

Juan José Restrepo Bonilla
Josue Rodriguez.
Manuel Castaño.

Enunciado:

El sistema integrado de transporte masivo de trenes (SITMT) de la ciudad colombiana de Santiago de Cali, ha estado en quiebra durante los dos últimos años por la caída exponencial de usuarios que han decidido pasarse a otras plataformas de mejor calidad y menor tiempo de retraso dado las rutas que establecen, entonces la alcaldía municipal preocupada por la situación actual del sistema, ha buscado desesperadamente ingenieros de sistemas y telemáticos que solucionen sus problemas. sorprendentemente, recibieron una llamada de un profesor de la universidad Icesi donde les ofrecía al mejor grupo de ingeniería que cursan algoritmos y estructuras de datos para dar soluciones a sus problemáticas. Entonces el alcalde de Cali, se comunica con ustedes para pedirles que soluciones sus problemáticas, a través de un software, donde necesitan buscar la mejor ruta dada una estación en específica, saber cual es el última estación para poder decirle a sus usuarios hasta donde se puede llegar y además buscar una estación en específica para poder mostrarle su información que tiene hasta su momento. Entonces su grupo ni corto ni perezoso ha decidido emprender su viaje en este gran problema, donde todo depende si la alcaldía se va a la quiebra o buscan la luz de la esperanza.

Método de la ingeniería:

1. Especificación del problema:

Un sistema de transporte de trenes necesita implementar un software de transportes para la búsqueda de la mejor ruta de transporte.

2. Recopilación de la información:

Un grafo es la representación simbólica de los elementos constituidos de un sistema o conjunto, mediante esquemas gráficos. se puede decir también, que un grafo consiste en un conjunto de nodos (también llamados vértices) y un conjunto de arcos (aristas) que establecen relaciones entre nodos.

Aristas: Son las líneas con las que se unen las aristas de un grafo y con la que se construyen también caminos. Si la arista carece de dirección se denota indistintamente $\{a, b\}$ o $\{b, a\}$, siendo a y b los vértices que une.

Vértices: Son los puntos o nodos con los que está conformado un grafo. Llamaremos grado de un vértice al número de aristas de las que es extremo. Se dice que un vértice es 'par' o 'impar' según lo sea su grado.

Caminos: Se llama camino a una secuencia de vértices de un grafo tal que exista una arista, cada vértice y el siguiente.

Propiedades de los grafos:

Adyacencia:

Dos aristas son adyacentes si tienen un vértice en común, y dos vértices son adyacentes si una arista los une.

Incidencia

Una arista es incidente a un vértice si ésta lo une a otro.

Ponderación

Corresponde a una función que a cada arista le asocia un valor, para aumentar la expresividad del modelo.

Etiquetado.

3. Functional requirements:

Name	1. Find minimum path.
Summary	The program needs to find the minimum time to travel between two stations and the path of that travel.
Inputs	Two stations.
Outputs	The time and the minimum path.

Name	2. Last station
Summary	The program needs to find the last station of the route to inform to the user that he can't go further.
Inputs	None inputs.
Outputs	The last station.

Name	3. Find station
Summary	The program needs to find a station to show his information.
Inputs	Station
Outputs	The information of the station.

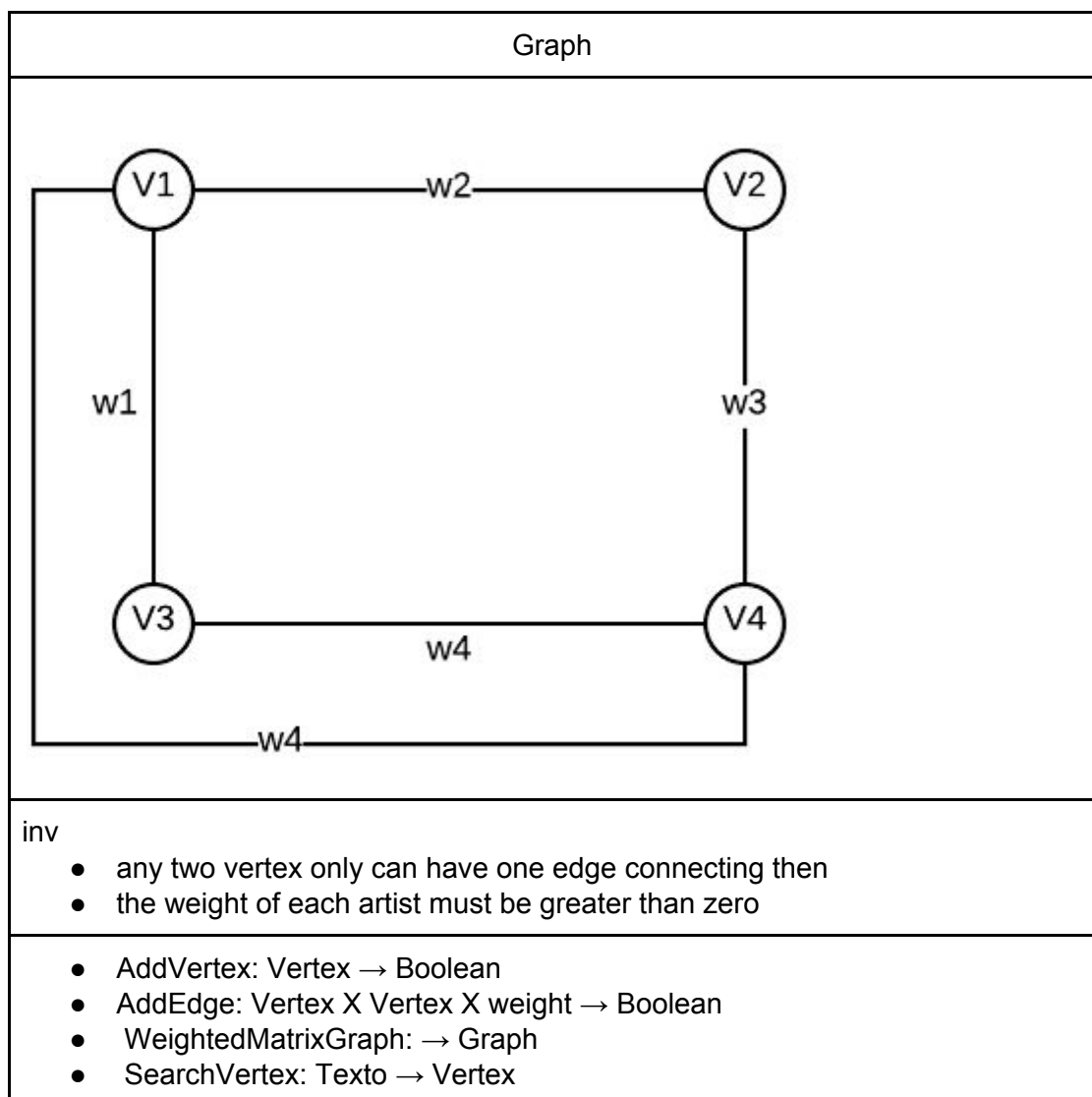
4. Creative solutions:

- Implement a simple graph where the vertex is the train station and the edges connects two stations and his weight is the time to travel between them.
- Implement a direct graph where a edge can connect two stations in only one direction.
- Implement a direct multigraph where two train stations can have more than one edge connecting them in one direction.

5. Selection of the solutions:

We are going to choose the simple graph to model the problem, because according to the Cali's major the path between two stations are made to function in both directions, so a direct graph is less efficient because we should connect two stations with two edges in both directions.

TAD



- DepthFirstSearch: Graph X Vertex \rightarrow List
- BreadthFirstSearch: Graph X Vertex \rightarrow List
- Djikstra: Graph X Vertex \rightarrow List
- FloydWarshall: Graph \rightarrow List
- RemoveEdge: Vertex X Vertex \rightarrow Boolean
- RemoveVertex Vertex \rightarrow Boolean