Observaciones del Reto 4

Nicolás Guerrero Cod 201731839

Santiago Duque Cod 202021020

Análisis de complejidad:

Requerimiento 1: On^2

La complejidad del requerimiento es de On^2 pues el requerimiento cuenta con dos ciclos for in anidados para recorrer listas con los landing points.

Requerimiento 2: On^2

Este requerimiento cuenta con dos ciclos anidados, operación que tiene complejidad de On^2. Además de estos ciclos las demás operaciones son consultas sobre listas y mapas por lo que su complejidad es menor a la de los ciclos anidados.

Requerimiento 3: E log V

Este requerimiento no cuenta con ciclos anidados, no obstante, el requerimiento hace uso del algoritmo de Dijkstra el cual tiene una complejidad de E log V en el peor de los casos.

Requerimiento 4: On^2

Este requerimiento, además de usar un ciclo solo usa el algoritmo prim por lo que su complejidad, en el peor de los casos es de On^2 siendo n el numero de vértices del grafo.

Requerimiento 5: On log n

Este algoritmo utiliza varios ciclos for para hacer operaciones sobre listas pero la mayor complejidad viene del algoritmo merge sort que utiliza sobre una lista, de ahí la complejidad On log n.

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 |  |  |