**Documento de análisis**

**Reto 2-EDA-2021-10-Grupo 12**

Estudiante A: Julian Castro del Valle - [j.castrod@uniandes.edu.co](mailto:j.castrod@uniandes.edu.co) - 202020847

Estudiante B: Tomás Otero - [t.otero@uniandes.edu.co](mailto:t.otero@uniandes.edu.co) – 202021733

**Análisis de complejidad por requerimiento:**

* Requerimiento 1: O(N) ya que cuenta con un ciclo for para recorrer la lista de los valores del catálogo, el cual es un mapa ordenado/árbol.
* Requerimiento 2 (Estudiante A): O(N) ya que cuenta con un ciclo for para recorrer la lista de los valores del catálogo, el cual es un mapa ordenado/árbol.
* Requerimiento 3 (Estudiante B): O(N) ya que cuenta con un ciclo for para recorrer la lista de los valores del catálogo, el cual es un mapa ordenado/árbol.
* Requerimiento 4: O(2N) ya que cuenta con dos ciclos for, uno para recorrer la lista de los géneros dados por parámetro para corroborar que existan, y otro para recorrer la nueva lista.
* Requerimiento 5: O(5N2) ya que cuenta con 6 ciclos for, para recorrer la lista de los géneros, datos de los usuarios y canciones del mapa, así como la lista de los valores ordenados. Puede verse como N2.

Se utiliza el algoritmo merge sort que tiene complejidad O(N log N) en promedio, siendo el algoritmo más rápido de los vistos en clase.

Se utiliza la estructura de datos tipo arreglo (ARRAY\_LIST), al ser la más óptima en este caso en cuanto a velocidad y eficacia para almacenar datos. Adicionalmente, se usa el mecanismo de colisión Linear Probing, ya que también es el más eficiente entre los 2 mecanismos estudiados en clase, y se usan Árboles Rojo-Negro y Mapas ordenados, que son otras estructuras y TADs altamente eficientes.

**Observaciones importantes:**

* Las funciones de impresión de resultados son O(2N) y O(4N), sin embargo al ser auxiliares para que se muestre la información requerida en el view, no influyen en cómo se organizan/filtran los datos.

**Análisis de tiempo de ejecución y memoria:**

|  |  |  |
| --- | --- | --- |
| **Requerimiento** | **Consumo de Datos [kB]** | **Tiempo de Ejecución [ms]** |
| 1.00 | 14.945 | 5334.618 |
| 2.00 | 37.062 | 10916.354 |
| 3.00 | 32.344 | 11416.5 |

Gracias a los análisis de tiempo y memoria realizados, se puede ver que este ha sido el reto más eficiente de todos los desarrollados hasta el momento. Los requerimientos analizados consumen una cantidad bajísima de memoria y toman tiempos de ejecución muy cortos, y eso que dichos datos fueron tomados en un portátil con apenas 8 GB de RAM y un procesador de 1.8 GHz. Esto nos puede llevar a concluir que, para máquinas lentas, la combinación entre los mapas ordenados, árboles, arreglos y mapas con Linear Probing es óptima en términos de economización de tiempo y de memoria consumida.