OBSERVACIONES DEL LA PRACTICA

Andrés Serrano Conti Cod 201731766

Cesar David Quiros Sierra Cod 201711029

Andrés Felipe Melgarejo Sarmiento Cod 202012655

1)     ¿Cuáles son los mecanismos de interacción (I/O: Input/Output) que tiene el **view.py**con el usuario?

En primer lugar, el programa muestra al usuario las diferentes opciones que tiene para seleccionar. En ese momento, aparece el primer input, en el que el usuario ingresa un número del 0 al 4 en la consola, acorde con las opciones que había mostrado el programa, utilizando una función input habitual en Python.  Con cada uno de esos diferentes inputs, se llaman funciones del controlador, que retornan diferentes elementos dependiendo de la opción escogida por el usuario y se imprimen en la consola.

2)     ¿Cómo se almacenan los datos de **GoodReads** en el **model.py**?

Como diccionario

3)     ¿Cuáles son las funciones que comunican el el **view.py**y el**model.py**?

Son las funciones a partir de la función “def loadData(catalog):” incluyendo a las funciones del menú, ya que estas hacen un uso de “catalog” que es un diccionario con la información de los libros  que creamos en la parte de “model.py “

4)     ¿Cómo se crea una lista?

Una lista se crea mediante la función newList de la librería DISClib.DataStructures.liststructure, que recibe como parámetros datastructure, cmpfunction, key, filename y delimeter.

5)     ¿Qué hace el parámetro **cmpfunction=None**en la función **newList()**?

Este parámetro se comprueba en la función newList de la clase arraylist. Con él, se determina si se utilizará una función de comparación por defecto (en caso de ser none), o no. Sin embargo, si se desea utilizar la función por defecto, el parámetro key no puede ser none. Por el contrario, key debe ser none.

6)     ¿Qué hace la funció **addLast()**?

Agrega un elemento al final de cada lista, además, se actualiza el apuntador a la última posición.

7)     ¿Qué hace la función **getElement()**?

Esta función recibe dos parámetros: (lst, pos) “lst” es la lista sobre la que estamos trabajando y “pos” es un int. La función recorre la lista y retorna el elemento que está en la posición “pos”

8)     ¿Qué hace la función **subList()**?

La función sublist se encarga de encontrar una sublista a partir de los 3 parámetros siguientes. En primer lugar, lst que es la lista a la que se quiere, pos, que es la posición desde la que se quiere obtener la sublista y numelem, que es el número de elementos a partir de pos que se desea incluir en la sublista.

9)     ¿Observó algún cambio en el comportamiento del programa al cambiar la implementación del parámetro **“ARRAY\_LIST”** a **“SINGLE\_LINKED”**?

No se presenta ninguna diferencia aparente en el funcionamiento del programa a la hora de cambiar ARRAY\_LIST por SINGLE\_LINKED. Sin embargo, es probable que de esta manera el programa funcione de manera más eficiente y se encuentren los resultados más rápidamente.