



### Problema 1.º - Secuencia de Collatz

El siguiente programa solicita al usuario un número inicial y, a continuación, escribe en la pantalla la secuencia de Collatz<sup>1</sup> que comienza en dicho valor. ¿Qué escribiría en la pantalla cuando en una ejecución el usuario introdujera 13? ¿Y si introdujera 6?

```
#include <iostream>
using namespace std;

/*
 * Programa que pide al usuario un número natural y escribe en la pantalla la secuencia de
 * Collatz generada por dicho número.
 */
int main() {
    cout << "Escriba un número natural: ";
    unsigned n;
    cin >> n;

    cout << "La secuencia de Collatz generada por " << n << " es:" << endl;
    cout << n;
    while (n != 1) {
        if (n % 2 == 0) {
            n = n / 2;
        } else {
            n = 3 * n + 1;
        }
        cout << ", " << n;
    }
    cout << endl;
}
```

### Problema 2.º - Divisores

Diseña un programa que solicite al usuario un número entero positivo y escriba en la pantalla un listado de sus divisores, a razón de uno por línea. Se muestra a continuación un ejemplo de ejecución del programa solicitado cuando el usuario introduce el número 28:

```
Escriba un número entero positivo: 28

DIVISORES DE 28:
1
2
4
7
14
28
```

### Problema 3.º - Números perfectos

Parte del control de prácticas del 24-6-2004

Diseña un programa que solicite al usuario un número entero positivo (asegurándose de que es positivo, volviendo a pedir un nuevo número si es preciso) y que le indique si se trata de un número perfecto o no. Un número perfecto es aquel que es igual a la suma de sus divisores propios (es decir, sus divisores positivos menores que él mismo). Así, 6 es un número perfecto, porque sus divisores propios son 1, 2 y 3; y  $6 = 1 + 2 + 3$ . Los siguientes números perfectos son 28, 496 y 8128.

---

<sup>1</sup> [https://en.wikipedia.org/wiki/Collatz\\_conjecture](https://en.wikipedia.org/wiki/Collatz_conjecture)



# Problemas de Programación 1

## 4. Instrucciones simples y estructuradas Composición iterativa

A continuación, se muestran ejemplos de ejecución del programa solicitado:

Escriba un número entero positivo: 496  
496 es un número perfecto.

Escriba un número entero positivo: 32  
32 no es un número perfecto.

Escriba un número entero positivo: -8  
El número debe ser entero positivo: -125  
El número debe ser entero positivo: 8128  
8128 es un número perfecto.

### Problema 4.º - Múltiplos de 3 y 5

Problema 1 de Project Euler<sup>2</sup>

Si se listan todos los números naturales inferiores a 10 que son múltiplos de 3 o 5, se obtienen los números 3, 5, 6 y 9. La suma de estos números es 23. Halla la suma de todos los múltiplos de 3 o 5 menores que 1000.

### Problema 5.º - Suma de cuadrados vs. cuadrado de la suma

Problema 6 de Project Euler<sup>3</sup>

La suma de los cuadrados de los diez primeros números naturales es:

$$1^2 + 2^2 + \dots + 10^2 = 385$$

El cuadrado de la suma de los diez primeros números naturales es:

$$(1 + 2 + \dots + 10)^2 = 55^2 = 3025$$

Por lo tanto, la diferencia entre la suma de los cuadrados de los diez primeros números naturales y el cuadrado de su suma es  $3025 - 385 = 2640$ .

Halla la diferencia entre la suma de los cuadrados de los cien primeros números naturales y el cuadrado de su suma.

<sup>2</sup> <https://www.projecteuler.net/problem=1>

<sup>3</sup> <https://www.projecteuler.net/problem=6>