Observaciones del Reto 3

Nicolás Guerrero Cod 201731839 (Estudiante A)

Santiago Duque Cod 202021020 (Estudiante B)

Análisis de complejidad:

Requerimiento 1: On^2

La complejidad es de On^2 pues se diversos ciclos en el requerimiento, pero los de mayor complejidad son los ciclos anidados que en total tendrían complejidad de On^2.

Requerimiento 2: On^2

En este requerimiento se usa la función de los RBTs om.values además de un for in para recorrer dicha lista. Adicionalmente se utiliza un ciclo anidado para poder recorrer dos listas y combinarlas, este procedimiento tiene complejidad de On^2 por lo que esta seria la complejidad final del requerimiento.

Requerimiento 3: On^2

Al igual que en el requerimiento 2, las funciones utilizadas usan ciclos anidados para combinar dos listas, como hay dos ciclos uno dentro del otro, la complejidad es de On^2.

Requerimiento 4: On^2

El requerimiento 4 utiliza funciones y ciclos parecidos a los previos requerimientos, en donde se lleva a cabo un ciclo anidado de dos for ins para poder recorrer una lista y compararla con otra, de ahí su complejidad.

Requerimiento 5: On^2

El requerimiento cuenta con ciclos for, consultas sobre listas, mapas y árboles, además de un merge sort que se realiza, pero la mayor complejidad se encuentra en los for ins anidados que, en el peor de los casos, se recorren todos los datos dos veces, por lo que la complejidad de este requerimiento es de On^2.

Estudiante A

|  |  |  |
| --- | --- | --- |
| **Requerimiento** | **Consumo de Datos [kB]** | **Tiempo de Ejecución [ms]** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |

Estudiante B

|  |  |  |
| --- | --- | --- |
| **Requerimiento** | **Consumo de Datos [kB]** | **Tiempo de Ejecución [ms]** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |