

1. Cree un programa que tome un valor entero desde el teclado y lo imprima en la pantalla.
2. Modifique el programa anterior para que nos diga si el entero ingresado es par o impar. **Ayuda:** El operador % devuelve el resto de la división entre 2 enteros. En C y C++ se toma, por convención, que 0 es equivalente a FALSO, y cualquier entero (positivo o negativo) distinto de 0 es equivalente a VERDADERO.
3. Cree un programa que tome dos valores enteros desde el teclado y nos diga si son múltiplos.
4. Encontrar los efectos de embeber las secuencias: `\n`, `\t`, `\b`, `\r` en medio del string "**Hola Mundo**".
5. Qué salida espera encontrar si compila y ejecuta el siguiente programa, verifíquelo.

```
#include "icom_helpers.h"

int main (void)
{
    cout << "Testing...";
    cout << "...1";
    cout << "...2";
    cout << "...3";
    cout << '\n';
    return 0;
}
```

6. ¿Cuál sería la salida del siguiente programa? Verifíquelo.

```
#include "icom_helpers.h"

int main (void)
{
    char c, d;
    c = 'd';
    d = c;
    cout << "d = " << d << '\n';
    return 0;
}
```

7. Escriba un programa que evalúe el polinomio  $3x^3 - 5x^2 + 6$  para  $x = 2.55$ . Modifique el programa para que se solicite el valor de  $x$ .
8. Cree un programa que defina una variable de tipo double, y nos diga la dirección de memoria en que se guardó. **Ayuda:** El operador & aplicado a una variable (ej: &var) devuelve su dirección de memoria.
9. Cree un programa que defina una variable X de tipo double inicializada con el valor 1.9 y una variable Y de tipo int. Copie el valor de X en la porción de memoria reservada para Y. Luego, copie el valor de Y nuevamente en la porción de memoria reservada para X. Imprima en pantalla el valor final de X. ¿Cómo justifica el resultado?
10. Para expresar el color de un píxel en una imagen es común empaquetar las intensidades de las componentes roja, verde y azul (RGB) dentro de un número entero sin signo, en donde los bits 0-7 corresponden a la componente R, los bits 8-15 a la componente G y los bits 16-23 a la componente B. (Los bits 24-31 no son utilizados)
  - a) Realice un programa que solicite las componentes RGB de un píxel (con intensidades entre 0 y 255 para cada componente) e imprima su color como un entero utilizando el empaquetamiento citado.
  - b) Realice un programa que reciba un color empaquetado e imprima las componentes RGB del píxel.
11. Modifique el UDT **Complejo** definido en la teórica (ejemplo 3\_11) para que soporte operadores de resta (-) multiplicación (\*) y división (/). Realice un programa que solicite al usuario el ingreso de 2 números complejos

(se deben ir solicitando cada atributo por separado) y luego los sume, multiplique, divida e imprima los resultados obtenidos. Tenga en cuenta que los números complejos son tipos definidos por el usuario por lo que las operaciones sobre este tipo (suma, resta, multiplicación, etc.) deben ser implementadas también por el usuario.

12. Cómo definiría un tipo de dato **Punto3D\_t** que represente un punto en el espacio (coordenadas x, y, z). Realice un programa que pida al usuario 2 puntos en el espacio (representados por instancias de **Punto3D\_t** y que calcule e imprima la distancia entre ellos.