

Ejercicios Bucles - 01

1. 01-Bucles → Escribe el código necesario para que se muestre por pantalla las siguientes cinco series aritméticas dado un número **n**. Ejemplo con n=10:

```
1 2 3 4 5 6 7 8 9 10
2 4 6 8 10 12 14 16 18 20
20 23 26 29 32 35 38 41 44 47
10 14 18 22 26 30 34 38 42 46
45 40 35 30 25 20 15 10 5 0
```

Puedes repetir el ejercicio con while

2. 02-Bucles → Escribe el código necesario para generar las siguientes siete secuencias de números utilizando bucles for.

```
1 4 9 16 25 36 49 64 81 100
2 5 10 17 26 37 50 65 82 101
8 27 64 125 216 343 512 729 1000 1331
2 6 12 20 30 42 56 72 90 110
1 10 100 1000 10000 100000 1000000 10000000 100000000 1000000000
1 0.1 0.01 0.001 0.0001 ...
-1 1 -1 1 -1 1 -1 1 -1 1
```

3. 03-Bucles → Escribir un programa que muestre los N primeros números naturales

3.1 Utilizando for

3.2 Utilizando while

4. 04-Bucles → Escribir un programa que muestre los N primeros números naturales pares

1.1. Utilizando for

1.2. Utilizando while

5. 05-Bucles → Haz un programa que vaya leyendo números y mostrándolos por pantalla hasta que el usuario introduzca un número negativo. En ese momento, el programa mostrará un mensaje de despedida y finalizará su ejecución.

6. 06-Bucles → Haz un programa que vaya leyendo números hasta que el usuario introduzca un número negativo. En ese momento, el programa mostrará por pantalla el número mayor de cuantos ha visto.

7. 07-Bucles → Haz un programa que muestre la tabla de multiplicar de un número introducido por teclado por el usuario.
8. 08-Bucles → Haz un programa que muestre la raíz n-ésima de un número leído por teclado, para n tomando valores entre 2 y 100.
9. 09-Bucles → Escribir un programa que calcule la suma de los N primeros números naturales
- 1.3. Utilizando for
 - 1.4. Utilizando while

10. 10-Bucles → Escribir un programa que, para un número N introducido por teclado, calcule la suma de:

1.5. $1 + 1/2 + 1/3 + \dots + 1/N$

1.6. $1 + 4 + 9 + 16 + \dots + N^2$

1.7. $1/2 + 1/4 + 1/8 + 1/16 + 1/32 + \dots + 1/2^N$

1.8. $1/2 + 2/2^2 + 3/2^3 + \dots + n/2^n$.

11. 11-Bucles → Utilizando la serie de Taylor para e^x , hacer un programa que calcule dicha función con una precisión dada por el número de términos que se suman. Cuando $x=1$ obtenemos el valor de e .

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!} + \dots, \forall x \in \mathbb{R}$$

12. 12-Bucles → Dada la siguiente serie matemática:

$$\begin{aligned} a_1 &= 0 \\ a_2 &= 1 \\ a_i &= a_{i-1} + 2 \cdot a_{i-2} \text{ para } i \geq 3 \end{aligned}$$

Determinar cuál es el valor y el primer término cuyo valor sea mayor o igual a 2000 (i tal que $a_i \geq 2000$).

13. 13-Bucles → Utilizando un bucle for y un único if sencillo con la condición ($\text{numero} \% 3 == 0$), crea un programa que pida un número entero y muestre una lista con todos los números enteros menores o iguales que el introducido que no sean múltiplos de 3.

Introduce un número entero: 4

1
2
4

14. 14-Bucles → Repite el ejercicio anterior sustituyendo el bucle for por un bucle while.

15. 15-Bucles → Escribe el código necesario para obtener la siguiente salida:

```
0123456
1234567
2345678
```

Pista:

Pista:
Necesitamos dos bucles anidados (uno dentro del otro), de manera que el bucle externo controle las filas que se van a escribir y el interno para escribir los números.
Fijarse que la fila 0 empieza por 0, la fila 1 por 1 y así sucesivamente

16. 16-Bucles → Escribir un programa que dado un valor no negativo n , visualice la siguiente salida:

```
1 2 3 ..... n-1 n
1 2 3 ..... n-1
.....
1 2 3
1 2
1
```

17. 17-Bucles → Crea un programa que pida dos números enteros. Utilizando caracteres producto (*) el programa ha de dibujar un rectángulo. La longitud de la base viene dada por el valor del primer número y la de la altura por el valor del segundo

```
--- DIBUJO DE RECTANGULOS ---
¿Qué dimensiones tiene el rectángulo?
- Base: 6
- Altura: 3
* * * * *
* * * * *
* * * * *
```

18. 18-Bucles → Modifica el programa anterior de tal forma que los caracteres utilizados en el perímetro del rectángulo sean caracteres producto (*) y los interiores sean puntos (.)

```
--- DIBUJO DE RECTANGULOS ---
¿Qué dimensiones tiene el rectángulo?
- Base: 5
- Altura: 4
* * * * *
* + + + *
* + + + *
* + + + *
* * * * *
```

19. 19-Bucles → Modifica el programa anterior de tal forma que los caracteres utilizados en el perímetro del rectángulo sean caracteres producto (*) y el interior esté vacío.

--- DIBUJO DE RECTANGULOS ---

¿Qué dimensiones tiene el rectángulo?

- Base: 5

- Altura: 4

* * * * *

* *

* *

* * * * *

20. 20-Bucles → Modifica el código del ejercicio anterior para conseguir que el programa pregunte al usuario por el número de rectángulos que se desea dibujar. Sigue el modelo:

--- DIBUJO DE SERIES DE RECTANGULOS ---

¿Qué dimensiones tiene cada rectángulo?

- Base: 5

- Altura: 3

¿Cuántos rectángulos quieres dibujar?: 2

* * * * *

* *

* * * * *

* * * * *

* *

* * * * *

21. 21-Bucles → Modifica el código anterior. Ahora los rectángulos han de estar conectados entre sí: El lado inferior del rectángulo superior sea también el lado superior del rectángulo inferior.

--- DIBUJO DE SERIES DE RECTANGULOS ---

¿Qué dimensiones tiene cada rectángulo?

- Base: 5

- Altura: 3

¿Cuántos rectángulos quieres dibujar? 3

* * * * *

* *

* * * * *

* *

* * * * *

* *

* * * * *

22. 22-Bucles → Compliquemos las cosas un poco más. Partiendo del ejercicio número 4, ahora los rectángulos han de estar a la misma altura y separados entre sí.

--- DIBUJO DE SERIES DE RECTANGULOS ---

¿Qué dimensiones tiene cada rectángulo?

- Base: 5

- Altura: 4

¿Cuántos rectángulos quieres dibujar?: 2

```
* * * * *
*       * *       *
*       * *       *
* * * * *
* * * * *
```

23. 23-Bucles → Modifica el código anterior para que ahora los rectángulos compartan los lados laterales:

--- DIBUJO DE SERIES DE RECTANGULOS ---

¿Qué dimensiones tiene cada rectángulo?

- Base: 5

- Altura: 4

¿Cuántos rectángulos quieres dibujar?: 3

```
* * * * *
*       *       *
*       *       *
*       *       *
* * * * *
```

24. 24-Bucles → Todo junto. Vamos a dibujar una matriz de rectángulos

--- DIBUJO DE SERIES DE RECTANGULOS ---

¿Qué dimensiones tiene cada rectángulo?

- Base: 5

- Altura: 3

- Número de columnas de rectángulos: 3

- Número de filas de rectángulos: 2

```
* * * * *
*       *       *
* * * * *
*       *       *
* * * * *
```

25. 25-Bucles → Escribir un programa que muestre por pantalla la siguiente figura pidiendo el número de filas (Ancho):

Ancho = 5

```
*
**
***
****
*****
```

26. 26-Bucles → Escribir un programa que muestre por pantalla la siguiente figura introduciendo el número de filas (también podría ser pidiendo el ancho, debe de ser impar):

Filas: 5

```
*
***
*****
*****
*****
```

Ancho: 9

```
*
***
*****
*****
*****
```

27. 27-Bucles → Realizar un programa que dibuje la siguiente figura introduciendo el ancho:

Ancho: 5

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

Este ejercicio es posible realizarlo con solo un bucle anidado, para ello necesitamos la función floor

28. 28-Bucles → Escribe el código para conseguir:

```
0 1 2 3 4 5 6 7 8 9
 0 1 2 3 4 5 6 7 8
   0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6
     0 1 2 3 4 5
      0 1 2 3 4
       0 1 2 3
        0 1 2
         0 1
          0
```

Pista:

Pista:
 Necesitamos dos bucles anidados (uno dentro del otro), de manera que el bucle externo controle las filas que se van a escribir y el interno para escribir los números, de esta manera obtenemos esto:

```
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8
0 1 2 3 4 5 6 7
0 1 2 3 4 5 6
0 1 2 3 4 5
0 1 2 3 4
0 1 2 3
0 1 2
0 1
0
```

Ahora hemos de añadir un bucle antes del bucle interno para escribir los espacios en blanco.

29. 29-Bucles → Realizar un programa que dibuje la siguiente figura introduciendo el ancho:

Ancho: 7

```
      *
     * * *
    * * * * *
   * * * * * * *
```

30. 30-Bucles → Realizar un programa que dibuje la siguiente figura introduciendo el alto:

Alto: 5

```
* * * * * * * *
* * * *   * * * *
* * * *   * * * *
* * *     * * *
* *       * *
*         *
```

31. 31-Bucles → Realizar un programa que muestre el siguiente triángulo:

```
      1
     232
    34543
   4567654
  567898765
 67890109876
7890123210987
890123454321098
90123456765432109
0123456789876543210
123456789010987654321
```

32. 32-Bucles → Escribir un programa que dado un número nos de el factorial de ese número. El factorial de n se define como el producto de todos los números enteros positivos desde 1 hasta n:

$$n! = 1 \times 2 \times 3 \times 4 \times \dots \times (n-1) \times n$$

$$10! = 1 \times 2 \times 3 \times 4 \times \dots \times 9 \times 10$$

33. 33-Bucles → Escribir un programa que dado un número nos diga si es primo o no ese número.
34. 34-Bucles → Escribir un programa que calcule la suma de los dígitos de un número entero.
Ejemplo: para 3913 debe devolver 16 (3+9+1+3)
35. 35-Bucles → Crea un programa que solicite una palabra por teclado. El programa a continuación debe mostrar cada una de las letras de esa palabra en una fila distinta. Si alguna de las letras es la h, el programa se debe detener.

```
Introduce la palabra que quieres deletrear: PIE
P
I
E
Introduce la palabra que quieres deletrear: Python
P
y
t
```

36. 36-Bucles → Modifica el programa anterior para que en caso de que el programa deletee la palabra completa, al final escriba de nuevo la palabra deletreada.

```
Introduce la palabra que quieres deletrear: PIE
P
I
E
PIE
Introduce la palabra que quieres deletrear: Python
P
y
t
h
o
n
Python
```

37. 37-Bucles → Crea un programa con el siguiente código:

```
for (numero=10; numero<20; numero++):
    for (i=2; i<numero; i++)
        if (numero%i == 0) {
            j=numero/i
            system (numero,'=',i,' x ',j)
            System.out.println(numero + " = " + i + " x " + j)
            break
        }
    else
        print (numero, 'es un número primo')
```

Añade los comentarios necesarios para explicar la función de cada una de las líneas de código

Licencia



Reconocimiento – NoComercial – CompartirIgual (by-nc-sa): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.