

Grupo 9

Santiago Prieto Álvarez 202013173 s.prietoa@uniandes.edu.co

Brian Manuel Rivera Hernández 202015320 b.riverah@uniandes.edu.co

Requerimiento A

TAD: Map (Probing) y List (Array List)

Por qué: El mapa da la posibilidad de ir guardando los datos asociados a la cantidad de taxis y también dice la cantidad de servicios por lo tanto todo esto al final se encontrará condensado en un único mapa que contiene las compañías, por lo tanto si establezco 2 ordenamientos uno orientado por la cantidad de taxis y el otro por la cantidad de servicios es muy fácil tener el top que pide el usuario al final, se intentó usar una sola lista y que se fueran actualizando mientras llegaban y así dejar el top organizado para la consulta pero la complejidad subía al cuadrado a medida que los datos subían por lo tanto tener un mapa y posteriormente sacar los valores de ese mapa y ordenarlos bajo 2 criterios distintos es mucho más eficiente y casi instantáneo a la hora de realizar la búsqueda y a la hora de cargar los datos

Complejidad: $O(N)$

Requerimiento B (Bono)

TAD: Mapa ordenado(BST), Mapa(Probing) y List (Array List)

Por qué: Para poder consultar un rango de fechas es esencial un mapa ordenado además también se puede usar como un mapa normal, por lo tanto si se usa BST los datos se guardarán más rápido en la carga de datos y no afectará mucho en la consulta debido a que este tiende a realizar pocas búsquedas dentro del mapa ordenado, el mapa que se usa dentro del mapa ordenado permite llevar una cuenta eficiente de los servicios y puntos que lleva un taxi en una fecha en específico para posteriormente usarse en los puntos de este requerimiento, también tiene una función muy útil como la función values que me permite obtener todos los taxis de ese día y poder ordenarlos en una lista de tipo ARRAY LIST y pasarles un algoritmo de ordenamiento con una función de comparación específica, logrando así reducir la complejidad hasta lo menor posible.

Complejidad: $O(N)$

Requerimiento C:

TAD: Grafo (Lista de adyacencia), Map (Probing) y Lista (Array List)

Por qué: El grafo se escogió porque es la única estructura que permite hallar caminos de costo mínimo y otras funciones que las otras estructuras de datos no tienen. El mapa fue escogido para almacenar llaves que son las community areas, las cuales dentro tienen las

horas, que están dentro de una Array List, lo cual es útil para hallar rápido los horarios de una community área, y la lista de horas es útil para iterarlas rápidamente.

Complejidad: $O(N^2)$