OBSERVACIONES DEL LA PRACTICA

Diego Alejandro Ramírez Garrido 202013233

Sebastián Murcia Gómez 202015229

1. ¿Cuáles son los mecanismos de interacción (I/O: Input/Output) que tiene el **view.py** con el usuario?
   1. (Output) printMenu() imprime el menú de funciones a través de la consola
   2. (Input) input(‘Seleccione…’) es un switch que toma el índice de la función escogida a través de la consola.
   3. (Output) En la opción **uno** después de cargar la información se imprime en la consola:
      1. Número de libros
      2. Número de autores
      3. Número de géneros
      4. Número de asociaciones entre libros y géneros
   4. (Input) En la opción **dos** se recibe el número del top de libros a consultar.
   5. (Output) se imprime el top x de mejores libros por promedio de calificación.
   6. (Input) En la opción **tres** se recibe el nombre del autor por medio de la consola.
   7. (Ouput) Se imprimen los libros del autor.
   8. (Input) En la opción **cuatro** se recibe la etiqueta a buscar.
   9. (Output) Se imprime el número de libros con la etiqueta.
2. ¿Cómo se almacenan los datos de **GoodReads** en el **model.py**?

Crea un diccionario de una LinkedList y 3 ArrayLists vacías llamado catálogo. Luego guarda los elementos a través de 4 diferentes funciones para las 4 diferentes listas.

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

* controller.initCatalog()
* controller.loadData()
* controller.getBestBooks()
* controller.getBooksByAuthor()
* controller.countBooksByTag()

1. ¿Cómo se crea una lista?

lt.newList(‘TIPO\_LISTA’,cmpfunction=función\_de\_ordenamiento)

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

Asigna el método de comparación por default (es decir, el orden clásico de mayor y menor)

1. ¿Qué hace la función **addLast()**?

Añade un elemento al final de la lista.

1. ¿Qué hace la función **getElement()**?

Obtiene el elemento en el índice dado en la lista dada.

1. ¿Qué hace la función **subList()**?

Obtiene una sublista de la lista a partir de la posición de inicio de la sublista y el número de elementos que debe tener esta.

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

* El principal cambio que surgió a partir de la modificación de la estructura fue el tiempo que tarda con la función “Consultar los Top x libros por promedio” con x=10 000, es decir con el número de elementos de los archivos a leer. Usando listas encadenadas, el programa respondió en aproximadamente tres segundos menos que con un arreglo. Para la cantidad de elementos con los que estamos trabajando no resulta significativo, pero a medida que se aumenten los datos será más evidente la diferencia.