

Algoritmos y estructuras de datos.

Diseñe una estructura de datos que permita la recuperación eficiente de rutas en función de una parada determinada.

Visión de la solución:

Ruta R1 contiene las siguientes paradas: **A → B → C → D**

Las paradas deben ser ordenadas, para llegar de A a D se debe transitar por B y C previamente (Lista enlazada).

Por otro lado, una parada B puede estar presente en las rutas **R1, R3 y R7**

Solo importa conocer que rutas están asociadas a una parada sin importar orden pero... debe ser no repetitivo (Set).

pseudocode

```
structure RouteMap:  
    Map<route_id, LinkedList<stop_id>>  
    Map<stop_id, Set<route_id>>
```

Para abordar el problema de gestionar rutas de transporte público y permitir operaciones eficientes de recuperación, adición y eliminación de paradas, se propone una solución basada en la combinación de dos estructuras de datos principales. En primer lugar, se utiliza un mapa hash cuya clave es el identificador único de cada ruta (`route_id`) y cuyo valor es una lista enlazada simple que representa las paradas asociadas a dicha ruta. Esta elección, permite conservar el orden secuencial de las paradas, fundamental en el contexto de recorridos físicos, y facilita operaciones de inserción o eliminación en cualquier punto de la secuencia sin necesidad de reordenamientos costosos. En segundo lugar, se implementa un mapa inverso en el que la clave corresponde a una parada (`stop_id`) y el valor es un conjunto de identificadores de ruta en los que dicha parada aparece. Esta estructura permite recuperar de manera eficiente, en tiempo constante promedio, todas las rutas que contienen una parada específica, garantizando además la unicidad de los elementos sin

James Montealegre Gutiérrez, Ingeniero Telemático, Universidad ICESI, Cali – Colombia
MSc. Ingeniería de Software, Universidad de los Andes, Bogotá – Colombia, Senior Fullstack Engineer.

necesidad de verificaciones adicionales. Ambas estructuras trabajan de forma complementaria y se actualizan de manera sincronizada para mantener la integridad y coherencia de los datos en todas las operaciones del sistema.

Mapa de rutas: Map<route_id, LinkedList<stop_id>>

Este mapa almacena, para cada identificador de ruta (route_id) una lista enlazada de paradas asociadas. La elección de una lista enlazada simple como estructura interna es una representación intuitiva de orden secuencial de las paradas dentro de una ruta de transporte que responde a la necesidad de:

- Conservar el orden de las paradas dentro de la ruta
- Permitir inserciones en tiempo constante
- Eliminar elementos conocidos mediante recorrido lineal (en el peor de los casos $O(N)$).

Mapa de paradas: Map<stop_id, Set<route_id>>

Este “índice inverso” permite recuperar para una parada determinada, el conjunto de rutas en la que aparece. El uso de una estructura de tipo conjunto garantiza unicidad de los identificadores de ruta y facilita operaciones de inserción y eliminación en tiempo constante promedio también.

Este mapa se actualiza dinámicamente con cada operación de adición o eliminación de paradas, manteniendo la consistencia entre ambos mapas. Gracias a esta estructura, se logra una consulta eficiente de rutas por parada, sin necesidad de recorrer todas las rutas del sistema.