



1. Factorial

Implemente un programa que reciba un número entero positivo n de la consola y calcule el factorial de ese número.

El factorial de un número n (denotado como $n!$) se define como el producto de todos los números enteros positivos desde 1 hasta n , o lo que es lo mismo:

$$n! = \begin{cases} 1 & \text{si } n = 0 \\ n \cdot (n-1)! & \text{si } n > 0 \end{cases}$$

2. Suma de impares

Implemente un programa que reciba un entero n e imprima la suma de los primeros n números impares.

3. Mayor, menor y promedio de forma perezosa

Implemente un programa que lea una secuencia de números de la consola (uno por línea) hasta que se escriba una línea en blanco y de estos imprimir:

- El mayor
- El menor
- Su promedio

4. Recorriendo arrays

Implemente un método para cada inciso, que reciba un array de enteros y devuelva:

1. El mayor elemento de un array.
2. El segundo menor elemento de un array.
3. Si un número n pertenece al array a .
4. El promedio de todos los elementos de un array.
5. La cantidad de elementos que son mayor que el promedio en un array.

5. Invirtiendo

Implemente un método que dado el array a que recibe como entrada, devuelva otro array con los elementos de a en orden inverso. Ejemplo: recibe: $[2, 7, -9]$ y devuelve $[-9, 7, 2]$.

6. Filtrando Positivos

Implemente un método que reciba un array a y devuelva un nuevo array con todos los elementos positivos del array a .

7. Rotando

Implemente un método que reciba un array a y un entero $veces$ y rote los elementos del array tantas veces como indique el parámetro $veces$. Si $veces$ es positivo, rota los elementos a la derecha; si es negativo, rota los elementos a la izquierda. Si $veces$ es 0, el array no se modifica. Por ejemplo, si rotamos el array $[25, 40, 17, 83, 9]$ 2 veces, obtenemos el array $[83, 9, 25, 40, 17]$, y si lo rotamos -2 veces entonces nos queda $[17, 83, 9, 25, 40]$.

8. Mezcla ordenada

Implemente un método que a partir de los arrays ordenados a y b deberá devolver un nuevo array que sea la mezcla ordenada de estos. Por ejemplo, si el array a es $[23, 40, 83]$ y el array b es $[5, 17, 23, 24, 51]$, entonces el resultado será el array $[5, 17, 23, 23, 24, 40, 51, 83]$.

9. Añadiendo al Final

Implemente un método que reciba un valor val y añada dicho valor al final del array a , devolviendo un nuevo array con el elemento añadido.

10. Insertando

Implemente un método que, dado un entero pos y un valor val , inserte el valor val en la posición pos de a , desplazando los elementos existentes hacia la derecha, devuelva un nuevo array con el elemento insertado.

11. Eliminando

Implemente un método que, dado un entero pos referente a determinada posición del array a , elimine el elemento que se encuentra en dicha posición, devuelva un nuevo array sin ese elemento.

12. Representación binaria

1. Implemente un método que reciba un número entero no negativo y devuelva un string con su representación binaria.
2. Implemente un método que convierta un número de binario a decimal. (El número binario está representado por un string compuesto de 0s y 1s).

13. Es primo

Determinar si un número entero positivo es primo. Un número es primo si solo tiene dos divisores: 1 y el propio número.

14. Número perfecto

Determinar si un número entero positivo es perfecto. Un número es perfecto si la suma de sus divisores propios es igual a él. Ejemplo: $28 = 1 + 2 + 4 + 7 + 14$

15. Substring

Un substring (subcadena) es una secuencia de caracteres que aparece consecutivamente en una cadena mayor. Dados los string `s` y `x`, implemente un método que diga si `x` es substring de `s`. Ejemplo: `""` es substring de toda cadena, `"a"` es substring de `"casa"`, `"asap"` no es substring de `"casa"`.

16. Es palíndromo

Implemente un método que determine si `s` es palíndromo (se lee igual al derecho que al revés). Ejemplos: `ana`, `anitalavalatina`, `zz`.

17. Menor sufijo para ser palíndromo

Implemente un método que compute el menor string `t` tal que `s + t` es palíndromo.

18. Ordenando

Implemente un método que reciba un array de enteros `a` y devuelva un nuevo array con los mismos elementos ordenados de menor a mayor.

19. Mediana

Implemente un método que reciba un array de enteros y devuelva el elemento mediana. La mediana de un array es el elemento que tiene la misma cantidad de elementos mayores y menores en el array.

Ejemplos:

- Para el array `[3, 5, 2, 8, 1]`: La cantidad de números menores que 3 es la misma que la cantidad de números mayores que 3, por lo que 3 es la mediana.

- Para el array [3, 5, 2, 8]: El tamaño del array es par, por tanto definiremos la mediana como el elemento que tiene $\frac{n}{2}$ elementos menores y $\frac{n}{2} - 1$ elementos mayores, por lo tanto, la mediana es 5.