedad (8 bits).

Realice un código que se ocupe de, cada 1s, leer estos datos e imprimirlos en pantalla.

9. Realice un código que, a partir de un uint8 devuelva otro con los valores de los primeros y segundos 4 bits invertidos  
Ej. 0001 1010 → 1010 0001
10. Realice un código que, a partir de dos uint8 devuelva un uint16 con los valores de los primeros 2, concatenados.  
Ej 1001 0101 y 1110 0100 → 1001 0101 1110 0100

## Ejercicio Integrador

11. Realice un código que
  - a. Pida al usuario que ingrese dos numeros
    - i. El primero en Hexadecimal (Input → 0xFAB o 10E.)
    - ii. El segundo en binario (mascara)
  - b. Convierta ambos números ingresados al resto de los sistemas de numeración. (Binario, hex, decimal y BCD)
  - c. Pida al usuario que elija entre las operaciones binarias AND y OR.
  - d. Solicite al usuario que prediga, en binario, el output de esa operacion.  
(i.e Prediccion → 0b101010 o 101010)
  - e. Realice la operación, imprima el resultado, comparelo con la predicción e indique cuantos bits errados hay en la predicción. C = Correcto, E=Errado  
Ejemplo del output:

```
Prediccion  0101 0110
Output      0001 1110
-----
(2 Errados) CECC ECCC
```

Consideraciones: El usuario puede ingresar los números hexadecimales y binarios con los sufijos 0x y 0b, respectivamente, o no hacerlo. Asegúrese que su código sea capaz de manejar ambas situaciones.