

RETO INTEGRADOR

| Nombre | Correo | Código |
|-------------------------------|------------------------------|-----------|
| Juan Esteban Rodríguez Ospino | j.rodriguezo@uniandes.edu.co | 202011171 |
| Juan Sebastián Peña Muñoz | j.penam@uniandes.edu.co | 202013078 |
| German Alberto Rojas Cetina | g.rojasc@uniandes.edu.co | 202013415 |

REQUERIMIENTO A

Para este requerimiento hicimos uso de listas ordenadas y Tablas de Hash.

Elegimos esta estructura de datos debido a que en este requerimiento nos estaban preguntando por el total de un grupo de datos, y el top de estos. Por lo cual, supimos que esto se podría resolver con una lista ordenada. Las tablas de Hash nos permiten acceder de forma rápida y directa a un elemento por consultar.

La complejidad del algoritmo implementado para este requerimiento fue: **$O(1)$**

REQUERIMIENTO B

Para este requerimiento hicimos uso de árboles binarios, con Tablas de Hash en los nodos.

La elección de esta estructura de datos vino con la necesidad de organizar un grupo de fechas de tal modo que sea fácil recorrerlas y acceder a ellas, por lo cual implementamos un árbol binario. Para los datos de consulta lo más práctico fue ordenarlos en una Tabla de Hash en la cual el acceso a estos sería muy simple.

La complejidad del algoritmo implementado para este requerimiento fue: **$O(\log n)$**

REQUERIMIENTO C

Para este requerimiento hicimos uso de un grafo.

En este requerimiento, al tener que implementar una estructura que la cual se visualice las diferentes *Community Areas* y el desplazamiento entre ellas, llevando registro de sus tiempos, viene fácilmente a nuestras mentes el concepto de grafos. Los grafos nos permitieron estructurar y almacenar viajes entre vértices de forma que pudimos responder el mejor horario para un rango de horas, dado un origen y destino.

La complejidad del algoritmo implementado para este requerimiento fue: **$O(n \log n)$**