

## Taller 1

1. Realiza un programa que solicite al usuario dos números enteros y muestre su suma, resta, multiplicación y división.
2. Escribe un programa que pida al usuario un número entero y determine si es par o impar.
3. Crea un programa que calcule y muestre el área y el perímetro de un círculo. El usuario debe proporcionar el radio del círculo.
4. Pide al usuario su edad y muestra un mensaje indicando si es mayor de edad o menor de edad.
5. Realiza un programa que solicite al usuario dos números enteros y determine cuál es el mayor de ellos.
6. Escribe un programa que pida al usuario un número y verifique si es positivo, negativo o cero.
7. Crea un programa que pida al usuario un número entero positivo y muestre su tabla de multiplicar hasta el 10.
8. Realiza un programa que simule un juego de adivinar un número. El programa debe generar un número aleatorio entre 1 y 100, y el usuario debe adivinarlo. El programa debe indicar si el número ingresado es mayor o menor que el número a adivinar.
9. Escribe un programa que solicite al usuario un número entero positivo y calcule su factorial.
10. Ejercicio 10: Crea un programa que muestre los primeros 20 números de la serie Fibonacci. La serie Fibonacci se forma sumando los dos números anteriores para obtener el siguiente número: 0, 1, 1, 2, 3, 5, 8, 13, ...
11. Escribe un programa que calcule el área de un triángulo utilizando la fórmula de Herón. El usuario debe ingresar las longitudes de los tres lados.
12. Realiza un programa que pida al usuario un número entero y determine si es un número primo o no
13. Pide al usuario un número decimal y muestra su valor redondeado a un número específico de decimales
14. Escribe un programa que solicite al usuario un número entero positivo y verifique si es un número perfecto. Un número perfecto es aquel cuya suma de sus divisores propios (excluyendo al propio número) es igual al número.
15. Crea un programa que pida al usuario un número entero y determine si es un número capicúa. Un número capicúa es aquel que se lee igual de izquierda a derecha que de derecha a izquierda.

16. Realiza un programa que calcule e imprima la serie de Fibonacci hasta un número dado ingresado por el usuario.
17. Pide al usuario dos números enteros y muestra todos los números primos que se encuentran en ese rango.
18. Crea un programa que genere una contraseña aleatoria de 8 caracteres que contenga letras mayúsculas, letras minúsculas y dígitos. Puedes utilizar la clase Math para generar números aleatorios y la clase String para manipular la contraseña.
19. Escribe un programa que solicite al usuario su nombre y luego lo imprima en mayúsculas y minúsculas
20. Realiza un programa que solicite al usuario una cadena y luego invierta su orden
21. Pide al usuario una cadena y muestra cuántas veces aparece una letra específica en ella
22. Escribe un programa que solicite al usuario una frase y verifique si es un palíndromo (se lee igual de izquierda a derecha que de derecha a izquierda)
23. Crea un programa que pida al usuario una oración y muestre cuántas palabras contiene
24. Realiza un programa que solicite al usuario una cadena y reemplace todas las apariciones de una letra específica por otra
25. Pide al usuario una frase y muestra cada palabra por separado
26. Escribe un programa que pida al usuario una cadena y muestre cuántos caracteres tiene sin contar los espacios en blanco
27. Crea un programa que solicite al usuario una frase y luego muestre las palabras ordenadas alfabéticamente
28. Suma de elementos: Escribe un programa que calcule la suma de todos los elementos en un arreglo de enteros.
29. Encontrar el número más grande: Crea un programa que encuentre el número más grande en un arreglo de enteros.
30. Eliminar duplicados: Diseña un programa que elimine los elementos duplicados de un arreglo.
31. Ordenar elementos: Implementa un algoritmo de ordenamiento (por ejemplo, el algoritmo de burbuja o el de selección) para ordenar un arreglo de enteros de manera ascendente.
32. Buscar un elemento: Escribe un programa que busque un número específico en un arreglo y muestre su índice (o un mensaje si no se encuentra).
33. Frecuencia de elementos: Escribe un programa que cuente la frecuencia de cada elemento en un arreglo.



34. Rotación de elementos: Implementa un programa que rote los elementos de un arreglo hacia la izquierda o la derecha una cierta cantidad de posiciones.
35. Tabla de multiplicar: Crea un programa que imprima las tablas de multiplicar del 1 al 10. Usa para esto una matriz