

## Programación modular I:

1. Crea un programa que genere 100 números de forma aleatoria y que posteriormente ofrezca al usuario la posibilidad de:
  - a. Conocer el mayor
  - b. Conocer el menor
  - c. Obtener la suma de todos los números
  - d. Obtener la media
  - e. Sustituir el valor de un elemento por otro número introducido por teclado
  - f. Mostrar todos los números

⇒ Realiza cada una de las opciones con funciones.

⇒ Utiliza la función siguiente para generar números aleatorios (entre 0 y 1000).

```
from random import randint  
  
numero = randint(0,1000)
```

2. Realiza un programa que lea 10 números, los imprima separados por coma y a continuación los desplace una posición (y los muestre por pantalla desplazados), de tal forma que el último pase a la primera posición, el primero a la segunda, el segundo a la tercera, y así sucesivamente.

Opcional: Añade un parámetro (D/I) a la función para que el controle el sentido del desplazamiento (a derechas/izquierdas) y otro que indique el número de posiciones a desplazar (0: quedaría igual, 1: desplaza una posición, etc.).

3. Realiza un programa que solicite la fecha como tres datos numéricos (día, mes y año) y muestre a continuación la fecha en formato largo.

Introduce el día de la fecha: 15

Introduce el mes de a fecha: 3

Introduce el año de a fecha: 2009

La fecha en formato largo es 15 de Marzo de 2009

Debe validar los datos y ejecutarse hasta que se introduzca un día negativo.

4. Crea un programa que lea por teclado números de forma sucesiva y los guarde en una lista; el proceso de lectura y guardado finalizará cuando metamos un número negativo. En ese momento se mostrará el elemento mayor y los números pares.

5. Realiza una función **reverse** que reciba una lista y devuelva una nueva lista cuyo contenido sea igual a la original pero invertida. Así, dada la lista ['Di', 'buen', 'día', 'a', 'papa'], deberá devolver ['papa', 'a', 'día', 'buen', 'Di'].

6. Diseña una función llamada **estaOrdenada** que reciba una lista de elementos y devuelva True si está ordenada o False en caso contrario.

7. Escribir una función denominada **encajan** que indique si dos fichas de dominó encajan o no. Las fichas son recibidas en dos cadenas de texto con el siguiente formato

[3,4] [2,5]

8. Realiza un programa que añada números enteros a una lista hasta que se introduzca un número negativo. Haciendo uso de esta lista, elabora funciones que devuelvan:

- una lista con todos los que sean primos.
- el sumatorio
- el promedio de los valores.
- una lista con el factorial de cada uno de esos números.

9. Desarrolla un programa que a partir de una lista de números y un entero k, realice la llamada a tres funciones: a) para devolver una lista de números con los menores de k, b) otra con los mayores y c) otra con aquellos que son múltiplos de k.

10. Diseña una función **convertor** que convierta un número de binario a decimal o de decimal a binario. Esta función recibirá un número en formato de cadena de texto cuya última posición indica el sistema numérico utilizado (D-decimal, B-binario).

Debe validar la información, así, por ejemplo, el número '1020101B' no sería válido puesto que los valores en binario son 0 y 1.

11. Escribe una función **intersect** que reciba dos listas y devuelva otra lista con los elementos que son comunes a ambas, sin repetir ninguno.

12. Escribe una función **unionListas** que reciba dos listas y devuelva los elementos que pertenecen a una, o bien, a la otra, pero sin repetir ninguno (unión de conjuntos).
13. Escribe una función que, dada una lista de nombres y una letra, devuelva una lista con todos los nombres que empiezan por dicha letra. Debe validar los datos.
14. Escribe una función que, dada una lista de cadenas, devuelva la cadena más larga. Si dos o más cadenas miden lo mismo y son las más largas, la función devolverá la que tenga el mayor número de caracteres repetidos .