

Relación de ejercicios - Bucles

1. Diseña una aplicación que muestre la edad máxima y mínima de un grupo de alumnos. El usuario introducirá las edades y terminará escribiendo - 1.
2. Diseñar un programa que muestre, para cada número introducido por teclado, si es par, si es positivo y su cuadrado. El proceso se repetirá hasta que el número introducido sea 0.
3. Implementar una aplicación para calcular datos estadísticos de las edades de los alumnos de un centro educativo. Se introducirán datos hasta que uno de ellos sea negativo, y se mostrará: la suma de todas las edades introducidas, la media, el número de alumnos y cuántos son mayores de edad.
4. Codificar el juego, el número secreto, que consiste en acertar un número entre 1 y 100 (generado aleatoriamente). Para ello se introduce por teclado una serie de números, para los que se indica: mayor o, menor, según sea mayor o menor con respecto al número secreto. El proceso termina cuando el usuario acierta o cuando se rinde (introduciendo un -1).
5. Un centro de investigación de la flora urbana necesita una aplicación que muestre cuál es el árbol más alto. Para ello se introducirá por teclado la altura (en centímetros) de cada árbol (terminando la introducción de datos cuando se utilice -1 como altura). Los árboles se identifican mediante etiquetas con números únicos correlativos, comenzando en 0. Diseña una aplicación que resuelva el problema planteado.
6. Desarrollar un juego que ayude a mejorar el cálculo mental de la suma. El jugador tendrá que introducir la solución de la suma de dos números aleatorios comprendidos entre 1 y 100. Mientras la solución introducida sea correcta, el juego continuará. En caso contrario, el programa terminará y mostrará el número de operaciones realizadas correctamente.
7. Escribir todos los múltiplos de 7 menores que 100.
8. Pedir diez números enteros por teclado y mostrar la media.
9. Implementar una aplicación que pida al usuario un número comprendido entre 1 y 10. Hay que mostrar la tabla de multiplicar de dicho número, asegurándose de que el número introducido se encuentra en el rango establecido.
10. Pedir un número y calcular su factorial. Por ejemplo, el factorial de 5 se denota 5! y es igual a $5 \times 4 \times 3 \times 2 \times 1 = 120$.
11. Dadas 6 notas, escribir la cantidad de alumnos aprobados, condicionados (nota igual a cuatro) y suspensos.

12. Pedir por consola un número n y dibujar un triángulo rectángulo de n elementos de lado, utilizando para ello asteriscos(*). Por ejemplo, para $n=4$:

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

13. Realiza un programa que nos pida un número n , y nos diga cuántos números hay entre 1 y n que sean primos. Un número primo es aquel que solo es divisible por 1 y por él mismo.
14. Diseña una aplicación que dibuje el triángulo de Pascal, para n filas. Numerando las filas y elementos desde 0, la fórmula para obtener el m -ésimo elemento de la n -ésima fila es:

$$E(n, m) = \frac{n!}{m!(n - m)!}$$

Donde $n!$ es el factorial de n .

Un ejemplo de triángulo de Pascal con 5 filas ($n = 4$) es:

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

15. Para dos números dados, a y b , es posible buscar el máximo común divisor (el número más grande que divide a ambos) mediante un algoritmo ineficiente pero sencillo: desde el menor de a y b , ir buscando, de forma decreciente, el primer número que divide a ambos simultáneamente. Realiza un programa que calcule el máximo común divisor de dos números.
16. Crea un bucle while que genere números aleatorios entre 1 y 100 hasta que se genere un número mayor a 90.
17. Escribe un programa que determine si un número es un número primo o no, utilizando un bucle for.
18. Escribe un programa que solicite al usuario ingresar un número entero positivo. Luego, utiliza un bucle while para calcular y mostrar la suma de los cuadrados de los dígitos de ese número. Por ejemplo, si el usuario ingresa 123, el programa debería mostrar: $1^2 + 2^2 + 3^2 = 14$.
19. Escribe un programa que diga si un número introducido por teclado es o no capicúa. Los números capicúa se leen igual hacia delante y hacia atrás. El

programa debe aceptar números de cualquier longitud siempre que lo permita el tipo, en caso contrario el ejercicio no se dará por bueno.

20. Realiza un conversor del sistema decimal al sistema de “palotes”.

Ejemplo:

Por favor, introduzca un número entero positivo: 47021

El 47021 en decimal es el | | | | - | | | | | | - - | | - | en el sistema de palotes

21. Escribe un programa que cambie un dígito dentro de un número dando la posición y el valor nuevo. Las posiciones se cuentan de izquierda a derecha empezando por el 1. Suponemos que el usuario introduce correctamente los datos.

Ejemplo:

Por favor, introduzca un número entero positivo: 406783

Introduzca la posición dentro del número: 3

Introduzca el nuevo dígito: 1

El número resultante es 401783

22. Realiza un programa que diga los dígitos que aparecen y los que no aparecen en un número entero introducido por teclado. El orden es el que se muestra en los ejemplos. Utiliza el tipo long para que el usuario pueda introducir números largos.

Ejemplo:

Introduzca un número entero: 67706

Dígitos que aparecen en el número: 0 6 7

Dígitos que no aparecen: 1 2 3 4 5 8 9

23. Realiza un programa que calcule el máximo, el mínimo y la media de una serie de números enteros positivos introducidos por teclado. El programa terminará cuando el usuario introduzca un número primo. Este último número no se tendrá en cuenta en los cálculos. El programa debe indicar también cuántos números ha introducido el usuario (sin contar el primo que sirve para salir).
24. Realiza un programa que sea capaz de desplazar todos los dígitos de un número de izquierda a derecha una posición. El dígito de más a la derecha, pasaría a dar la vuelta y se colocaría a la izquierda. Si el número tiene un solo dígito, se queda igual.
25. Realiza un programa que calcule la media de los dígitos que contiene un número entero introducido por teclado.
26. Según cierta cultura oriental, los números de la suerte son el 3, el 7, el 8 y el 9. Los números de la mala suerte son el resto: el 0, el 1, el 2, el 4, el 5 y el 6. Un número es afortunado si contiene más números de la suerte que de la mala suerte. Realiza un programa que diga si un número introducido por el usuario es afortunado o no.
27. Realiza un programa que muestre al azar el nombre de una carta de la baraja francesa. Esta baraja está dividida en cuatro palos: picas, corazones,

diamantes y tréboles. Cada palo está formado por 13 cartas, de las cuales 9 cartas son numerales y 4 literales: 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K y A (que sería el 1). Para convertir un número en una cadena de caracteres podemos usar `String.valueOf(n)`.

28. Escribe un programa que simule la tirada de dos dados. El programa deberá continuar tirando los dados una y otra vez hasta que en alguna tirada los dos dados tengan el mismo valor.
29. Realiza un simulador de máquina tragaperras simplificada que cumpla los siguientes requisitos:
- a) El ordenador mostrará una tirada que consiste en mostrar 3 figuras. Hay 5 figuras posibles: corazón, diamante, herradura, campana y limón.
 - b) Si las tres figuras son diferentes se debe mostrar el mensaje “Lo siento, ha perdido”.
 - c) Si hay dos figuras iguales y una diferente se debe mostrar el mensaje “Bien, ha recuperado su moneda”.
 - d) Si las tres figuras son iguales se debe mostrar “Enhorabuena, ha ganado 10 monedas”.