OBSERVACIONES DEL LA PRACTICA

Helena Vegalara Correa 201823328

Juan Esteban Rodriguez 202011178

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

Los mecanismos de interacción entre el **view.py** y el usuario son a través de inputs que genera el **view.py** y que el usuario llena con la información que desea obtener. Los outputs serían entonces los resultados que obtiene el **view.py** (a través de llamar al y **controller.py** al **model.py**), y que le muestra al usuario.

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

En primer lugar, en el **model.py** se crea un catálogo con la utilización de un diccionario. Luego, para cada llave del diccionario, se crea una lista que luego se le va a agregar la información específica para cada dato (ej. Books, authors, tags, book\_tags). En este caso, las listas se crean como arreglos.

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

Las funciones que comunican el **view.py** y el **model.py** son **controller.initCatalog()** y también **controller.loadData(catalog)**, las cuales interactúan directamente con el **controller.py**, y a su vez, este interactúa con el **model.py** a través de las funciones **model.newCatalog()** y demás funciones como **model.addBook(catalog, book)**, **model.addTag(catalog, tag)**, **model.addBookTag(catalog, booktag)**, **model.sortBooks(catalog)**, **model.GetBooksByAuthor(catalog, authorname)**, entre otras funciones. Es decir, la vista y el modelo interactúan a través del controlador, llamando funciones de cada una de los módulos.

1. ¿Cómo se crea una lista?
2. ¿Qué hace el parámetro **cmpfunction=None** en la función **newList()**?
3. ¿Qué hace la funció **addLast()**?
4. ¿Qué hace la función **getElement()**?
5. ¿Qué hace la función **subList()**?
6. ¿Observó algún cambio en el comportamiento del programa al cambiar la implementación del parámetro **“ARRAY\_LIST”** a **“SINGLE\_LINKED”**?