






# Ejercicios Bucles 'Básicos'

[Descargar estos ejercicios](#)

## Índice

- [Ejercicio 1](#)
- [Ejercicio 2](#)
- [Ejercicio 3](#)
- [Ejercicio 4](#)
- [Ejercicio 5](#)
-  [Ejercicio 6](#)
- [Ejercicio 7](#)
-  [Ejercicio 8](#)
- [Ejercicio 9](#)
-  [Ejercicio 10](#)
-  [Ejercicio 11](#)
- [Ejercicio 12](#)
-  [Ejercicio 13](#)
- [Ejercicio 14](#)

### Ejercicio 1

Programa que calcula y muestra la **suma** y el **producto** de los **10 primeros** números naturales.

**Nota:** Se deben usar acumuladores para resolverlo.

**Ejemplo de ejecución:**

SUMA: 55

PRODUCTO: 3628800

## Ejercicio 2

Programa que lee 100 números y cuenta cuántos de ellos son positivos ( $n > 0$ ).

**Nota:** Se deben usar contadores para resolverlo.

### Ejemplo de ejecución:

```
Introduzca valor 1: 10
Introduzca valor 2: -5
Introduzca valor 3: 2
Introduzca valor 4: 4
Introduzca valor 5: -7
Introduzca valor 6: -9
Introduzca valor 7: -3
Introduzca valor 8: -6
Introduzca valor 9: -8
Introduzca valor 10: 1
Números positivos introducidos: 4
```

## Ejercicio 3

Programa que lea números hasta que se introduzca un cero y escriba la media de los números leídos, **sin incluir el 0 en el conteo de números**.

**Nota:** Se deben usar contadores y acumuladores para resolverlo.

**Ejemplo de ejecución:**

```
Introduzca un valor: 1
Introduzca un valor: 2
Introduzca un valor: 3
Introduzca un valor: 0
MEDIA: 2,00
```

## Ejercicio 4

Programa que lea notas y que termine con el valor -1.

- Las notas deben estar incluidas en el rango que va de 0 al 10, descartando y avisando del error si no es una nota permitida.
- La salida nos mostrará la cantidad de dieces que se han introducido.

**Ejemplo de ejecución:**

```
Introduzca la nota número 1: 5
Introduzca la nota número 2: 10
Introduzca la nota número 3: 2
Introduzca la nota número 4: 10
Introduzca la nota número 5: 11
Nota incorrecta
Introduzca la nota número 5: -1

Ha(n) habido 2 sobresaliente(s)
MEDIA: 2,00
```

## Ejercicio 5

Programa que lee una secuencia de 100 números y nos dice cuántos hay positivos, cuántos negativos y cuantos ceros.

**Nota:** Para hacer las pruebas puedes reducir el número de entradas.

## Ejercicio 6

Programa que lee una secuencia de números no nulos, terminada con la introducción de un 0, y muestra el mayor de la secuencia.

### Ejemplo de ejecución:

```
Introduzca valor: 7
Introduzca valor: 3
Introduzca valor: 8
Introduzca valor: 1
Introduzca valor: 0
El mayor de los valores introducidos es: 8
```

## Ejercicio 7

Programa que obtenga el producto de dos números enteros positivos mediante sumas sucesivas. Esto es, para calcular  $2 * 5$  haga  $2 + 2 + 2 + 2 + 2$

### Ejemplo de ejecución:

```
Introduzca operador 1: -1
Introduzca operador 2: 5
ERROR: Sólo se permiten números positivos

Introduzca operador 1: 4
Introduzca operador 2: 3
Sumando....
4 x 3 = 12
```

## Ejercicio 8

Programa que obtenga el cociente y el resto de dividir dos números enteros positivos utilizando restas restas. Por ejemplo, para calcular  $n / 2$  haga  $n -= 2$  mientras  $n >= 2$  y cuente el número de veces que ha restado.

### Ejemplo de ejecución:

```
Introduzca dividendo: 10
Introduzca divisor: -2
ERROR: Sólo se permiten números positivos

Introduzca dividendo: 10
Introduzca divisor: 2

10 / 2 = 5
Resto: 0
```

## Ejercicio 9

Haz un programa que muestre en pantalla la **tabla de códigos UTF-8**

HEX	DEC	CAR
0x20	32	
0x22	34	"
0x24	36	\$
0x26	38	&
0x28	40	(
0x2a	42	*

... Así hasta 255 (256 caracteres).

**Nota:** Se parará la pantalla cuando sea necesario para que se puedan ver todos los caracteres, para ello podemos usar un `Console.ReadKey()` .

## Ejercicio 10

**Programa que determina si dos números enteros positivos son amigos**

Dos números son amigos si la suma de los divisores del primer número excepto él mismo, es igual al segundo número, y viceversa. Puedes saber un poco más de la historia de esta relación entre números leyendo la entrada en la [Wikipedia](#).

**Ejemplo de ejecución:**

```
Introduzca valor 1: -3
Introduzca valor 2: 5
ERROR: Sólo se permiten números positivos
```

```
Introduzca valor 1: 3
Introduzca valor 2: 5
```

```
Los valores no son amigos
```

### Ejemplo de ejecución 2:


```
Introduzca valor 1: 220
Introduzca valor 2: 284
```

```
Los valores son amigos
```

## Ejercicio 11

### Simulación de una calculadora

- Realiza un programa que sea capaz de **sumar**, **restar**, **multiplicar** y dividir.
- El programa presentará un menú con las cuatro operaciones que puede realizar.
- Saldrá del programa con la tecla **ESC**.

 **Tip:** Para controlar que se ha pulsado la tecla ESC podemos usar la instrucción `Console.ReadKey()`

en un esquema similar a la siguiente propuesta dentro del bucle...

```
// ... Mostrar menú
Console.Write("Pulsa una opción: ");
var tecla = Console.ReadKey(); // Esta instrucción se para hasta que se pulse una tecla.

// Introducción de operandos ...
double o1 = ....;
double o2 = ....;

bool esEscape = tecla.Key == ConsoleKey.Escape; // Comprueba si la tecla pulsada es esca

if (!esEscape)
{
    char caracterAsociadoALaTecla = tecla.KeyChar; // Obtener el caracter asociado a la
    double resultado = caracterAsociadoALaTecla switch
    {
        '1' => o1 + o2,
        //...
        _ => double.NaN
    };

    // Mostrar resultado.
}
```

## Ejercicio 12

Mostrar los **múltiplos de 7** que hay entre **7** y **112**.

**Ejemplo de ejecución :**

```
7 14 21 28 35 42 49 56 63 70 77 84 91 98 105
```

## Ejercicio 13

Pide un número, por ejemplo el 4, y saca en pantalla 1223334444.

**Nota:** Deber usar bucles `for` para hacerlo.

**Ejemplo de ejecución:**



Introduzca un numero entero: 8  
122333344445555566666677777788888888

## Ejercicio 14

Lee un número y escribe la suma de sus dígitos.

**Nota:** No puedes usar **arrays** porque aún no los hemos visto.

**Ejemplo de ejecución:**

Introduzca un numero: 1234  
Número de dígitos: 4  
Suma de dígitos: 10