

## Objetivo

El objetivo de este proyecto es poner en práctica los conceptos aprendidos en clase acerca del uso de grafos como estructura de datos.

## Fechas Límite de Entrega

**Diseño:** 14 de mayo, 11:59 p.m.

**Implementación:** 1 de junio, 9:00 a.m.

## Contexto

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, los asuntos relacionados con la regulación del transporte; y entre ellos las infracciones de tránsito; ya que, a través de estas las unidades administrativas buscan regular el comportamiento del transporte tanto público como privado. El tema del proyecto está relacionado con el sistema de comparendos implementado en la ciudad de Bogotá D.C.. Esta ciudad cuenta con un portal para el registro de información en diferentes aspectos de su funcionamiento (portal oficial de datos abiertos <http://datosabiertos.bogota.gov.co/>). Para el análisis del sistema de comparendos utilizaremos como fuente de información el portal oficial el sistema PRUDENCIA de Bogotá en el enlace: <http://datosabiertos.bogota.gov.co/dataset/comparendos-dei-2018-bogota-d-c1>.

En este proyecto, vamos a construir una aplicación que permita gestionar datos de la malla vial de la ciudad de Bogotá D.C. utilizando la información del repositorio de comparendos y estaciones de policía. La malla vial debe ser modelada como un grafo no dirigido el cual debe ser construido tomando como base la información suministrada.

## Fuentes de Datos

A continuación, se presenta una descripción de las fuentes de datos que se utilizarán en el proyecto.

## Comparendos

La información de los comparendos se debe construir a partir de la carga del archivo *Comparendos\_DEI\_2018\_Bogotá\_D.C.geojson* (formato JSON) que se pueden descargar del portal Web de consulta de datos abiertos del sistema PRUDENCIA de Bogotá en el enlace: <http://datosabiertos.bogota.gov.co/dataset/comparendos-dei-2018-bogota-d-c1>.

## Estaciones de Policia

La información de las estaciones de policía se debe construir a partir de la carga del archivo *estacionpolicia.geojson* (formato JSON) que se pueden descargar del portal Web de consulta de datos abiertos del sistema PRUDENCIA de Bogotá en el enlace: <https://datosabiertos.bogota.gov.co/dataset/estacion-de-policia-para-bogota>

El archivo de estaciones de policia tiene la siguiente estructura:

```
{
  "type": "FeatureCollection",
  "crs": {
    "type": "name",
    "properties": {
      "name": "EPSG:4686"
    }
  },
  "features": [
    {
      "type": "Feature",
      "id": 1637,
      "geometry": {
        "type": "Point",
        "coordinates": [
          -74.103158699999938,
          4.5856918000000633
        ]
      },
      "properties": {
        "OBJECTID": 1637,
        "EPOCOD_PLAN": 5,
        "EPOCOD_ENT": "137",
        "EPOCOD_PROY": "7507",
        "EPOANIO_GEO": 2016,
        "EPOFECHA_INI": 1464739200000,
        "EPOFECHA_FIN": 1609372800000,
        "EPODESCRIP": "Estación de Policía Antonio Nariño",
        "EPOEST_PROY": "TERM",
        "EPOINTERV_ESP": "NIES",
        "EPODIR_SITIO": "KR 24 CL 18 - 90 SUR",
        "EPOCOD_SITIO": "020102",
        "EPOLATITUD": 4.5856236599999995,
        "EPOLONGITU": -74.103130640000003,
      }
    }
  ]
}
```

```

        "EPOSERVICIO":"Tiene como finalidad asegurar y ejercer el
        control de la jurisdicción, prestar un servicio integral de
        vigilancia urbana y rural, desarrollar los procesos de
        prevención, disuasión, investigación y control de los delitos
        y contravenciones apoyados en la gestión territorial de la
        seguridad ciudadana con las autoridades locales político
        administrativas.",
        "EPOHORARIO":"24 horas",
        "EPOTELEFON":"5521110",
        "EPOCELECTR":"mebog.e15@policia.gov.co",
        "EPOCONTACT":"Policía Nacional",
        "EPOPWEB":"http://www.policia.gov.co",
        "EPOIUUPLAN":"UPZ38",
        "EPOIUSCATA":"002103",
        "EPOIULOCAL":"15",
        "EPOEASOCIA":"No Aplica",
        "EPOFUNCION":"Coercion",
        "EPOTEQUIPA":"Justicia",
        "EPONOMBRE":"Estación de Policía Antonio Nariño",
        "EPOIDENTIF":"EPO003",
        "EPOFECHA_C":1565740800000
    }
}

{
    ...

},

...

]

}

```

## Malla Vial de Bogotá

La malla vial de Bogotá, representada como un grafo donde los vértices son las intersecciones de las calles y los arcos las calles, puede ser utilizada en una gran cantidad de aplicaciones para navegación terrestre. A continuación, se mostrarán unos ejemplos con el contenido de los archivos de *bogota\_vertices.txt* y *bogota\_arcos.txt*:

Se representarán las intersecciones de la malla vial como los vértices del grafo. Estas intersecciones están representadas en el archivo *bogota\_vertices.txt* con id, longitud, y latitud:

```

0,-74.08921298299998,4.582989396000016
1,-74.08952746199998,4.582560966000017

```

Las vías de la malla vial que conectan las intersecciones están almacenadas en el archivo *bogota\_arcos.txt*. Estas vías se representan como un conjunto de adyacencias donde el

primer campo especifica el id del vértice origen, y los demás campos son los ids de sus vértices adyacentes:

0 1 733  
1 49 ...



## Carga de Información

Para responder a los requerimientos presentados más adelante, usted deberá cargar la información de los archivos: *Comparendos\_DEI\_2018\_Bogotá\_D.C.geojson*, *estacionpolicia.geojson* y el archivo json con el grafo de la malla vial de Bogotá creado por ustedes en el Taller 7; es importante anotar que solo es permitido leer **una vez** la información de los archivos.

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

1. Total de comparendos en el archivo.
2. Mostrar la información del comparendo (OBJECTID, *FECHA\_HORA*, INFRACCION, CLASE\_VEHICULO, TIPO\_SERVICIO, LOCALIDAD) con el mayor OBJECTID encontrado.
3. Total de Estaciones de Policía en el archivo
4. Mostrar la información de la estación de policía (OBJECTID, EPODESCRIP, EPODIR\_SITIO, EPOLATITUD, EPOLONGITU, EPOSERVICIO, EPOHORARIO, EPOTELEFON, EPOIULOCAL) con el mayor OBJECTID encontrado.
5. Total de vértices en el grafo de la malla vial de Bogotá
6. Mostrar la información del vértice (ID, LATITUD, LONGITUD) con el mayor ID encontrado.
7. Total de arcos en el grafo de la malla vial de Bogotá
8. Mostrar la información de los arcos (IDOrigen, IDDestino1, ..., IDDestinoN) con el mayor ID encontrado.

## Impresión en consola de muchos datos

En algunos de los requerimientos el conjunto de datos retornado puede ser muy grande; por lo tanto, se imprimirán solo los primeros **N datos**, donde N es una constante que inicialmente se puede igualar a 20.

## Diseño de la Solución

Para cada requerimiento se quiere usar la estructura más conveniente para que se logre la **mejor eficiencia en tiempo y en espacio**. Las soluciones deben estar basadas en el uso de las Estructuras de Datos genéricas. En cada uno de esos casos usted es libre de usar la estructura que considere apropiada, pero debe justificar su elección.

## Requerimientos - Parte Inicial (ambos estudiantes)

1. Dada una localización geográfica con latitud y longitud, encontrar el Id del vértice de la malla vial más cercano por distancia haversiana.
2. Adicionar la información de cada uno de los comparendos del año 2018 al grafo de la malla vial. Para este fin, ubique el vértice de la malla vial más cercano a la ubicación geográfica de cada comparendo y sobre este vértice almacene la información del comparendo que considere relevante.
3. Agregar al grafo información de costo. El grafo tendrá 2 costos en sus arcos:
  - a. El primer costo asociado a un arco es la **distancia haversiana** (en kilómetros) entre las localizaciones geográficas de los vértices que conecta. Esta distancia es calculada en el taller 7.
  - b. El segundo costo asociado a un arco es el **total de comparendos** entre los vértices que conecta.
4. Adicionar la información de cada una de las estaciones de policía al grafo. Para este fin, ubique el vértice de la malla vial más cercano a la ubicación geográfica de cada estación de policía y sobre este vértice almacene la información de la estación que considere pertinente.

## Requerimientos - Parte A (estudiante 1 de cada grupo)

1. **Obtener el camino de costo mínimo entre dos ubicaciones geográficas por distancia**

Para encontrar el camino de costo mínimo se debe tomar la **distancia haversiana** en cada arco como medida base. El punto de origen y destino son ingresados por el usuario como latitudes y longitudes (debe validarse que dichos puntos se encuentren dentro de los límites encontrados de la ciudad). Estas ubicaciones deben aproximarse a los vértices más cercanos en la malla vial.

**Respuesta en consola:** Muestre en la consola de texto el camino a seguir informando: el total de vértices, sus vértices (Id, latitud, longitud), el costo mínimo (menor distancia haversiana) y la distancia estimada (sumatoria de distancias haversianas en Km).

**Visualización mapa:** Muestre el camino resultante en Google Maps (incluyendo la ubicación de inicio y la ubicación de destino).

2. **Determinar la red de comunicaciones que soporte la instalación de cámaras de video en los  $M$  puntos donde se presentan los comparendos de mayor gravedad.**  
Se debe ingresar el número  $M$  de comparendos que se requieren.

El distrito quiere instalar una red de comunicaciones que le permita la instalación de cámaras de video en  $M$  sitios; sin embargo, se requiere que esta red tenga el **menor**

**costo de instalación posible.** El costo de instalación de la red es de U\$10000 por cada kilómetro extendido.

Con la finalidad de que la red sea eficiente se seleccionaron como puntos de supervisión los  $M$  vértices donde se presentan los comparendos de mayor gravedad. Para saber si un comparendo es más grave que otro primero se mira el tipo de servicio: Público es más grave que Oficial y Oficial es más grave que Particular; si dos comparendos tienen el mismo tipo de servicio se compara el código de la infracción (campo INFRACCION) usando el orden lexicográfico (forma de comparación de los Strings en Java, A12 es más grave que A11 y B10 es más grave que A10).

**Respuesta en consola:** Muestre en la consola de texto el tiempo que toma el algoritmo en encontrar la solución, y la siguiente información de la red propuesta: los vértices (identificadores) y los arcos incluidos, y el costo (monetario) total.

**Visualización mapa:** Muestre en un mapa en Google Maps la red de comunicaciones propuesta. Resalte las  $M$  ubicaciones de las cámaras y los arcos de la red que las unen.

## Requerimientos - Parte B (estudiante 2 de cada grupo)

### 1. Obtener el camino de costo mínimo entre dos ubicaciones geográficas por número de comparendos

Para encontrar el camino de costo mínimo se debe tomar la cantidad de **comparendos** en cada arco como medida base. El punto de origen y destino son ingresados por el usuario como latitudes y longitudes (debe validarse que dichos puntos se encuentren dentro de los límites encontrados de la ciudad). Estas ubicaciones deben aproximarse a los vértices más cercanos en la malla vial.

**Respuesta en consola:** Muestre en la consola de texto el camino a seguir, informando el total de vértices, sus vértices (Id, latitud, longitud), el costo mínimo (menor cantidad de comparendos) y la distancia estimada (sumatoria de distancias harvesianas en Km).

**Visualización mapa:** Muestre el camino resultante en Google Maps (incluyendo la ubicación de inicio y la ubicación de destino).

### 2. Determinar la red de comunicaciones que soporte la instalación de cámaras de video en los $M$ puntos donde se presenta el mayor número de comparendos en la ciudad. Se debe ingresar el número $M$ de vértices que se requieren.

El distrito quiere instalar una red de comunicaciones que le permita la instalación de cámaras de video en  $M$  sitios; sin embargo, se requiere que esta red tenga **el menor costo de instalación posible**. El costo de instalación de la red es de U\$10000 por cada

kilómetro extendido. Con la finalidad de que la red sea eficiente se seleccionaron como puntos de supervisión los M vértices donde se presentan el mayor número de comparendos en la ciudad.

**Respuesta en consola:** Muestre en la consola de texto el tiempo que toma el algoritmo en encontrar la solución (en milisegundos), y la siguiente información de la red propuesta: el total de vértices en el componente, los vértices (identificadores), los arcos incluidos (Id vértice inicial e Id vértice final) y el costo (monetario) total.

**Visualización mapa:** Muestre en un mapa en Google Maps la red de comunicaciones propuesta. Resalte las M ubicaciones de las cámaras y los arcos de la red que las unen.

## Parte C (trabajo en grupo)

### 1. Obtener los caminos más cortos para que los policías puedan atender los M comparendos más graves.

Se debe ingresar el número M de comparendos que se quieren atender.

Se espera que cada ubicación de un comparendo grave debe ser atendida por la estación de policía más cercana (mínima distancia de desplazamiento usando la malla vial). Asuma que cualquier policía en una estación puede atender un comparendo y que inicialmente todos los policías están en sus estaciones.

**Respuesta en consola:** Muestre en la consola de texto el tiempo que toma el algoritmo en encontrar la solución (en milisegundos) y los caminos resultantes para cada uno de los M comparendos: su secuencia de vértices y arcos, así como su costo total (sumatoria de distancia de los arcos en kilómetros).

**Visualización mapa:** Muestre los caminos resultantes en Google Maps diferenciando la ubicación del comparendo (origen) y la ubicación de las estaciones de policía (destinos). Asigne un color diferente para graficar las rutas más eficientes de cada uno de los M comparendos.

### 2. Identificar las zonas de impacto de las estaciones de policía.

Se deben ingresar todos los comparendos

Para poder identificar las zonas de impacto de cada una de las estaciones de policía, se debe **asignar** la estación de policía que puede atender cada comparendo en el menor tiempo posible; es decir, la estación que tenga la ruta más corta (mínima distancia de desplazamiento usando la malla vial) al lugar del comparendo.

Tomando como base los caminos identificados anteriormente (entre cada comparendo y la estación de policía que lo atiende) cree un **Grafo** tomando únicamente los vértices y arcos involucrados en dichos caminos. Defina un esquema JSON para persistir su grafo.

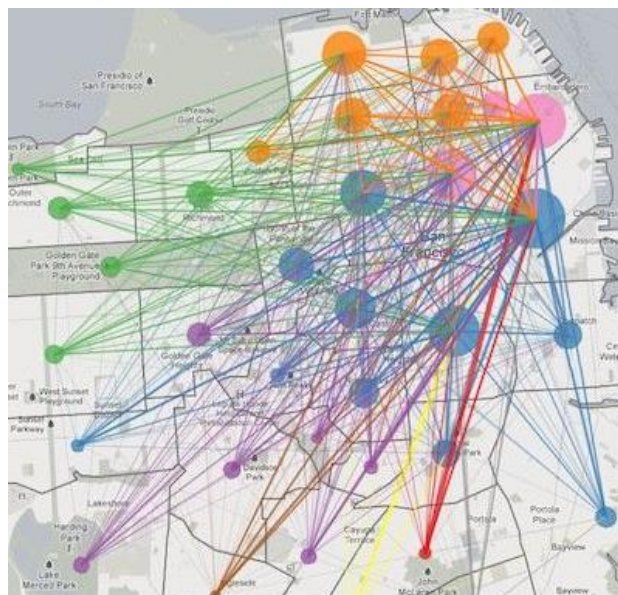
A continuación, calcule los componentes conexos presentes en el grafo no dirigido construido. Asígnele un color a cada uno de los componentes identificados y asígneles dicho color a todos los vértices de cada componente. Cada componente debe tener un color diferente.

**Respuesta en consola:** Muestre en la consola de texto:

- El tiempo que toma el algoritmo en encontrar la solución (en milisegundos),
- El número de vértices y arcos que tiene el grafo no dirigido, recuerde que cada arco es bidireccional,
- La cantidad de comparendos que atiende cada una de las estaciones de policía,
- Para cada componente conexa imprima: el color, el ObjectId de las estaciones de policía dentro del componente, y el número de vértices incluidos.

**Visualización mapa:** A partir del grafo construido pinte sobre el mapa de la red vial de Bogotá utilizando Google Maps:

- Un círculo en la posición de cada estación de policía. El área del círculo es proporcional al porcentaje de comparendos atendidos por dicha estación. El color del círculo es el mismo que el del componente conexo a la que pertenece.
- Genere arcos del color del componente conexo al cual pertenece el vértice del grafo donde se producen los M primeros comparendos de cada estación.





## Restricciones

- Los datos contenidos en los archivos sólo se pueden leer una vez
- Se deberá trabajar en Java 8
- El proyecto se debe implementar en Eclipse
- La entrada/salida de información adicionales se debe realizar por consola
- **No usar las colecciones del API Java.**

## Entrega de Diseño (33% Nota del Proyecto)

- Repositorio GitHub con el nombre de la forma Proyecto\_3\_202010\_sec\_Y\_team\_Z (reemplazar Y por el número de la sección de su curso y Z su número de grupo)
- Verificar que el repositorio tiene configurado como usuarios a los monitores y al profesor con acceso de lectura.
- Entregables:
  - Documento con:
    - Documentación de los requerimientos funcionales (incluyendo la descripción, los datos de entrada, los datos de salida y estimación de complejidad temporal de cada requerimiento funcional). Por cada requerimiento hay que precisar la(s) estructura(s) de datos a utilizar.
    - Diseño de las Estructuras de Datos a utilizar (diagrama de clases UML, imagen)
    - Diseño de la Solución al proyecto usando una arquitectura MVC y haciendo explícito el uso de las Estructuras de Datos (diagrama de clases UML, imagen)
  - Proyecto Eclipse/Java con:
    - Implementación y Pruebas Unitarias Automáticas de las Estructuras de Datos a utilizar en proyecto Eclipse/Java.
    - Lectura/carga de los datos, presentar el reporte de los resultados descritos en la sección Carga de Información.
    - Implementación de los Requerimientos - Parte Inicial
- Hacer el desarrollo del proyecto en la rama (*branch*) *master* del repositorio. Como parte final de esta entrega de Diseño, crear la rama (*branch*) entrega-diseNo.
  - Verificar que en la cuenta GitHub de entrega del proyecto la rama entrega-DiseNo queda con la copia de los entregables de Diseño del proyecto.
  - Los entregables en la rama entrega-diseNo **NO** deben actualizarse después de la fecha/hora límite de esta entrega.

## Entrega Final (67% Nota del Proyecto)

- Repositorio GitHub Proyecto\_3\_202010\_sec\_Y\_team\_Z (reemplazar Y por el número de la sección de su curso y Z su número de grupo) creado para la entrega de diseño. El proyecto completo (documentación e implementación) debe estar accesible en la rama (*branch*) *master*.
- Entregables:

- Proyecto Eclipse/Java con:
    - Implementación completa de los requerimientos funcionales
- Los entregables en la rama master No pueden tener fecha de actualización posterior a la fecha/hora límite de esta entrega.