

# RETO 4: Red de Buses en Barcelona

---

## Objetivo

Poner en práctica los conceptos aprendidos en clase acerca de las estructuras de datos módulo No. 4 sobre del ADT Graph de tipo dirigido y no-dirigido.

Específicamente se desea:

1. **Utilizar** grafos como estructura de datos principal para resolver consultas y requerimientos.
2. **Integrar** el uso de las estructuras lineales (tablas de hash, Lista, pilas, colas y arboles) a los grafos dirigidos y no dirigido.
3. **Utilizar** adecuadamente el patrón MVC (Modelo-Vista-Controlador).
4. **Aprender** a cargar y procesar en memoria datos en formato CSV.
5. **Utilizar** adecuadamente el administrador de versiones GIT y GitHub.
6. **Aprender** a trabajar en equipo

## Fecha Límite de Entrega

Máxima fecha de entrega para el 6 de diciembre antes de la media noche (11:59 p.m. hora BrightSpace).

## Actualizaciones

A continuación, encontrará un listado de todas las actualizaciones realizadas sobre el enunciado para claridad de los requerimientos y objetivos del Reto.

- 1) Se **agregó** el ejemplo funcional de la limpieza y carga de datos con un grafo dirigido y no dirigido utilizando el archivo con sufijo -small.
- 2) Se **modificó** los subconjuntos de datos en la fuente oficial del reto dentro de Bloque Neón para evitar errores en las pruebas de tiempos de ejecución.

## Contexto

Este conjunto de datos representa una subred de transporte público en la ciudad de Barcelona. Contiene la información del servicio diurno de las rutas de bus.

Debido al crecimiento continuo de las ciudades en población, infraestructura y servicios, las organizaciones y agencias que intervienen en su administración y funcionamiento tienen como preocupación contar con información actualizada en los aspectos que afectan la vida de sus ciudadanos: educación, salud, transporte, vivienda, infraestructura, entretenimiento, turismo, seguridad, economía, entre otras. Esta información

permite mantener informados a sus ciudadanos en sus actividades comunes y a los administradores y autoridades locales tomar decisiones que mejoren la calidad de vida de sus ciudadanos<sup>1</sup>.

El transporte es una de las problemáticas importantes en una ciudad. En particular, el transporte público debe ofrecer soluciones eficientes para transportar un alto volumen de la población con buena calidad. Y así aumentar la calidad de vida de sus ciudadanos y las condiciones de vida de la ciudad. Donde ofreciendo un buen servicio de transporte público se incentiva a reducir el uso del transporte privado<sup>2</sup>.

## Carga de Datos

Los datos para este reto fueron tomados del repositorio “**Discovering Barcelona Part II**”<sup>3</sup>, que contiene la información de la red de buses públicos de la ciudad de Barcelona.

Los archivos originales son del año 2019, que contienen la información de las rutas del servicio diurno y nocturno de buses, la cual es necesaria para completar los requerimientos del reto. En el archivo se tiene información de cada ruta de bus y la secuencia de sus estaciones desde su estación inicial hasta su estación final. Se considera que cada ruta de bus de día funciona en ambas direcciones usando las mismas estaciones.

En la Tabla 1 puede verse el resumen de los campos de información de una estación de bus.

Nombre de la Columna	Descripción
Code	Identificador numérico de una estación. Valor único (diferente) si la estación es exclusiva para una ruta de bus. Valor común para estaciones compartidas por diferentes rutas de bus.
Longitude	Coordenada longitud de la estación de bus en un mapa
Latitude	Coordenada latitud de la estación de bus en un mapa
Bus.Stop	Identificador de la ruta de bus que hace uso de la estación. Tiene la forma BUS-IdRuta con el identificador de la ruta.
District.Name	Nombre del distrito (zona) en la ciudad donde se ubica esa estación.
Neighborhood.Name	Nombre del vecindario (“barrio”) en la ciudad donde se ubica esa estación.
Transbordo	Indica si la estación es de Transbordo ('S') o no ('N'). Las estaciones de transbordo son compartidas por 2 o más rutas de bus (ver ejemplo 1). Las estaciones que No son de transbordo son estaciones exclusivas por las que pasa una única ruta de bus (ver ejemplo 2).

Tabla 1. Descripción de los campos del archivo CSV de las rutas y estaciones de bus<sup>4</sup>.

**Ejemplo 1:** Las estaciones compartidas son registros diferentes en el archivo que comparten el Code, la longitud y la latitud. Pero cada estación compartida reporta una ruta de bus diferente. Estas estaciones además tienen valor de Transbordo 'S'. Por ejemplo, las estaciones compartidas por las rutas BUS-109, BUS-115, BUS-CJ, BUS-V7 y BUS-D40. Estas estaciones son estaciones de transbordo.

---

<sup>1</sup> TMB's strategy for an efficient bus network in Barcelona, by Dídac Pestaña, 2021, URL:

<https://www.intelligenttransport.com/transport-articles/4959/tmbs-strategy-for-an-efficient-bus-network-in-barcelona/>

<sup>2</sup> What it is and the completed bus network, TMB URL: <https://www.tmb.cat/en/about-tmb/transport-network-improvements/new-bus-network/evolution>

<sup>3</sup> Discovering Barcelona Part II, URL: [https://www.kaggle.com/code/xvivancos/discovering-barcelona-part-ii/data?select=bus\\_stops.csv](https://www.kaggle.com/code/xvivancos/discovering-barcelona-part-ii/data?select=bus_stops.csv)

<sup>4</sup> Discovering Barcelona Part II, Introduction, URL <https://www.kaggle.com/code/xvivancos/discovering-barcelona-part-ii/introduction>.

**Ejemplo 2:** Identifica estaciones exclusivas. Por ejemplo, la ruta 109 tiene múltiples estaciones exclusivas con Code diferente, identificador BUS-109 y valor de Transbordo 'N'.

**NOTA:** Como identificador único de una estación para el grafo, se sugiere concatenar el código y la ruta. Por ejemplo: 26-109, donde 26 es el código de la estación y 109 es el número de la ruta del BUS-109.

Adicionalmente se cuenta con un archivo que describe las conexiones entre estaciones de bus (arcos) y la ruta que las conecta. Estas conexiones son unidireccionales (de origen a destino). En la Tabla 2 puede verse el resumen de los campos de información más importantes de las conexiones entre estaciones de bus.

Nombre de la Columna	Descripción
Code	Identificador numérico de la estación origen.
Bus.Stop	Identificador de la ruta de bus (de la forma BUS-IdRuta) que conecta las estaciones.
Code_destino	Identificador numérico de la estación destino.

Tabla 2. Descripción de los campos del archivo CSV de las conexiones (arcos) entre estaciones de bus

## Trabajo Propuesto

### Parte 1: Configuración Repositorio

Complete los siguientes pasos para configurar su repositorio de trabajo:

1. Cree en GitHub un repositorio basado en la plantilla propuesta para el reto, el cual se encuentra en el URL: <https://github.com/ISIS1225DEVs/Reto4-Template>
2. Renombre el repositorio de su reto con el formato **Reto4-G<<Número del grupo>>** ej.: **Reto4-G01** para el grupo 1 de la sección 2.
3. Edite el **README** del repositorio e incluya los nombres completos, correo Uniandes y códigos de los miembros del equipo de trabajo.
4. Realice el procedimiento según lo aprendido en clase para clonar el repositorio en su máquina local y sincronizarlo con su repositorio en GitHub.
5. Descargue los datos desde la sección unificada del curso y cópielos en la carpeta **data** del repositorio local.

### Parte 2: Carga de Datos

En la sección unificada en la sección de reto 4 encontrarán los datos oficiales del proyecto. el ZIP contiene varios archivos con los sufijos **-small**, **-large**, **-5pct**, **-10pct**, **-20pct**, **-30pct**, **-50pct** y **-80pct**. Son archivos con diferente número de registros. Esto facilita la implementación y pruebas en computadores con memoria RAM y procesadores reducidos.

Para responder a los requerimientos presentados deberán cargar la información de los archivos entregados; recuerde que solo se permite leer una vez la información de cada archivo y que las pruebas finales sobre sus algoritmos serán sobre el archivo **\_large.csv**.

Los archivos contienen la información de rutas de bus y sus respectivas estaciones de Barcelona (**bus\_stops**). A partir de esta información hay que construir un grafo que representa la red de buses (ver ejemplo ilustrativo anexo al enunciado). El grafo se debe construir bajo las siguientes características:

- Una ruta de bus está descrita por la secuencia de sus estaciones.

- Las conexiones entre estaciones de bus son unidireccionales. Pero se garantiza que si existe la conexión A -> B entonces existe la conexión B->A.
- Una ruta de bus es bidireccional.
- Las estaciones de bus son los vértices del grafo. El identificador en el grafo para cada estación es la combinación de su identificador numérico (campo Code) con el identificador de la ruta (Bus.Stop). Este identificador en el grafo debe crearse con la forma Code-IdRuta.

Por ejemplo, si las dos primeras estaciones de la ruta 109 son exclusivas, su Code es un número que no se repetirá (por ejemplo 53 y 54), su Bus.Stop es BUS-109 y deben tener un valor Transbordo de 'N'. Estas estaciones serán dos vértices en el grafo con los identificadores únicos: 53-109 y 54-109.

- Las estaciones compartidas van a tener un tratamiento especial. Las estaciones compartidas se reconocen porque tienen 'S' en su valor Transbordo y valores iguales para el Code, la longitud y la latitud. Para cada estación compartida que aparezca en una ruta se debe generar un vértice con el identificador respectivo Code-IdRuta.

Por ejemplo, para las estaciones compartidas por las rutas BUS-100, BUS-109 y BUS-110 se tiene valor Transbordo de 'S' y valor Code 58. Los identificadores de los vértices respectivos en el grafo serán 58-100, 58-109 y 58-110.

- A partir del archivo de los arcos (bus\_edges), se encuentra la secuencia de los arcos del grafo; cada arco definido por una estación origen, una ruta que conecta las estaciones y una estación final. Por ejemplo, el registro "26, BUS-109, 45" quiere decir que existe una conexión desde la estación 26 hasta la estación 45 conectadas por la ruta 109. Esta conexión en el grafo conecta el vértice inicial 26-109 con el vértice final 45-109.
- A partir de la geolocalización de dos estaciones conectadas por un arco se puede calcular su distancia Harvesine con la **Fórmula del semiverseno (Haversine formula)**<sup>5,6,7,8</sup> (en kilómetros o metros). Esta distancia corresponderá al "peso" del arco que conecta dos estaciones consecutivas de una misma ruta.
- Luego de cargar la información de estaciones y conexiones dada en el grafo, se tendrá la conectividad de cada ruta con sus estaciones. Las rutas estarán independientes. Por lo tanto, como paso final se debe crear estaciones de transbordo donde se conectan las estaciones compartidas. Una estación de transbordo permite llegar a una estación por una ruta de bus y cambiar a otra ruta de bus. Entonces para conectar las estaciones compartidas se debe construir un nuevo vértice con identificador T-Code. Este nuevo vértice debe conectarse con arcos de ida y vuelta a los vértices de sus estaciones compartidas.

Una ilustración de un grafo resultante a partir de la información de las estaciones de las rutas de buses (simplificadas) para las estaciones 100, 109 y 110 se puede ver en Ilustración 1

---

<sup>5</sup> Haversine formula, URL: [https://en.wikipedia.org/wiki/Haversine\\_formula](https://en.wikipedia.org/wiki/Haversine_formula)

<sup>6</sup> Calculate distance, bearing and more between Latitude/Longitude points, URL: <https://www.movable-type.co.uk/scripts/latlong.html>

<sup>7</sup> Haversine Formula in Python (Bearing and Distance between two GPS points), URL: <https://stackoverflow.com/questions/4913349/haversine-formula-in-python-bearing-and-distance-between-two-gps-points>

<sup>8</sup> haversine 2.5.1, URL: <https://pypi.org/project/haversine/>

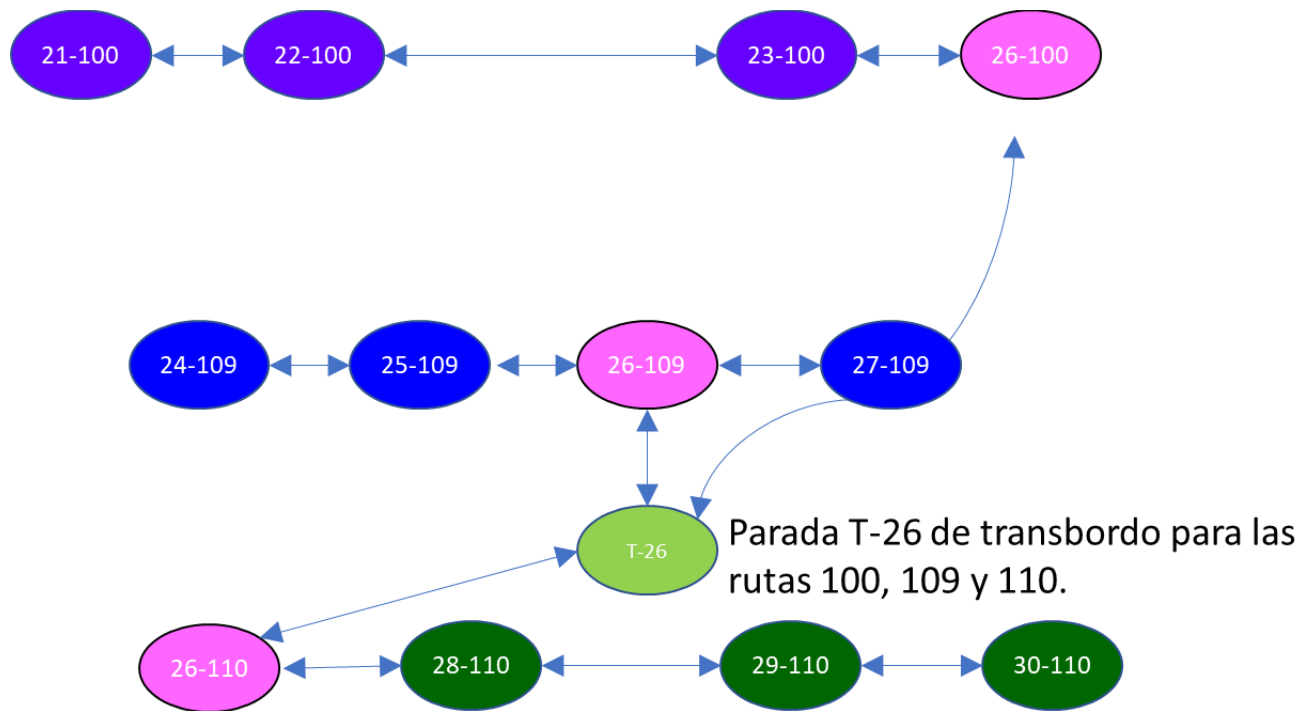


Ilustración 1. Esquema para solucionar estaciones de transbordo.

Al final de la carga hay que reportar los siguientes datos:

- El total de rutas de bus disponibles.
- El total de estaciones exclusivas de todas las rutas de bus.
- El total de estaciones de transbordo (estaciones compartidas).
- El total de arcos utilizados en todas las rutas de buses.
- Mostrar el rango del área rectangular de Barcelona que cubre la red de buses con sus valores [Longitud mínima, Longitud máxima] y [Latitud mínima, Latitud máxima]
- Mostrar las primeras cinco y últimas cinco estaciones registradas en el grafo con las siguientes características:
  - Identificador de la estación (Code-Ruta).
  - Geolocalización (Longitud, Latitud).
  - Número de estaciones de conexión (o número de estaciones adyacentes).

#### Recomendaciones:

- Recuerde que el grafo resultante posee un número diferente de nodos compartidos que depende de la cantidad de estaciones y rutas disponibles en el conjunto de datos.

**Nota:** Los ejemplos del documento están hechos con el subconjunto de datos más pequeño ("\_small.csv").

EJEMPLO: Carga de archivos -small.

```
===== CSV Loading Specs =====

CSV files loaded successfully...

Bus Stops CSV file size: 134
Routes CSV file size: 10

===== Bus-Stops-Routes Processing Stats =====

--- Bus-Stops ---
Exclusive Bus-Stops: 134
Shared Bus-Stops: 94
Total Bus-Stops: 228

--- Bus-Routes ---
Exclusive Bus-Routes: 10
Shared Bus-Routes: 198
Total Bus-Stops-Routes: 288

Creating Graph + DiGraph...

===== Bus-Stops-Routes DiGraph Specs =====
DiGraph Nodes: 228
DiGraph Edges: 208
First 5 & Last 5 Bus-Stops loaded in the DiGraph.
```

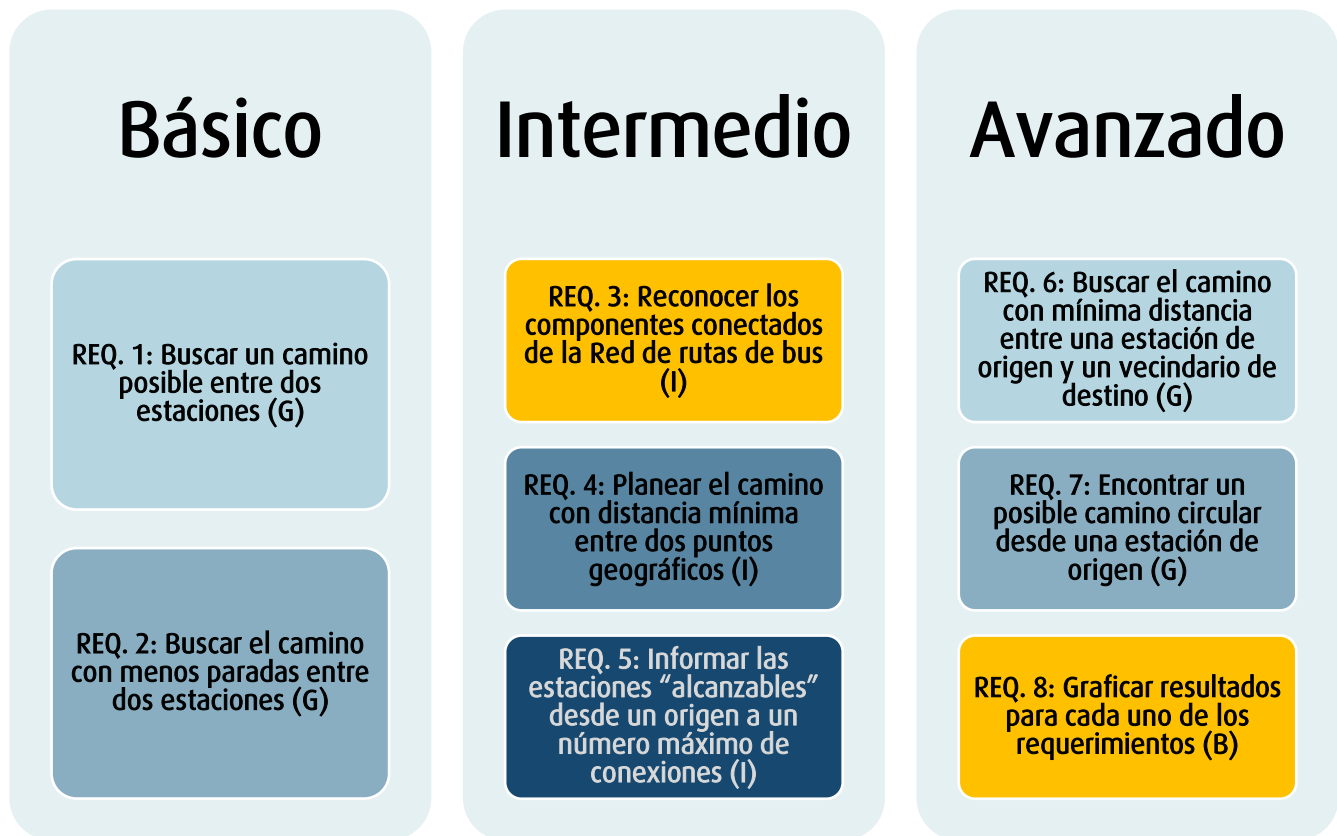
Node_ID	Code	Bus_Stop	Transport	Longitude	Latitude	District_Name	Neighborhood_Name	In Degree (Routes)	Out Degree (Routes)
780-19	780	BUS-19	Day bus stop	2.1633	41.4278	Horta-Guinardà	Horta	1	1
43-109	43	BUS-109	Day bus stop	2.1287	41.3301	Sants-Montjuïc	la Marina del Prat Vermell	1	1
101-11	101	BUS-11	Day bus stop	2.1893	41.4519	Sant Andreu	la Trinitat Vella	1	1
778-19	778	BUS-19	Day bus stop	2.1741	41.4087	Eixample	la Sagrada Família	1	1
1194-82	1194	BUS-82	Day bus stop	2.1723	41.4359	Nou Barris	la Guineueta	1	1
T-856	856	T-BUS	Transfer bus	2.1374	41.3311	Sants-Montjuïc	la Marina del Prat Vermell	1	1
T-1189	1189	T-BUS	Transfer bus	2.1692	41.4391	Nou Barris	la Guineueta	1	1
T-1440	1440	T-BUS	Transfer bus	2.1836	41.4630	Nou Barris	Vallbona	1	1
T-925	925	T-BUS	Transfer bus	2.1598	41.4113	Gràcia	el Camp d'en Grassot i Gràcia Nova	1	1
T-1394	1394	T-BUS	Transfer bus	2.1398	41.4194	Gràcia	Vallcarca i els Penitents	1	1

```
===== Bus-Stops-Routes Graph Specs =====
Graph Nodes: 228
Graph Edges: 104
First 5 & Last 5 Stations loaded in the Graph.
```

Node_ID	Code	Bus_Stop	Transport	Longitude	Latitude	District_Name	Neighborhood_Name	Degree (Rep Routes)
780-19	780	BUS-19	Day bus stop	2.1633	41.4278	Horta-Guinardà	Horta	1
43-109	43	BUS-109	Day bus stop	2.1287	41.3301	Sants-Montjuïc	la Marina del Prat Vermell	1
101-11	101	BUS-11	Day bus stop	2.1893	41.4519	Sant Andreu	la Trinitat Vella	1
778-19	778	BUS-19	Day bus stop	2.1741	41.4087	Eixample	la Sagrada Família	1
1194-82	1194	BUS-82	Day bus stop	2.1723	41.4359	Nou Barris	la Guineueta	1
T-856	856	T-BUS	Transfer bus	2.1374	41.3311	Sants-Montjuïc	la Marina del Prat Vermell	1
T-1189	1189	T-BUS	Transfer bus	2.1692	41.4391	Nou Barris	la Guineueta	1
T-1440	1440	T-BUS	Transfer bus	2.1836	41.4630	Nou Barris	Vallbona	1
T-925	925	T-BUS	Transfer bus	2.1598	41.4113	Gràcia	el Camp d'en Grassot i Gràcia Nova	1
T-1394	1394	T-BUS	Transfer bus	2.1398	41.4194	Gràcia	Vallcarca i els Penitents	1

## Parte 3: Desarrollo de los Requerimientos

El resumen de los requerimientos se muestra en la siguiente tabla y se explican detalladamente en la siguiente sección.



Para este reto se han identificado ocho (8) requerimientos, siete (7) obligatorios y un (1) bono. Divididos de acuerdo con su dificultad en nivel básico, intermedio y avanzado. Además, es importante resaltar que tres (3) de estos requerimientos se deben desarrollar de forma individual y los cinco (5) restantes son grupales.

Por último, es importante resaltar que los dos (2) requerimientos avanzados son de análisis, es decir que la algorítmica a implementar debe ser diseñada por los estudiantes y **la librería DISCLib no ofrece ningún algoritmo directo para su solución.**

**NOTA:** En caso de que el equipo este conformado solamente por dos integrantes deberán resolver solamente cinco (5) requerimientos obligatorios; el equipo podrá escoger cuál de los requerimientos básicos va a realizar y la pareja podrá elegir dos (2) de los tres (3) requerimientos individuales a implementar.

## Requerimiento No. 1 (Grupal): Buscar un camino posible entre dos estaciones

Como pasajero **deseo** conocer si existe un camino entre una estación de origen y una de destino.

Los **parámetros de entrada** de este requerimiento son:

- Identificador de la estación origen (corresponde al identificador único de una estación de la forma Code-IdBus).
- Identificador de la estación destino (corresponde al identificador único de una estación de la forma Code-IdBus).

La **respuesta esperada** debe contener la siguiente información:

- La distancia total que tomará el camino entre la estación origen y la estación destino.
- El total de estaciones que contiene el camino solución.
- El total de transbordos de ruta que debe realizar el usuario.
- Las estaciones que definen el camino resultante (incluyendo el origen y el destino) con la siguiente información:
  - El identificador de la estación.
  - La distancia a la siguiente estación en el camino.

**Recomendación:**

- El usuario solo ingresa los identificadores de las estaciones de la forma Code-IdBus.



## Requerimiento No. 2 (Grupal): Buscar el camino con menos paradas entre dos estaciones

Como pasajero **deseo** conocer el camino “más corto” (es decir, con menor número de conexiones) entre una estación de origen y otra de destino.

Los **parámetros de entrada** de este requerimiento son:

- Identificador de la estación origen (corresponde al identificador único de una estación de la forma Code-IdBus).
- Identificador de la estación destino (corresponde al identificador único de una estación de la forma Code-IdBus).

La **respuesta esperada** debe contener la siguiente información:

- La distancia total que tomará el camino entre la estación origen y la estación destino.
- El total de estaciones que contiene el camino solución.
- El total de transbordos de ruta que debe realizar el usuario.
- Las estaciones que definen el camino resultante (incluyendo el origen y el destino) con la siguiente información:
  - El identificador de la estación.
  - La distancia a la siguiente estación en el camino.

**Nota:** Los algoritmos utilizados para solucionar los requerimientos 1 y 2 deben ser diferentes.

**Recomendación:**

- El usuario solo ingresa los identificadores de las estaciones de la forma Code-IdBus.

### Requerimiento No. 3 (Individual): Reconocer los componentes conectados de la Red de rutas de bus

Como analista de movilidad **deseo** conocer si existen componentes conectados en el grafo y de ser así cuántos son y conocer sus características.

**No** se requieren **parámetros de entrada** para este requerimiento, se utiliza la totalidad del grafo.

La **respuesta esperada** debe contener la siguiente información:

- El total de componentes conectados dentro del grafo.
- Mostrar los 5 componentes conectados más grandes (de mayor a menor número de estaciones en la componente fuertemente conectada):
  - El número de estaciones que pertenecen a dicho componente.
  - Los identificadores de las tres primeras y tres últimas estaciones pertenecientes al componente.

Si existen menos de 5 componentes dentro del grafo, mostrar la información de todos los componentes conectados identificados.

**Recomendación:**

- Ordene los componentes conectados por su tamaño (el número de estaciones por componente) y su índice (identificador) para facilitar la lectura de los resultados.

**EJEMPLO:** Deseo conocer los componentes fuertemente conectados de la red de buses de Barcelona.

## Requerimiento No. 4 (Individual): Planear el camino con distancia mínima entre dos puntos geográficos

Como analista de movilidad **deseo** ayudar a los usuarios a planear sus viajes en el sistema de transporte. Para ello se les informará el camino de distancia mínima que les servir entre dos localizaciones geográficas. De esta manera el usuario podrá planear mejor sus rutas y visitas dentro de la ciudad, para facilitar la visita a sitios turísticos o importantes y ayudándoles a retomar su viaje rápidamente.

Los **parámetros de entrada** de este requerimiento son:

- Localización geográfica origen (longitud y latitud) del usuario.
- Localización geográfica destino (longitud y latitud) del usuario.

La **respuesta esperada** debe contener la siguiente información:

- La distancia entre la localización de origen y la estación de bus más cercana.
- La distancia total que tomará el recorrido entre la estación origen y la estación destino.
- La distancia entre la estación destino más cercana y la localización destino.
- El total de estaciones que contiene el camino solución.
- El total de transbordos de ruta que debe realizar.
- Las estaciones que definen el camino resultante (incluyendo el origen y el destino) con la siguiente información:
  - El identificador de la estación.
  - La distancia a la siguiente estación en el camino.

**Recomendación:**

- Buscar la estación de bus más cercana a la localización geográfica de origen y de destino utilizando la función **Haversine**.
- Buscar el camino de distancia mínima entre las estaciones de origen y la de destino determinadas previamente.

## Requerimiento No. 5 (Individual): Localizar las estaciones “alcanzables” desde un origen a un número máximo de conexiones dado

Como analista de movilidad **deseo** conocer las estaciones a las cuales se puede llegar desde una estación inicial y que estén a un máximo número de conexiones determinado. De esta manera le puedo facilitar a los usuarios visitar múltiples lugares de la manera más rápida posible.

Los **parámetros de entrada** de este requerimiento son:

- Identificador de la estación origen (corresponde al identificador único de una estación de la forma Code-IdBus).
- Número de conexiones permitidas desde la estación origen.

La **respuesta esperada** debe contener la siguiente información:

- Un reporte consolidado que incluya la información de las estaciones “alcanzables” con el número de conexiones dado con la siguiente información:
  - El identificador de la estación.
  - La geolocalización de la estación (latitud y longitud)
  - Longitud del camino desde la estación origen a la estación alcanzada.

### Recomendación:

- Es posible que si la estación de consulta está hacia el final de una ruta y/o el número de conexiones es muy alto NO se alcance a llegar a estaciones en alguna ruta.
- Seleccionar como estaciones de origen las estaciones de bus que se conecten con estaciones de TRANSBORDO para tener más opciones de rutas posibles donde buscar.

## Requerimiento No. 6 (Grupal): Buscar el camino con distancia mínima entre una estación de origen y un vecindario de destino

Como pasajero **deseo** conocer el camino de distancia mínima entre una estación de origen y un vecindario destino de la ciudad.

Los **parámetros de entrada** de este requerimiento son:

- Identificador de la estación origen (corresponde al identificador único de una estación de la forma Code-IdBus).
- El identificador del vecindario (Neighborhood) destino.

La **respuesta esperada** debe contener la siguiente información:

- La distancia total que tomará el recorrido entre la estación origen y la estación destino del vecindario destino.
- El total de estaciones que contiene el camino.
- El total de transbordos de ruta que debe realizar.
- El camino calculado entre las estaciones (incluyendo el origen y el destino) y para cada estación en el camino se debe mostrar la siguiente información:
  - El identificador de la estación.
  - El identificador del vecindario de la estación.
  - La distancia a la siguiente estación en el camino.

**Recomendación:**

- **DISCLIB** no provee ningún algoritmo por defecto capaz de resolver este requerimiento. Los estudiantes deben implementar su propia lógica/algoritmo de solución utilizando las funciones básicas/primitivas del ADT Graph.

## Requerimiento No. 7 (Grupal): Encontrar un posible camino circular desde una estación de origen

Como pasajero **deseo** encontrar un camino circular saliendo de una estación inicial. Esto me sirve para planear rutas turísticas rápidas dentro de la ciudad. E importante, este camino debe permitirme volver al origen (estación inicial) a través de una estación de transbordo de ser necesario.

Los **parámetros de entrada** de este requerimiento son:

- Identificador de la estación de origen (corresponde al identificador único de una estación de la forma Code-IdBus).

La **respuesta esperada** debe contener la siguiente información:

- La distancia total que tomará el recorrido del camino circular. La distancia total de desplazamiento debe ser mayor a 0.0.
- El total de estaciones que contiene el camino. El total de estaciones debe ser mayor a 1.
- El total de transbordos de ruta que deben realizarse.
- El camino calculado entre las estaciones (incluyendo el origen y el destino) y para cada estación en el camino se debe mostrar la siguiente información:
  - El identificador de la estación.
  - La distancia a la siguiente estación en el camino.

**Recomendación:**

- Para que el camino se considere circular se debe garantizar que el usuario coge un bus en la estación inicial y se mueve a una o varias estaciones con ubicación geográfica diferente a la estación inicial.
- Seleccionar como estaciones de origen las estaciones de bus que se conecten con estaciones de Transbordo para tener más opciones de rutas posibles donde buscar.

## Requerimiento No. 8 (Bono Grupal): Graficar resultados para cada uno de los requerimientos

Como analista de movilidad **deseo** graficar visualmente todos los requerimientos de la aplicación. De esta manera puedo facilitar el uso y análisis de resultados a los demás usuarios.

Se otorgará una bonificación a los equipos de trabajo que grafiquen los mapas de resultados para cada uno de los requerimientos obligatorios del enunciado (del primer al séptimo).

- **Básico**
  - REQ. 1: Buscar un camino posible entre dos estaciones (G)
  - REQ. 2: Buscar el camino con menos paradas entre dos estaciones (G)
- **Intermedio**
  - REQ. 3: Reconocer los componentes conectados de la Red de rutas de bus (I)
  - REQ. 4: Planear el camino con distancia mínima entre dos puntos geográficos (I)
  - REQ. 5: Informar las estaciones “alcanzables” desde un origen a un número máximo de conexiones (I)
- **Avanzado**
  - REQ. 6: Buscar el camino con mínima distancia entre una estación de origen y un vecindario de destino (G)
  - REQ. 7: Encontrar un posible camino circular desde una estación de origen (G)

Para completar este requerimiento recomendamos utilizar la librería por extensión de Python llamada “**folium**” que se puede instalar en su ambiente por medio del comando “**pip install folium**”.

Para más información sobre esta librería dirigirse a los siguientes enlaces:

- Enlace oficial de PYPI, URL: <https://pypi.org/project/folium/>
- Enlace oficial de la librería, URL: <https://github.com/python-visualization/folium>
- Enlace al tutorial de la Librería. URL: <https://python-visualization.github.io/folium/quickstart.html>

## Parte 4: Análisis de resultados

Dentro del proyecto debe incluir un documento en la carpeta **Docs** en formato **PDF** donde se evidencie el análisis de complejidad, las pruebas de tiempos de ejecución y de memoria utilizada para cada requerimiento. Se sugiere que el documento tenga la siguiente distribución del contenido:

- Nombres, código y correo Uniandes de los integrantes del grupo.
- Para los requerimientos individuales se debe indicar que estudiante del equipo lo realizó.
- Análisis de complejidad de cada uno de los requerimientos en **Notación O**.
- Breve justificación de la complejidad temporal dada.
- Pruebas de tiempos de ejecución y memoria utilizada para cada uno de los requerimientos. En estas pruebas se deben incluir:
  - las tablas de tiempos de ejecución registrados para los requerimientos del 1 al 7 (incluye el bono si aplica).
  - Gráficas de los tiempos de ejecución registrados para los requerimientos del 1 al 7 (incluye el bono si aplica).
  - Un análisis de resultados comparándolo con la estimación en **Notación O**.

### Recomendaciones:

- Tomen como guía las herramientas, metodología y análisis realizados en los laboratorios; en especial los laboratorios 4 y 5.
- Ejecute las pruebas de los requerimientos siempre con los mismos parámetros de entrada y con los archivos más grande que pueda procesar su computador.

## Entrega

Para realizar la entrega del reto deben:

1. Agregar los usuarios de los monitores y profesores del curso a su organización de GitHub para hacer la entrega adecuada de la actividad,
2. Dar permisos adecuados repositorio a los monitores y al profesor, de lo contrario el taller NO podrá ser calificado,
3. Asegurarse que la visibilidad del repositorio entregado sea privada y que solo pueda accederse con los permisos configurados para los integrantes del grupo,
4. Enviar el enlace de GitHub en la actividad correspondiente dentro de Bloque Neón Uniandes (BrightSpace).
5. Incluir en el repositorio GIT todo el material, código y documentos solicitados durante la actividad.

**IMPORTANTE:** Recuerde que solo se calificará el material hasta el último **COMMIT** realizado previo a la fecha límite **(11:59 PM) del 6 de diciembre de 2022**.



## Apéndice 1: como crear el grafo para el caso de la red de buses de Barcelona

Este es un ejemplo ilustrativo (simplificado) de las rutas de bus 100, 109 y 110 de Barcelona.

Extracto de datos de las rutas con 9 estaciones exclusivas (con valor N al final) y 1 estación compartida (con código 26 y valor S al final):

### Bus\_Stop

21,...,BUS-100,...,N  
22,...,BUS-100,...,N  
23,...,BUS-100,...,N  
24,...,BUS-109,...,N  
25,...,BUS-109,...,N  
26,...,BUS-109,...,S  
26,...,BUS-110,...,S  
26,...,BUS-100,...,S  
27,...,BUS-109,...,N  
28,...,BUS-110,...,N  
29,...,BUS-110,...,N  
30,...,BUS-110,...,N

### Edges

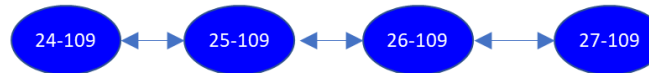
21, BUS-100,22  
22, BUS-100,23  
23, BUS-100,26  
24, BUS-109,25  
25, BUS-109,26  
26, BUS-109,27  
26, BUS-110,28  
28,BUS-110,29  
29,BUS-110,30

### Paso 1/3: Definir las rutas con sus estaciones en el grafo

1. Procesamiento de las 4 paradas de la ruta 100



2. Procesamiento de las 4 paradas de la ruta 109



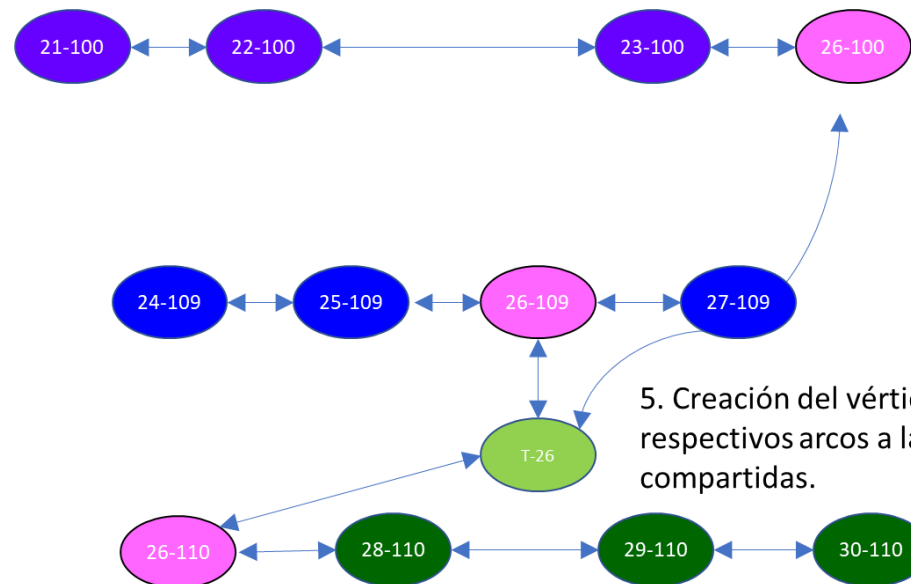
3. Procesamiento de las 4 paradas de la ruta 110



NOTA: Se usa los dos archivos para saber que arcos conectan que parada de bus.

## Paso 2/3: Completar las rutas con estaciones compartidas 109-110-100

4. Identificación de las paradas compartidas. Hay tres estaciones que tienen el Code con valor 26 (y que tendrán la misma ubicación) con el valor de transbordo en "S"



5. Creación del vértice T-26 con los respectivos arcos a las estaciones compartidas.

### Paso 3/3: Terminar la creación de las rutas dentro el grafo

