

# Reto 4: Toronto Bikeshare

---

## Objetivo

El objetivo de este reto es poner en práctica los conceptos aprendidos en clase acerca del TAD Grafo. Específicamente se desea:

1. Utilizar grafos, en conjunto con las demás estructuras de datos del curso para solucionar los requerimientos del reto.
2. Utilizar adecuadamente el patrón MVC (Modelo-Vista-Controlador).
3. Aprender a cargar y procesar en memoria datos en formato CSV.
4. Utilizar adecuadamente el administrador de versiones GIT y GitHub.
5. Aprender a trabajar en equipo.

## Fecha Límite de Entrega

Miércoles 1 de junio de 2022, 11:59 p.m.

## 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 **agregaron** las recomendaciones para implementar la limpieza y carga de datos.
- 3) Se **agregaron** las recomendaciones para implementar los requerimientos del 1 al 6.
- 4) Se **agregaron** los ejemplos funcionales de la carga de datos utilizando el archivo con sufijo -small.
- 5) Se **agregaron** los ejemplos para los requerimientos del 1 al 6 utilizando el archivo con sufijo -small.

## Contexto

El tema del proyecto está relacionado con el sistema bicicletas públicas **Bike Share** implementado en la ciudad de **Toronto, Canadá**<sup>1</sup>

---

<sup>1</sup> Bike Share Toronto, URL: <https://bikesharetoronto.com/>

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, 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.

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.

## Fuente de Datos

Los datos para este reto fueron tomados del repositorio "About Bike Share Toronto Ridership Data"<sup>2</sup> que contiene la información de dominio público dispuesta por la ciudad de Toronto Canadá para este proyecto de transporte público.

Los archivos originales para utilizar son los del año 2021, que contienen cada mes del año y contienen la información necesaria para completar los requerimientos del reto. en la Tabla 1 puede verse el resumen de los campos para el análisis.

Nombre de la Columna	Descripción
Trip Id	Identificador único del viaje
Trip Duration	Duración del viaje en segundos
Start Station Id	Identificador de la estación de origen del viaje
Start time	Tiempo de inicio del viaje (formato "%m/%d/%Y %H:%M")
Start Station Name	Nombre de la estación de origen
End Station Id	Identificador de la estación de destino del viaje
End Time	Tiempo de llegada al destino del viaje (formato "%m/%d/%Y %H:%M")
End Station Name	Nombre de la estación de destino.
Bike Id	Identificador único de la bicicleta
User type	Tipo de usuario (Annual Member, Casual Member)

Tabla 1. Descripción de los campos del archivo CS en el portal Bikeshare de Toronto, Canadá<sup>3</sup>.

<sup>2</sup> About Bike Share Toronto Ridership Data, URL: <https://open.toronto.ca/dataset/bike-share-toronto-ridership-data/>

<sup>3</sup> Tomado de **About Bike Share Toronto Ridership Data**.

# Trabajo Propuesto

## Parte 1: Configuración del Repositorio

Complete los siguientes pasos para configurar su repositorio de trabajo:

- a) 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>
- b) Renombre el repositorio de su grupo con el esquema **Reto4-G<<Número del grupo>>** ej.: **Reto4-G01** para el grupo 1 de la sección 2.
- c) Edite el **README** del repositorio e incluya los nombres completos, correo Uniandes y códigos de los miembros del equipo de trabajo.
- d) Realice el procedimiento según lo aprendido en clase para clonar el repositorio en su máquina local y sincronizarlo con su repositorio en GitHub.
- e) Descargue los datos desde el aula Bloque Neón 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. (ej.: el archivo del *bike-share-ridership-2021-utf8-20pct.csv* contiene un quinto de los datos con 2000 registros y el archivo *bike-share-ridership-2021-utf8-large.csv* contiene la totalidad de los datos con 10000 registros). Esto facilita la implementación y pruebas en computadores con memoria RAM y procesadores reducidos.

Para responder a los requerimientos a continuación, ustedes deberán cargar la información de los archivos de viajes de bicicleta y construir un grafo que representa el comportamiento de los usuarios.

Los viajes tienen una estación de origen y un destino, por lo que pueden representar arcos donde la estación de origen y la de destino, son los **vértices** del grafo; esta información puede ayudar a tener una visión general del uso de las bicicletas por parte de los usuarios.

Cada **arco** del grafo tendrá como peso, el promedio de los tiempos reportados por todos los viajeros que han iniciado y terminado un viaje en un par de estaciones.

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

- El total de viajes obtenidos de los datos.
- El total de vértices del grafo.
- El total de arcos del grafo.
- Mostrar los primeros cinco y últimos cinco vértices registrados en el grafo con las siguientes características:
  - ID de la estación.
  - Nombre de la estación.

- Numero de viajes de salida y de llegada.
- Rutas de entrada y salida del vértice (grado de entrada y salida).

### Recomendación:

- Recuerde limpiar los datos de los viajes para los vértices de los grafos utilizando los identificadores ("Start Station Id" y "End Station Id") y nombres ("Start Station Name" y "End Station Name") de los datos. EXISTEN identificadores repetidos para estaciones con diferente nombre. Para esto siga los pasos del siguiente ejemplo"

Ej.: cree un Mapa auxiliar para identificar los nodos del grafo por el nombre. Es decir, la llave es el nombre de la estación y el valor (que será el vértice del grafo) será la concatenación del ID y el nombre (<<ID>>-<<Name>>)

- **ID:** 7001, **Name:** "Wellesley Station Green P" pasa a ser **ID:** "7001-Wellesley Station Green P"
- **ID:** 7001, **Name:** "Lower Jarvis St / The Esplanade" pasa a ser **ID:** "7001-Lower Jarvis St / The Esplanade"

El mapa por lo tanto quedaría de la siguiente forma:

- **Key:** "Wellesley Station Green P", **Value:** "7001-Wellesley Station Green P"
- **Key:** "Lower Jarvis St / The Esplanade", **Value:** "7001-Lower Jarvis St / The Esplanade"

Donde la llave es el nombre de la estación y el valor el identificador único del vértice de esa estación en el grafo.

Con este mapa auxiliar puede crear el grafo adecuadamente y seguir haciendo las búsquedas de las estaciones por los nombres que los usuarios conocen.

- Después de limpiar los datos para los vértices, recuerde ajustar los identificadores de origen y destino ("Start Station Id" y "End Station Id") de los viajes para encontrar los arcos no repetidos reales para crear los grafos.
- Ignore todos los viajes que tengan información fundamental incompleta, es decir que los valores de "Start Station Id", "End Station Id", "Trip Duration" y "Bike Id" sean Nulos o vacíos.
- Ignore todas las autorreferencias en los viajes, es decir que la estación de origen ("Start Station Id") sea la misma que la estación de destino ("End Station Id").
- Ignore todos los viajes que tengan un tiempo de viaje("Trip Duration") menor o igual a cero (0.0) para evitar errores de cálculo.
- De considerarlo necesario, puede crear un grafo no dirigido como estructura auxiliar considerando lo siguiente: solo cuente las rutas reciprocas entre nodos, ej.: solo es válido si existe un arco de 7001 a 7002 y de 7002 a 7001.
- Al crear un arco en el grafo no dirigido el peso debe ser el tiempo ponderado promedio más alto entre los dos nodos. Ej.: 7001 a 7002 tiene un peso promedio de 5500 s y de 7002 a 7001 un peso de 18700. Se selecciona el peso de 18700 para el arco no dirigido entre las estaciones 7001 y 7002.

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

```
===== Bike-Trips-Time DiGraphs =====
35751 trips in CSV file...
33893 trips with no self-reference data...
35689 trips with complete data...
62 trips with incomplete data...
1858 trips with 0.0 in 'Trip Duration'...

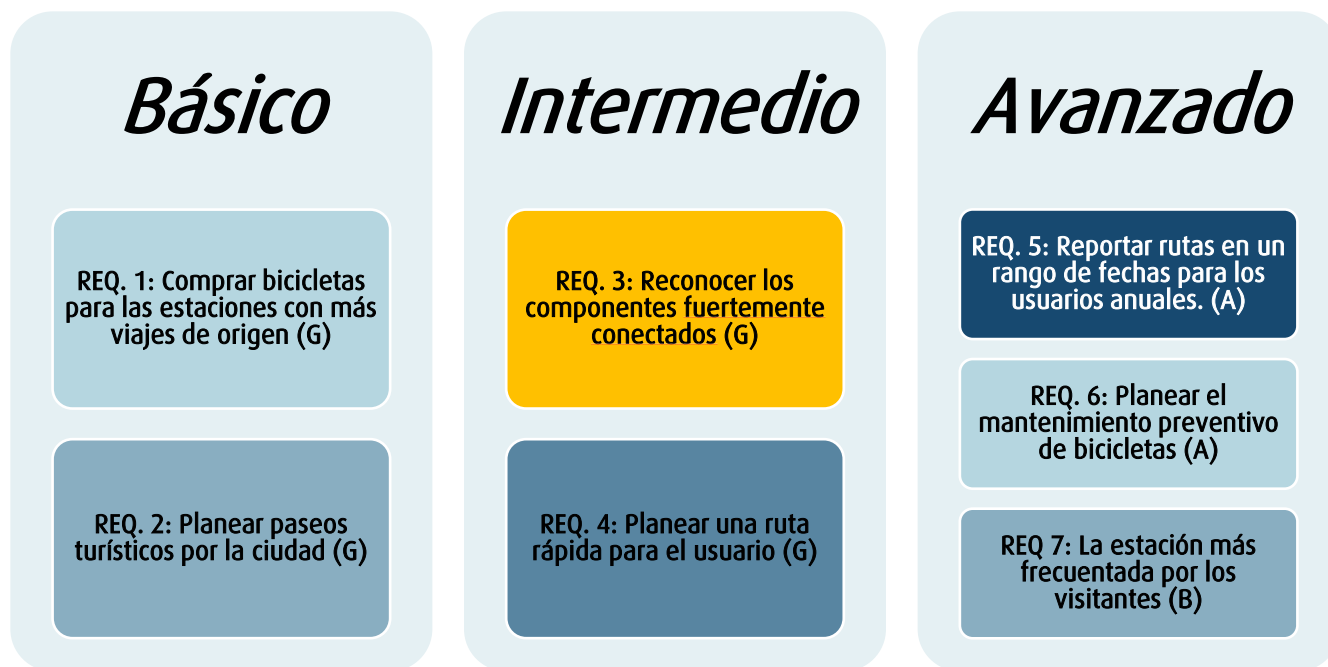
709 unique stations loaded...
24113 clean avg trips data loaded...

--- DiGraph specs ---
Nodes: 709 & Edges: 24113
First 5 & Last 5 Stations loaded in the DiGraph.
+-----+-----+-----+-----+-----+-----+
| Station ID | Station Name | In Degree (Routes) | Out Degree (Routes) | In Trips | Out Trips |
+-----+-----+-----+-----+-----+-----+
| 7000 | Fort York Blvd / Capreol Ct | 129 | 115 | 206 | 234 |
+-----+-----+-----+-----+-----+-----+
| 7001 | Lower Jarvis St / The Esplanade | 1 | 2 | 2 | 1 |
+-----+-----+-----+-----+-----+-----+
| 8001 | Wellesley Station Green P | 92 | 87 | 123 | 130 |
+-----+-----+-----+-----+-----+-----+
| 7002 | St. George St / Bloor St W | 96 | 97 | 139 | 141 |
+-----+-----+-----+-----+-----+-----+
| 7003 | Madison Ave / Bloor St W | 50 | 54 | 69 | 63 |
+-----+-----+-----+-----+-----+-----+
| 7683 | Unknown Station | 5 | 5 | 5 | 5 |
+-----+-----+-----+-----+-----+-----+
| 7684 | Unknown Station | 5 | 5 | 5 | 5 |
+-----+-----+-----+-----+-----+-----+
| 7685 | Unknown Station | 2 | 5 | 5 | 2 |
+-----+-----+-----+-----+-----+-----+
| 7686 | Unknown Station | 3 | 1 | 1 | 3 |
+-----+-----+-----+-----+-----+-----+
| 7687 | Unknown Station | 1 | 1 | 1 | 1 |
+-----+-----+-----+-----+-----+-----+

--- Graph specs ---
Nodes: 709 & Edges: 3969
First 5 & Last 5 Stations loaded in the Graph.
+-----+-----+-----+-----+
| Station ID | Station Name | Degree (Rep Routes) | Trips |
+-----+-----+-----+-----+
| 7000 | Fort York Blvd / Capreol Ct | 63 | 440 |
+-----+-----+-----+-----+
| 7001 | Lower Jarvis St / The Esplanade | 0 | 3 |
+-----+-----+-----+-----+
| 8001 | Wellesley Station Green P | 36 | 253 |
+-----+-----+-----+-----+
| 7002 | St. George St / Bloor St W | 39 | 280 |
+-----+-----+-----+-----+
| 7003 | Madison Ave / Bloor St W | 10 | 132 |
+-----+-----+-----+-----+
| 7683 | Unknown Station | 0 | 10 |
+-----+-----+-----+-----+
| 7684 | Unknown Station | 0 | 10 |
+-----+-----+-----+-----+
| 7685 | Unknown Station | 0 | 7 |
+-----+-----+-----+-----+
| 7686 | Unknown Station | 0 | 4 |
+-----+-----+-----+-----+
| 7687 | Unknown Station | 0 | 2 |
+-----+-----+-----+-----+
```

## 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 siete (7) requerimientos, seis (6) obligatorios y un (1) bono. Divididos de acuerdo con su dificultad en nivel básico, intermedio y avanzado. Adicionalmente, es importante resaltar que dos (2) de estos requerimientos 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.

## Requerimiento No. 1 (Grupal): Comprar bicicletas para las estaciones con más viajes de origen.

Como el equipo de análisis **deseo** indicar las 5 estaciones desde donde se inician más viajes por parte de los usuarios para adquirir nuevas bicicletas y ponerlas a disposición de los usuarios.

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

La **respuesta esperada** debe contener:

- las 5 estaciones solicitadas con la siguiente información:
  - Identificador de la estación.
  - Nombre de la estación.
  - Cantidad de viajes que han iniciado en esa estación.
  - El total de viajes iniciados por tipo de usuario (casual y por suscripción).
  - La fecha (formato "MM/DD/AAAA") del año en la que más viajes se inician.
  - la hora del día (0:00 – 0:59, 1:00 – 1:59 am, 2:00 – 2:59, ..., 23:00 – 23:59) en la que más viajes se inician.

### Recomendación:

- En caso de que no exista algún valor dentro de los campos solicitados de la estación o los viajes completarlo con el valor "Desconocido" o "Unknown".
- El día del año y la hora del día con más viajes son cálculos independientes.

**EJEMPLO:** Deseo conocer las cinco estaciones con más viajes de salida en el sistema de Bikeshare Toronto.

```
===== Req No. 1 Inputs =====
TOP 5 stations inside in the Bikeshre Network.
Number of stations in the network: 709

===== Req No. 1 Answer =====
TOP 5 stations used as trip origins are:
```

Station ID	Station Name	Out Trips	Suscriber Out Trips	Tourist Out Trips	Out Degree (Routes)	Rush hour	Rush date
7076	York St / Queens Quay W	333	135	198	177	19:00h, 77	2021-09-11, 11
7242	Lake Shore Blvd W / Ontario Dr	296	116	180	78	19:00h, 65	2021-04-24, 14
7171	Ontario Place Blvd / Lake Shore Blvd W (East)	295	116	179	97	17:00h, 66	2021-05-15, 17
7203	Bathurst St/Queens Quay(Billy Bishop Airport)	278	125	153	138	17:00h, 59	2021-05-16, 15
7016	Bay St / Queens Quay W (Ferry Terminal)	267	104	163	128	18:00h, 59	2021-08-02, 11

## Requerimiento No. 2 (Grupal): Planear paseos turísticos por la ciudad.

Como el equipo de análisis **deseo** generar las posibles rutas de viaje dadas unas condiciones de tiempo disponible y un número mínimo de estaciones para ayudar a los turistas a planificar un viaje por la ciudad.

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

- El nombre de la estación de inicio.
- La disponibilidad del usuario para su paseo.
- El número mínimo de estaciones de parada para la ruta (sin incluir la estación de inicio)
- El máximo número de rutas de respuesta.

La **respuesta esperada** debe contener:

- las posibles rutas (que no excedan el número máximo de respuesta, sin ningún orden en particular). Por cada una de ellas se debe mostrar la siguiente información:
  - El número de estaciones visitadas.
  - La secuencia de las estaciones de cada viaje (indicando el ID y nombre de cada una de ellas).
  - El tiempo total de la ruta.

**Recomendación:**

- En caso de que no exista algún valor dentro de los campos solicitados de la estación o los viajes completarlo con el valor "Desconocido" o "Unknown".
- Tenga en cuenta que el tiempo para volver a la estación de origen, esto quiere decir que el tiempo disponible para paseos turísticos se reduce a la mitad.



EJEMPLO: Deseo conocer **9 rutas posibles** desde la estación 'Sherbourne St / Wellesley St E'" para recorrer en **45 minutos** o menos y con por lo menos **5 estaciones** dentro de sus trayectos.

===== Req No. 2 Inputs =====  
Available time: 45.0 [min]  
Available time: 2700.0 [sec]  
Minimum number of stations: 5  
Maximum number of routes: 9  
Starting station: 'Sherbourne St / Wellesley St E'

===== Req No. 2 Answer =====  
There are 27 possible routes from station:  
7042:'Sherbourne St / Wellesley St E'

++++ The TOP 9 routes are: ++++

The first 3 and last 3 possible routes are:

Stop stations	Route time [s]	Route time [min]	Roundtrip time [min]	Station IDs	Station names
6	707.6	11.79	23.59	7042 -> 7311 -> 7031 -> 7024 -> 7272 -> 8001	Sherbourne St / Wellesley St E -> Sherbourne St / Isabella St -> Jarvis St / Isabella St -> Dundonald St / Church St -> Yonge St / Dundonald St - SMART -> Wellesley Station Green P
9	1295.1	21.58	43.17	7042 -> 7311 -> 7031 -> 7024 -> 7280 -> 8385 -> 7281 -> 7030 -> 7006	Sherbourne St / Wellesley St E -> Sherbourne St / Isabella St -> Jarvis St / Isabella St -> Dundonald St / Church St -> Charles St E / Jarvis St - SMART -> 20 Charles St E - SMART -> Charles St W / Balmuto St - SMART -> Bay St / Wellesley St W -> Bay St / College St (East Side)
7	788.6	13.14	26.29	7042 -> 7311 -> 7031 -> 7024 -> 7272 -> 7548 -> 7008	Sherbourne St / Wellesley St E -> Sherbourne St / Isabella St -> Jarvis St / Isabella St -> Dundonald St / Church St -> Yonge St / Dundonald St - SMART -> St Joseph St / Bay St - SMART -> "Wellesley St / Queens Park Cres"
6	594.6	9.91	19.82	7042 -> 7311 -> 7031 -> 7024 -> 7272 -> 7548	Sherbourne St / Wellesley St E -> Sherbourne St / Isabella St -> Jarvis St / Isabella St -> Dundonald St / Church St -> Yonge St / Dundonald St - SMART -> St Joseph St / Bay St - SMART
8	1177.6	19.63	39.25	7042 -> 7311 -> 7031 -> 7024 -> 7280 -> 8385 -> 7384 -> 8648	Sherbourne St / Wellesley St E -> Sherbourne St / Isabella St -> Jarvis St / Isabella St -> Dundonald St / Church St -> Charles St E / Jarvis St - SMART -> 20 Charles St E - SMART -> Cumberland Ave / Bay St SMART -> Rosedale Station
6	1006	16.77	33.53	7042 -> 7311 -> 7264 -> 7034 -> 7126 -> 7677	Sherbourne St / Wellesley St E -> Sherbourne St / Isabella St -> Bloor St E / Huntley St - SMART -> Church St / Bloor St E -> Yonge St / Yorkville Ave -> 1303 Yonge St - SMART

## Requerimiento No. 3 (Grupal): Reconocer los componentes fuertemente conectados del sistema.

Como equipo de análisis **deseo** conocer si existen componentes fuertemente conectados en el grafo, y de ser así cuántos son e incluir sus características.

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

La **respuesta esperada** debe contener:

- El total de componentes fuertemente conectados dentro del grafo.
- En caso de que existan demasiados elementos, despliegue la siguiente información para los 3 primeros y 3 componentes fuertemente conectados:
  - El número de estaciones en el componente.
  - El identificador y nombre de la estación donde más viajes inician.
  - El identificador y nombre de la estación donde más viajes terminan.

**Recomendación:**

- En caso de que no exista algún valor dentro de los campos solicitados de la estación o los viajes completarlo con el valor “Desconocido” o “Unknown”.
- Ordene los componentes fuertemente 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 del sistema **Bikeshare Toronto**.

```
===== Req No. 3 Inputs =====

+++ calculating the strongly connected components +++
...

===== Req No. 3 Answer =====
There are 45 Strongly Connected Components (SCC) in the graph.

+++ The SCC details are: +++
The first 3 and last 3 of the SCC are:
```

SCC size	SCCID	Max out station ID	Max out station name	Max out station trips	Max in station ID	Max in station name	Max in station trips
660	26	7076	York St / Queens Quay W	333	7076	York St / Queens Quay W	426
6	11	7616	Waterfront Trail (Rouge Hill)	26	7616	Waterfront Trail (Rouge Hill)	29
1	44	7597	Yonge St / Golddale Rd	2	7597	Yonge St / Golddale Rd	0
1	2	7676	Unknown Station	0	7676	Unknown Station	1
1	1	8677	Unknown Station	0	8677	Unknown Station	0
1	0	8680	Unknown Station	0	8680	Unknown Station	0

## Requerimiento No. 4 (Grupal): Planear una ruta rápida para el usuario.

Como equipo de análisis **deseo** ayudar a los usuarios a planear su viaje en bicicleta por la ciudad, para ello les presentaré una ruta con el tiempo mínimo que le puede tomar a una persona ir de una estación de origen a otra de destino. De esta manera el viajero podrá planear mejor sus rutas y visitas dentro de la ciudad, mejorando la visita a sitios turísticos o importantes y ayudándoles a retomar su viaje rápidamente.

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

- Nombre de la estación origen.
- Nombre de la estación destino.

La **respuesta esperada** debe contener:

- El tiempo total que tomará el recorrido entre la estación origen y la estación destino.
- La ruta calculada entre las estaciones (incluyendo el origen y el destino) y para cada estación en la ruta se debe mostrar la siguiente información:
  - El número de identificación de la estación.
  - El nombre de la estación.
  - El tiempo promedio a la siguiente estación en la ruta.

**Recomendación:**

- En caso de que no exista algún valor dentro de los campos solicitados de la estación o los viajes completarlo con el valor "Desconocido" o "Unknown".
- En caso de que la ruta sea extremadamente larga (12 estaciones o más), limite la respuesta de las estaciones a las tres primeras y tres últimas estaciones y rutas dentro del camino.

**EJEMPLO:** Deseo conocer la ruta optima entre la estación de origen con nombre “Fort York Blvd / Capreol Ct” y la estación de destino “Yonge St / St Clair Ave”.

```

===== Req No. 4 Inputs =====
Start station: 'Fort York Blvd / Capreol Ct' with ID: 7000
End station: 'Yonge St / St Clair Ave' with ID: 7642

===== Req No. 3 Answer =====
+++ Departure bike station data: 'Fort York Blvd / Capreol Ct' +++
+-----+-----+-----+-----+-----+-----+-----+-----+
| Station ID | Station Name | Out Trips | In Trips | Rush hour | Rush date | In Degree (Routes) | Out Degree (Routes) |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7000 | Fort York Blvd | 234 | 206 | 18:00h, | 2021-09-03, | 115 | 129 |
| | / Capreol Ct | | | 49 | 7 | | |
+-----+-----+-----+-----+-----+-----+-----+-----+

+++ Arrival bike station data: 'Yonge St / St Clair Ave' +++
+-----+-----+-----+-----+-----+-----+-----+-----+
| Station ID | Station Name | Out Trips | In Trips | Rush hour | Rush date | In Degree (Routes) | Out Degree (Routes) |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7642 | Yonge St / St | 56 | 56 | 17:00h, | 2021-06-15, | 44 | 46 |
| | Clair Ave | | | 22 | 3 | | |
+-----+-----+-----+-----+-----+-----+-----+-----+

+++ Dijkstra's Trip details +++
- Number of stops: 5
- Number of routes: 4
- Total time: 1614.5 [sec]
- Total time: 26.91 [min]

- Path details:
+-----+-----+-----+-----+-----+-----+-----+-----+
| Start Station Id | Start Station Name | End Station Id | End Station Name | Avg Route Duration |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7000 | Fort York Blvd | 7160 | King St W / | 345.5 |
| | / Capreol Ct | | Tecumseth St | |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7160 | King St W / | 7275 | Queen St W / | 2 |
| | Tecumseth St | | James St | |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7275 | Queen St W / | 7276 | Bloor St W / | 523 |
| | James St | | Balmuto St | |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7276 | Bloor St W / | 7642 | Yonge St / St | 744 |
| | Balmuto St | | Clair Ave | |
+-----+-----+-----+-----+-----+-----+-----+-----+

- Stations details:
+-----+-----+-----+-----+-----+-----+-----+-----+
| Station ID | Station Name | Out Trips | In Trips | Rush hour | Rush date | In Degree (Routes) | Out Degree (Routes) |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7000 | Fort York Blvd | 234 | 206 | 18:00h, | 2021-09-03, | 115 | 129 |
| | / Capreol Ct | | 49 | 7 | | |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7160 | King St W / | 135 | 119 | 18:00h, | 2021-07-10, | 80 | 90 |
| | Tecumseth St | | 25 | 5 | | |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7275 | Queen St W / | 110 | 85 | 16:00h, | 2021-09-24, | 61 | 76 |
| | James St | | 28 | 5 | | |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7276 | Bloor St W / | 58 | 60 | 18:00h, | 2021-08-12, | 52 | 48 |
| | Balmuto St | | 15 | 3 | | |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7642 | Yonge St / St | 56 | 56 | 17:00h, | 2021-06-15, | 44 | 46 |
| | Clair Ave | | 22 | 3 | | |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

## Requerimiento No. 5 (Análisis Grupal): Reportar rutas en un rango de fechas para los usuarios anuales.

Como equipo de análisis **deseo** crear un reporte para los usuarios que tienen membresía anual puedan planear mejor sus recorridos dentro de la ciudad.

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

- Fecha inicial de consulta (formato “%m/%d/%Y”).
- Fecha final de consulta (formato “%m/%d/%Y”).

La **respuesta esperada** debe generar un reporte consolidado que incluya la siguiente información:

- El total de viajes realizados.
- El total de tiempo invertido en los viajes.
- La estación de origen más frecuentada.
- La estación de destino más utilizada.
- La hora del día en la que más viajes inician (0:00–0:59, 1:00–1:59 am, 2:00–2:59, ..., 23:00–23:59)
- La hora del día en la que más viajes terminan (0:00–0:59, 1:00–1:59 am, 2:00–2:59, ..., 23:00–23:59)

**Recomendación:**

- En caso de que no exista algún valor dentro de los campos solicitados de la estación o los viajes completarlo con el valor “Desconocido” o “Unknown”.

**EJEMPLO:** Deseo conocer los detalles de las estaciones utilizadas para los viajes realizados entre las fechas "02/24/2021" y "07/09/2021".

```
===== Req No. 5 Inputs =====
Analyze trips between 02/24/2021 and 07/09/2021

===== Req No. 5 Answer =====
21 trips between 02/24/2021 and 07/09/2021

609 unique stations loaded...
10805 clean avg trips data loaded...

--- DiGraph specs ---
Nodes: 609 & Edges: 10805
Total trip time: 17013059.0 [sec].
Total trip time: 4725.85 [h].

+++ Top out station data: +++
+-----+-----+-----+-----+
| Station ID | Station Name | Out Degree (Routes) | Out Trips |
+-----+-----+-----+-----+
| 7242 | Lake Shore Blvd | 60 | 184 |
| | W / Ontario Dr | | |
+-----+-----+-----+-----+

+++ Top in station data: +++
+-----+-----+-----+-----+
| Station ID | Station Name | In Degree (Routes) | In Trips |
+-----+-----+-----+-----+
| 7016 | Bay St / Queens | 83 | 159 |
| | Quay W (Ferry | | |
| | Terminal) | | |
+-----+-----+-----+-----+

+++ TOP out trips rush hour station data: +++
+-----+-----+-----+-----+
| Station ID | Station Name | In rush hour | In rush hour count |
+-----+-----+-----+-----+
| 7242 | Lake Shore Blvd | 20:00h | 39 |
| | W / Ontario Dr | | |
+-----+-----+-----+-----+

+++ TOP in trips rush hour station data: +++
+-----+-----+-----+-----+
| Station ID | Station Name | Out rush hour | Out rush hour count |
+-----+-----+-----+-----+
| 7076 | York St / Queens | 19:00h | 37 |
| | Quay W | | |
+-----+-----+-----+-----+
```

## Requerimiento No. 6 (Análisis Grupal): Planear el mantenimiento preventivo de bicicletas.

Como equipo de análisis deseo programar el mantenimiento de las bicicletas en el sistema para prevenir accidentes y subutilización de los vehículos.

El **parámetro de entrada** de este requerimiento es:

- El identificador de la bicicleta en el sistema.

La **respuesta esperada** debe generar un reporte consolidado que incluya la siguiente información:

- El total de viajes en los que ha participado dicha bicicleta.
- El total de horas de utilización de la bicicleta.
- La estación en la que más viajes se han iniciado en esa bicicleta.
- La estación en la que más viajes ha terminado dicha bicicleta.

**Recomendación:**

- En caso de que no exista algún valor dentro de los campos solicitados de la estación o los viajes completarlo con el valor "Desconocido" o "Unknown".

**EJEMPLO:** Deseo conocer los detalles de uso de la bicicleta con identificador 5282.

```
===== Req No. 6 Inputs =====
Analyze trips with the bike ID 5282

===== Req No. 6 Answer =====
21 trips with bike ID 5282

38 unique stations loaded...
21 clean avg trips data loaded...

--- DiGraph specs ---
Nodes: 38 & Edges: 21
Total trip time: 16936.0 [sec].
Total trip time: 4.7 [h].

+++ Top out station data: +++
+-----+-----+-----+-----+
| Station ID | Station Name | Out Degree (Routes) | Out Trips |
+=====+=====+=====+=====+
|          7171 | Ontario Place |                2 |          2 |
|                | Blvd / Lake   |                |           |
|                | Shore Blvd W  |                |           |
|                | (East)        |                |           |
+-----+-----+-----+-----+

+++ Top in station data: +++
+-----+-----+-----+-----+
| Station ID | Station Name | In Degree (Routes) | In Trips |
+=====+=====+=====+=====+
|          7076 | York St / Queens |                2 |          2 |
|                | Quay W          |                |           |
+-----+-----+-----+-----+
```



## Requerimiento No. 7 (BONO Grupal): La estación más frecuentada por los visitantes.

Como equipo de análisis **deseo** conocer la estación más frecuentada por los usuarios turísticos (identificados como aquellos usuarios que no tienen pago anual) dado un rango de tiempo.

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

- Nombre de la estación.
- Fecha y hora de inicio.
- Fecha y hora de finalización.

La **respuesta esperada** debe generar un reporte consolidado que incluya la siguiente información:

- El total de viajes que iniciaron en dicha estación en el rango de tiempo solicitado.
- El total de viajes que terminaron en dicha estación en el rango de tiempo solicitado.
- El viaje de mayor duración promedio saliendo de la estación de consulta.
- La estación donde terminaron la mayoría de los viajes que iniciaron en la estación.

**Recomendación:**

- En caso de que no exista algún valor dentro de los campos solicitados de la estación o los viajes completarlo con el valor "Desconocido" o "Unknown".

## Parte 4: Análisis de resultados

Cree un documento en formato **PDF** para la entrega y guárdelo en la carpeta **Docs** del repositorio, el documento debe contener las siguientes secciones:

- Nombres, código y correo Uniandes de los integrantes del grupo.
- Análisis de complejidad de cada uno de los requerimientos en Notación O.
- Breve justificación de la complejidad temporal dada.

## 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.
5. Incluir en el repositorio GIT todo el material, código y documentos solicitados.

**IMPORTANTE:** Recuerde que solo se calificará el material hasta el último **COMMIT** realizado previo a la fecha límite **(11:59 PM) del 1 de junio de 2022**.