



1. Introducción a C++

Ejercicio 1. Plataforma online de compilación C++ En el link <https://www.onlinegdb.com/>, copiar y ejecutar el siguiente programa:

```
#include <iostream>

int f(int x){
    return x+1;
}

int main() {
    std::cout << "El resultado es: " << f(10) << std::endl;
    return 0;
}
```

Ejercicio 2. Modificar el programa anterior para que f tome dos parámetros de tipo `int` y los sume.

Ejercicio 3. Modificar el programa anterior para que f tome dos parámetros x e y de tipo `int` y los sume sólo si $x > y$, en caso contrario el resultado será el producto.

Ejercicio 4. Escribir la función que dado $n \in \mathbb{N}$ devuelve si es primo. Recuerden que un número es primo si los únicos divisores que tiene son 1 y el mismo.

Iteración vs Recursión

Los siguientes ejercicios deben ser implementados primero en su versión **recursiva**, luego iterativa utilizando **while** y por último iterativa utilizando **for**.

Ejercicio 5. Escribir la función de Fibonacci que dado un entero n devuelve el n -ésimo número de Fibonacci. Los números de Fibonacci empiezan con $F_0 = 0$ y $F_1 = 1$. $F_n = F_{n-1} + F_{n-2}$

Ejercicio 6. Escribir la función que dado $n \in \mathbb{N}$ devuelve la suma de todos los números impares menores que n .

Ejercicio 7. Escribir la función `sumaDivisores` que dado $n \in \mathbb{N}$, devuelve la suma de todos sus divisores entre $[1, n]$.

■ **Hint:** Recordar que para la versión recursiva es necesario implementar `divisoresHasta`

Ejercicio 8. Escribir la función combinatorio que dados $n, k \in \mathbb{N}$ compute el combinatorio: $\binom{n}{k}$. Para la versión recursiva usar la igualdad $\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$

Calcular el combinatorio $\binom{100}{5}$. ¿Funciona para ambas versiones? ¿Qué sucede con la versión iterativa?

Ejercicio 9. ¿Es mejor programar utilizando algoritmos recursivos ó iterativos? ¿Es mejor usar **while** o **for**?