

Ejercicios

Bucles

1. Función que recibe un número N como parámetro y escribe N asteriscos por pantalla
2. Realiza un programa que pida por pantalla continuamente que introduzcas un número. Te devuelva por pantalla si el número introducido es par o no y que acabe cuando se introduzca un número menor que 0
3. Realizar un juego para adivinar un número. Primero se pide el número a adivinar y después se piden números continuamente, indicando para cada número introducido si el deseado es mayor, menor o si es el número buscado, en cuyo caso acabará el programa
4. Realizar una función que reciba un parámetro N y muestre por pantalla todos los primos anteriores
5. Realizar una función que reciba 2 parámetros y muestre por pantalla los primos entre ambos
6. Realizar una función que dibuje un triángulo de asteriscos de N (parámetro de la función) niveles de altura
7. Realizar una función que dibuje un cuadrado con guiones (-) de tamaño N (parámetro)
8. Realizar una función que calcule el factorial de un parámetro de entrada
9. Realizar una función que pida N números por teclado, siendo N un parámetro de la función, y devuelva la suma de esos números
10. Realizar una función que pida N números por teclado, siendo N un parámetro de la función, y escriba por pantalla la medida de los números positivos, la media de los negativos y la cantidad de ceros.
11. Realizar una función que escriba por pantalla la tabla de multiplicar del número recibido como parámetro
12. Realizar una función que pida N números por teclado, siendo N un parámetro de la función, y devuelva cierto si se ha introducido un número negativo.

Arrays y bucles

1. `public static String aTexto(int[] array)`
Crear un String con el contenido contenido del array, devolviendo por ejemplo para el array {3,9,0,-2} la cadena "3, 9, 0, -2"
2. `public static boolean esta(int elem, int[] array)`
Devuelve verdadero si el elemento 'elem' de tipo entero, está en el array, y falso e.o.c.
3. `public static int sumaPosPares(int[] array)`
Calcular la suma de los elementos de las posiciones pares del array y devuelve dicha suma
4. `public static int sumaPares(int[] array)`
Calcular la suma de los elementos pares (contenido) del array y devuelve dicha suma
5. `public static int cuentaImpares(int[] array)`
Devuelve el número de elementos impares contenidos en el array
6. `public static int contar(int elem, int[] array)`
Contar las ocurrencias del elemento 'elem' de tipo entero dentro en el array
7. `public static boolean estaOrdenado(int[] array)`
Comprobar si los elementos del array están ordenados de menor a mayor
8. `public static int maximo(int[] array)`
Devolver el elemento máximo almacenado en el array
9. `public static int[] invertir(int[] array)`
Devolver un nuevo array con los elementos del original en orden inverso
10. `public static int[] suma (int[] array1, int[] array2)`
Devolver un array de enteros (cuyo tamaño será el mayor de los tamaños de array1 y array2) que es la suma de los elementos de array1 y array2
11. `public static int[] crearArray(int elem)`
Dado un número entero 'capacidad'. Declarar un array de enteros de dicha longitud y rellenarlo con los 'capacidad' números impares desde 'capacidad'.
Ejemplo, si capacidad es 6, el array resultado será: {7,9,11,13,15,17}
12. `public int[] pedirArray()`
Pedir un número entero 'capacidad' al usuario. Según el número introducido, crear un array de dicho tamaño y pedir por consola al usuario los elementos necesarios para rellenar el array.
13. `public int[] pedirArray()`
Pedir un número entero 'capacidad' al usuario. Según el número introducido, crear un array de dicho tamaño y pedir por consola al usuario los elementos necesarios para rellenar el array.
14. Realizar una función que muestre por pantalla la tabla de multiplicar de los números del 1 al 10.
15. Función que reciba dos listas de enteros y devuelva una lista con los enteros de ambas listas ordenados
16. Función que recibe una lista de enteros y la desplaza n posiciones.
17. Función que recibe un array de enteros y una posición y devuelve un array con el elemento de esa posición eliminado y sin dejar huecos (es decir que se decrementa la longitud del array)
18. Función que recibe un array de enteros y devuelve un array ordeando de menor a mayor donde primero se sitúan todos los números pares y después todos los impares.

19. Función que recibe un array de enteros y devuelve una matriz con dos filas, donde la primera fila son los pares y la segunda fila los impares
20. Idem pero con columnas
21. Función que recibe una matriz de 5x5 y decir si es simétrica o no
22. Función que recibe 2 matrices y devuelve una con la suma de éstas. Si las matrices recibidas no son cuadradas y del mismo tamaño debe mostrar un error.
23. Función que recibe una matriz cuadrada (error si no lo es) y devuelva la matriz traspuesta.
24. Función que recibe un parámetro de tipo entero N y devuelve una matriz NxN con todos los elementos a 0 excepto la diagonal a 1.
25. Función que recibe una matriz cuadrada (error si no lo es) y devuelve un array con la suma de sus filas
26. Lo mismo pero con columnas

Strings

1. Realizar dos funciones, una que realice el cifrado de una cadena de texto según una codificación de diccionario lineal (Julio César) y la segunda el descifrado
2. Crear un array con direcciones de email. Realizar una función que devuelva en número de esas direcciones de correo electrónico tienen como dominio 'gmail.com'
3. Realizar una función que recibe un string y devuelve un entero con la cantidad de vocales que tiene
4. Función que recibe un string y devuelve el string invertido
5. Función que recibe un string y devuelve un string con el mismo valor y su espejo (Ej: input-Hola , Resultado- HoolaolH)
6. Función que recibe una cadena de texto con nombres de alumnos separados por comas y devuelve un array donde cada posición es un nombre
7. Función que recibe una frase en string y devuelva un entero con la cantidad de palabras que había en el parámetro de entrada.
8. Función que recibe una frase en string y devuelve un string invertido palabra a palabra