|  |  |
| --- | --- |
| Nombre | R0 – Cargar grafo no definido de la ciudad completo |
| Resumen | Carga el grafo no dirigido con la información de la ciudad completa |
| Entradas | |
| N/A | |
| Resultados | |
| * Informa el total de vértices y arcos que definen al grafo cargado | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R1 – Agregar la información de costo al grafo |
| Resumen | Agrega los 3 costos en cada arco del grafo |
| Entradas | |
| N/A | |
| Resultados | |
| * Se agregan los 3 tipos de costos a cada arco del grafo. El primero viene dado por la distancia entre sus vértices, el segundo por el tiempo promedio entre sus vértices y el tercero por la velocidad, es decir, la distancia sobre el tiempo entre sus vértices. | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R3 – Crear archivo JSON a partir del grafo |
| Resumen | Carga el grafo no dirigido con la información de la ciudad completa |
| Entradas | |
| N/A | |
| Resultados | |
| * Crea un archivo JSON que contiene al grafo completo | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R4 – Cargar archivo JSON que contiene el grafo |
| Resumen | Carga el grafo no dirigido a partir de un archivo JSON |
| Entradas | |
| N/A | |
| Resultados | |
| * Se carga el grafo completo a partir de un archivo JSON | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R5 – Encontrar el camino de costo mínimo para un viaje entre dos localizaciones de la ciudad |
| Resumen | Encuentra el camino más corto entre dos localizaciones geográficas de la ciudad ingresadas por el usuario |
| Entradas | |
| Latitud y longitud de origen, latitud y longitud de destino | |
| Resultados | |
| * En consola se debe mostrar el camino a seguir, informando el total de vértices, sus vértices (Id, Longitud, Latitud), el costo mínimo, y la distancia estimada * Muestre el camino resultante en Google Maps (Incluyendo ubicación de inicio y de destino) | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R6 – Determinar n vértices con menor velocidad promedio en la ciudad de Bogotá |
| Resumen | Encuentra los n vértices que tienen la menor velocidad promedio en la ciudad de Bogotá, siendo esta velocidad el promedio de las velocidades en todos sus arcos |
| Entradas | |
| Número de vértices a determinar | |
| Resultados | |
| * En consola se debe mostrar los n vértices encontrados (Id, latitud y longitud) ordenados de mayor a menor por su velocidad promedio del vértice. Adicionalmente, debe informar el número de componentes conectados que se definen entre estos vértices en el grafo original. Por cada componente se deben informar los identificadores de los vértices que lo conforman. * Muestre en Google Maps los n vértices resultantes usando un color 1. Destaque la componente conectada más grande (ósea, con más vértices) usando un color 2. Para esta componente muestre sus vértices y sus arcos. | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R7 – Calcular MST con criterio distancia aplicado al componente conectado más grande de la malla vial de Bogotá. |
| Resumen | Encuentra el camino más corto entre dos localizaciones geográficas de la ciudad ingresadas por el usuario. |
| Entradas | |
| N/A | |
| Resultados | |
| * En consola se debe mostrar el tiempo (en ms) que se demora el algoritmo en encontrar la solución y la siguiente información del árbol generado: el total de vértices en el componente, los vértices (identificadores), los arcos incluidos(Id vértice inicial e Id vértice final) y el costo total (distancia en KM) del árbol. * Muestre el árbol generado resultante en Google Maps: sus vértices y sus arcos. | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R8 – Encontrar el camino de menor costo (menor distancia Haversine) para un viaje entre dos localizaciones geográficas |
| Resumen | Encuentra el camino más corto por distancias haversine,entre dos localizaciones geográficas de la ciudad ingresadas por el usuario. |
| Entradas | |
| Latitud y longitud de las dos localizaciones | |
| Resultados | |
| * En consola: Se muestra el camino a seguir, informando el total de vértices, sus vértices (Id, latitud, longitud), el tiempo estimado (la sumatoria de los tiempos de sus arcos) y la distancia Haversine estimada (sumatoria de distancias Haversine en Km). * Se muestra el árbol generado resultante en Google Maps: sus vértices y sus arcos. | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R9 – Indicar que coordendas de una localizacion son alcanzables en un tiempo determinado |
| Resumen | Encontrar todas las coordendas cuya distancia(tiempo) este entre los parametros ingresados |
| Entradas | |
| Latitud y longitud de la localizacion, tiempo limite | |
| Resultados | |
| * En consola: Se muestra los identificadores y la ubicación (lat, lon) de los vértices alcanzables en un tiempo T a partir de la localización de origen * Se marca la localización de origen en un color 1 y las localizaciones de los vértices alcanzables en un color 2 en Google Maps | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R10 – Indicar que coordendas de una localizacion son alcanzables en un tiempo determinado |
| Resumen | Encontrar todas las coordendas cuya distancia(tiempo) este entre los parametros ingresados |
| Entradas | |
| Latitud y longitud de la localizacion, tiempo limite | |
| Resultados | |
| * En consola: Se muestra los identificadores y la ubicación (lat, lon) de los vértices alcanzables en un tiempo T a partir de la localización de origen * Se marca la localización de origen en un color 1 y las localizaciones de los vértices alcanzables en un color 2 en Google Maps | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R11 – Dar MST usando algoritmo de Kruskal |
| Resumen | Calcular un árbol de expansión mínima (MST) con criterio distancia, utilizando el algoritmo de Kruskal, aplicado al componente conectado (subgrafo) más grande de la malla vial de Bogotá |
| Entradas | |
| Latitud y longitud de la localizacion, tiempo limite | |
| Resultados | |
| * En consola: Se muestra el tiempo que toma el algoritmo en encontrar la solución (en milisegundos), y la siguiente información del árbol generado: el total de vértices en la componente, los vértices (identificadores), los arcos incluidos (Id vértice inicial e Id vértice final) y el costo total (distancia en Km) del árbol   -Se muestra el árbol generado resultante en Google Maps: sus vértices y sus arcos. | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R12 – Construir un nuevo grafo simplificado No dirigido de las zonas Uber |
| Resumen | Genera un arbol que muestra la vecindad, representada por arcos, de las Zonas Uber |
| Entradas | |
| n/a | |
| Resultados | |
| * En consola: Al final de la construcción del grafo de zonas, se reporta la cantidad de vértices y arcos (cada arco debe contarse una única vez).   -Se muestra el grafo generado resultante en Google Maps: sus vértices y sus arcos. | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |

|  |  |
| --- | --- |
| Nombre | R13 – Calcular el camino de costo mínimo basado en el tiempo promedio entre una zona promedio entre una zona de origen y zona de destino sobre el grafo de zonas |
| Resumen | Calcula el camino de costo mínimo basado en el tiempo promedio entre una zona promedio entre una zona de origen y zona de destino sobre el grafo de zonas utilizando el algoritmo de Dijkstra |
| Entradas | |
| N/A | |
| Resultados | |
| * En consola: Se muestra el tiempo en milisegundos que se demoró el algoritmo en encontrar la solución. Del camino resultante se debe encontrar su secuencia de vértices/zonas y su costo total. Adicionalmente, se debe mostrar el tiempo promedio desde la zona de origen hasta la zona de destino teniendo en cuenta los días de la semana. * -Se muestra el grafo generado resultante en Google Maps, ósea, el camino de costo mínimo: sus vértices y sus arcos. | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa y el uso del algoritmo de Dijkstra. | |

|  |  |
| --- | --- |
| Nombre | R14 – Calcular los caminos de menor longitud de una zona de origen a todas sus zonas alcanzables |
| Resumen | A partir de una zona de origen, calcula los caminos de menor longitud a todas sus zonas alcanzables. Se debe seleccionar el camino más largo, que será el camino hasta la zona más distante |
| Entradas | |
| Zona de origen x | |
| Resultados | |
| * En consola: Se muestra el tiempo en milisegundos que se demoró el algoritmo en encontrar la solución. Del camino resultante se debe encontrar su secuencia de vértices/zonas y su número total de arcos. Si hay múltiples caminos con la longitud máxima, mostrar aquel que llegue al menor movement ID de destino. * -Se muestra el grafo generado resultante en Google Maps, ósea, el camino más largo: sus vértices y sus arcos. | |
| Complejidad temporal | |
| O(n) | |
| Estructura Escogida | |
| Se debe usar el grafo puesto que esta estructura es la que permite el modelado del mapa | |