

## EXAMEN TEMA 2 – PYTHON

### **NORMAS DEL EXAMEN – LEER ATENTAMENTE ANTES DE EMPEZAR!**

- Las excepciones deberán ser controladas
- No podrán usarse ni importarse funciones/módulos auxiliares a no ser que esté expresamente permitido en el ejercicio.
- Podrán usarse los apuntes, ejercicios o documentación de la que se disponga, pero NO PODRÁ HACERSE USO DE INTERNET DURANTE EL EXAMEN.

**EJERCICIO 1) (2,5 pto)** Define la función mezcla de forma que tome dos listas como parámetros y devuelva otra que es el resultado de mezclar los números de ambos de forma alterna, se coge un número de a, luego de b, luego de a, etc.. Los arrays a y b pueden tener longitudes diferentes; por tanto, si se terminan los números de un array, se terminan de coger todos los que quedan del otro.

**Se valorará que se use una función lambda.**

Ejemplos:

- Si a = [8,9,0] y b = [1,2,3], mezcla(a,b) devuelve [8,1,9,2,0,3]
- Si a = [4,3] y b = [7,8,9,10], mezcla(a,b) devuelve [4,7,3,8,9,10]
- Si a = [8,9,0,3] y b = [1], mezcla(a,b) devuelve [8,1,9,0,3]
- Si a = [] y b = [1,2,3], mezcla(a,b) devuelve [1,2,3]

**EJERCICIO 2) (2,5 ptos)** Define la función ordenaLista que ordene una lista siguiendo el método siguiente:

1. Si la lista es pequeña (vacía o de tamaño 1) ya está ordenada y no hay que hacer nada. Si no es así habrá que hacer los siguientes pasos de forma **RECURSIVA**:
2. Elegir el mayor elemento de la lista y ponerlo al final
3. Ordenar el resto de la lista

Recordad que en una función recursiva debe haber un “punto de escape”

**Nota:** Se puede importar la librería **numpy** para usar la función **max()** que te devuelve el valor máximo de una lista

**EJERCICIO 3) (2 ptos)** Se desea implementar la clase **Punto**, con las siguientes características:

- Sus atributos serán **x** e **y**.
- El **constructor**, que puede estar vacío, en cuyo caso el valor del punto sería (0,0).
- Hay que definir los **getters** y **setters** (usando **decoradores** de Python).
- Hay que definir una función **mostrar** (o también es válido **\_\_str()**) que muestre el punto por pantalla con el siguiente formato **(x,y)**
- Hay que definir un método **invierteCoordenadas** que invierta el orden de las coordenadas, es decir que **(x,y)** se convierta en **(y,x)**

**EJERCICIO 4) (3 ptos)** Se desea implementar un programa para gestionar la lista de la compra, que tenga un menú con 4 opciones:

- 1. Crear/añadir artículo** → Con esta opción pediremos al usuario por pantalla los elementos que quiere añadir a la lista de la compra, 1 o más, hasta que ya no quiera introducir más. Todos estos artículos los iremos añadiendo a un fichero que se llamará listaCompra.txt que tendrá un artículo en cada línea. Con esta primera opción habrá que abrir el fichero (se presupone que el fichero existe), y escribir en él los artículos que ha introducido el usuario, o añadirlos si ya hubiera artículos en él.
- 2. Borrar artículo** → Con esta opción se le pedirá al usuario que introduzca el o los artículos que desea borrar de la lista. Una vez que el usuario ha introducido todos los artículos a borrar, se procederá a borrarlos del fichero listaCompra.txt. Si algún artículo no estaba en la lista habrá que mostrar al usuario por pantalla el mensaje “El artículo {artículo} no se encontraba en la lista, por tanto no ha sido borrado” por cada artículo que no estaba en la lista.
- 3. Salir** → Esta opción solo servirá para salir del programa.