

8. Bucle while (1)

1. Escribe un programa que pida dos números enteros. El segundo número ha de ser mayor que el primero. En caso de no serlo, el programa volverá a solicitar el segundo número hasta que se cumpla la condición. Para terminar el programa debe mostrar los dos números por pantalla.

```
Escribe un número entero: 6
Escribe un número entero mayor que 6: 6
6 no es mayor que 6. Vuelve a intentarlo: 1
1 no es mayor que 6. Vuelve a intentarlo: 8

Los números que has introducido son el 6 y el 8.
Programa finalizado
```

2. Escribe un programa que pida un primer número. A partir de entonces el programa continuará pidiendo números hasta que el número introducido sea menor que el número inicial. En ese momento terminará la ejecución del programa.

```
Escribe un número: 4.5
Escribe un número mayor que 4.5: 5
Escribe otro número mayor que 4.5: 4.5
Escribe otro número mayor que 4.5: 2

2.0 es menor que 4.5.
Programa finalizado
```

3. Escribe el código de un programa que pida un número entero positivo. A continuación, el programa debe de pedir por pantalla esa cantidad de números positivos. El programa debe continuar pidiendo números positivos hasta que se hayan introducido la cantidad de números indicada en la primera variable. Como resultado final se indicará cuantos de los números introducidos cumplen la condición de ser positivos.

```
¿Cuántos números positivos vas a introducir?: 3

Escribe el primer número: 10
Escribe otro número: 2
Escribe otro número: -5
Escribe otro número: 2

Has escrito 3 números positivos. Programa finalizado.
```

```
¿Cuántos números positivos vas a introducir?: 0
La cantidad debe ser mayor que 0. Inténtalo de nuevo: 1

Escribe el primer número: 1

Has escrito 1 número positivo. Programa finalizado.
```

4. Escribe un programa que pida números positivos. El programa debe finalizar cuando el usuario introduzca un número negativo, en ese momento se debe mostrar la suma de todos los números positivos que se hayan introducido.

```
Escribe un número: 11
Escribe otro número: 3
Escribe otro número: 7
Escribe otro número: -1

Los números positivos introducidos suman 21.
Programa terminado
```

```
Escribe un número: -9

Los números positivos introducidos suman 0.
Programa terminado
```

5. Escribe un programa que pida por pantalla un valor positivo. A continuación, el programa pedirá números hasta que la suma de todos los números introducidos supere el valor del número inicial, mostrando el resultado de la suma.

```
Escribe la cantidad límite: 13

Escribe un número: 5.2
Escribe otro número: -1
Escribe otro número: 9

Has superado el límite. La suma de los números introducidos es 13.2.
Programa finalizado
```

```
Escribe la cantidad límite: -3
El número debe ser mayor que 0. Inténtalo de nuevo: 0
El número debe ser mayor que 0. Inténtalo de nuevo: 5.5

Escribe un número: 7.2

Has superado el límite. La suma de los números introducidos es 7.2.
Programa finalizado
```

6. Crea un programa que debe comenzar pidiendo dos números enteros (mínimo y máximo). A continuación, el programa debe ir pidiendo números enteros situados entre esos valores máximo y mínimo. El programa concluye cuando escribamos un número no comprendido entre los dos valores iniciales, mostrando la cantidad de números escritos incluidos en el intervalo abierto definido.

```
Escribe un número: 6
Escribe un número mayor que 6: 4
4 no es mayor que 6. Inténtalo de nuevo: 50

Escribe un número entre 6 y 50: 45
Escribe otro número entre 6 y 50: 6
Escribe otro número entre 6 y 50: 4

Ha escrito 2 números entre 6 y 50.
Programa terminado
```

```
Escribe un número: 8
Escribe un número mayor que 8: 20
```

```
Escribe un número entre 8 y 20: 45
```

```
No has escrito ningún número entre 6 y 50.  
Programa terminado
```

```
Escribe un número: -10
```

```
Escribe un número mayor que -10: -5
```

```
Escribe un número entre -10 y -5: -8
```

```
Escribe un número entre -10 y -5: 100
```

```
Ha escrito 1 número entre -10 y -5.
```

```
Programa terminado
```

7. Escribe el código que pida números múltiplos de cinco mientras el usuario indique que quiere seguir introduciendo números. Para indicar que queremos introducir un nuevo número, el usuario deberá contestar S a una pregunta.

```
Escribe un número múltiplo de 5: 10
```

```
¿Quieres escribir otro número múltiplo de 5? (S/N) S
```

```
Escribe otro número múltiplo de 5: 15
```

```
¿Quieres escribir otro número múltiplo de 5?(S/N) N
```

```
Has escrito 2 números múltiplos de 5.
```

```
Programa terminado
```

```
Escribe un número múltiplo de 5: 6
```

```
6 no es un número múltiplo de 5. Inténtalo de nuevo: 35
```

```
¿Quieres escribir otro número múltiplo de 5? (S/N) x
```

```
Has escrito 1 número múltiplo de 5.
```

```
Programa terminado
```

8. Realiza la descomposición en factores primos de un número dado por teclado.

```
Escribe un número mayor que 1: 500
```

```
Descomposición en factores primos: 2 2 5 5 5
```

```
Escribe un número mayor que 1: 521
```

```
Descomposición en factores primos: 521
```

```
Escribe un número mayor que 1: 1
```

```
1 no es un número mayor que 1. Inténtalo de nuevo: 720
```

```
Descomposición en factores primos: 2 2 2 2 3 3 5
```