

# ESTRUCTURAS REPETITIVAS

## Ejercicio 1: Sumatoria de números primos en un rango

**Enunciado del proyecto:** Escribir un programa que solicite dos números y calcule la sumatoria de los números primos que existen entre esos dos valores.

### Requerimientos funcionales:

- Solicitar dos números al usuario.
- Calcular y mostrar la sumatoria de los números primos en el rango.
- 

## Ejercicio 2: Números de Fibonacci hasta N términos

**Enunciado del proyecto:** Implementar un programa que genere la secuencia de Fibonacci hasta un número n de términos ingresado por el usuario.

### Requerimientos funcionales:

- Solicitar un número n al usuario.
- Mostrar los primeros n términos de la secuencia de Fibonacci.

## Ejercicio 3: Factorial de números grandes

**Enunciado del proyecto:** Escribir un programa que calcule la factorial de un número grande (por ejemplo, 100) utilizando estructuras repetitivas y el tipo de datos BigInteger.

### Requerimientos funcionales:

- Calcular el factorial de un número grande.
- Mostrar el resultado.

## Ejercicio 4: Inversión de un número

**Enunciado del proyecto:** Crear un programa que invierta los dígitos de un número entero ingresado por el usuario.

### Requerimientos funcionales:

- Solicitar un número entero al usuario.
- Mostrar el número invertido.

## Ejercicio 5: Suma de matrices NxN

**Enunciado del proyecto:** Escribir un programa que solicite dos matrices de tamaño N x N y luego realice la suma de las dos matrices.

**Requerimientos funcionales:**

- Solicitar el tamaño N de las matrices.
- Mostrar la suma de las dos matrices.

**Ejercicio 6: Número perfecto**

**Enunciado del proyecto:** Implementar un programa que encuentre y muestre todos los números perfectos entre 1 y 10,000.

**Requerimientos funcionales:**

- Mostrar todos los números perfectos en el rango especificado.

**Ejercicio 7: Matriz de espiral**

**Enunciado del proyecto:** Crear un programa que imprima una matriz cuadrada de tamaño  $n \times n$  en forma de espiral.

**Requerimientos funcionales:**

- Solicitar el tamaño de la matriz.
- Mostrar la matriz en forma de espiral.

**Ejercicio 8: Verificación de un número Armstrong**

**Enunciado del proyecto:** Escribir un programa que verifique si un número de  $n$  dígitos ingresado por el usuario es un número de Armstrong.

**Requerimientos funcionales:**

- Solicitar un número al usuario.
- Verificar y mostrar si es un número de Armstrong.

**Ejercicio 9: Cálculo de potencias usando multiplicación repetida**

**Enunciado del proyecto:** Crear un programa que calcule la potencia de un número usando multiplicación repetida, es decir, sin utilizar la función `Math.pow()`. El programa debe solicitar una base y un exponente, y luego calcular la potencia utilizando un bucle.

**Requerimientos funcionales:**

- Solicitar al usuario una base y un exponente.
- Calcular la potencia de la base elevada al exponente usando multiplicación repetida.
- Mostrar el resultado.