Tecnológico de Monterrey

Análisis y diseño de algoritmos avanzados

Reporte actividad integradora

Profesor:

Ramona Fuentes Valdéz

Alan Patricio González Bernal | A01067546
Alan Rodrigo Castillo Sánchez |A01708668

Índice

Problemática

Algoritmos usados

PRIM

El algoritmo de prim es un algoritmo que busca encontrar el *minimum spanning tree* o "árbol recubridor mínimo" de un grafo ponderado no dirigido, es decir que busca encontrar la forma de conectar todos los nodos o vértices de un grafo ocupando la menor cantidad posible de "peso" o "costo" dentro del mismo y evitando ciclos en las conexiones.

Este algoritmo nos permite encontrar la manera más "barata" del grafo, en términos de kilómetros, para conectar todas las colonias usando la menor cantidad de fibra óptica y permitiendo conectar una red por completo entre todas las colonias facilitando la comunicación de entre cualesquiera dos colonias. Este algoritmo tiene una complejidad computacional de O(VE)

TSP

La problemática del *Traveler Salesman Problem* supone sobre un viajero que busca la ruta más corta posible que visite todas las ciudades objetivo exactamente una vez y regrese a la ciudad de partida. Y si bien no existe una respuesta "correcta" ni óptima, si existen diferentes algoritmos con distintos enfoques que buscan resolver este problema, y el que aplicamos en concreto es uno de enfoque del "vecino más cercano", que consiste en buscar la ciudad más cercana no visitada en cada paso y agregarla al recorrido. Este proceso se repite hasta que todas las ciudades hayan sido visitadas, y luego el viajero regresa a la ciudad de partida.

Este algoritmo cuenta con una complejidad computacional de O(N^2)

Ford-Fulkerson

Este algoritmo nos permite encontrar el flujo máximo en una red, este lo que hace es buscar un camino desde el origen hasta el destino a lo largo de las aristas donde el flujo actual no ha alcanzado la capacidad máxima. Después se aumenta el flujo a lo largo del camino humectante hasta que esto ya no sea posible.

Estos pasos se repiten hasta que ya no haya caminos humectantes y el flujo máximo es la suma de todos los flujos que entran al nodo destino. Su complejidad computacional es de O(VE^2).

Distancias Euclidianas

Para encontrar la distancia más cercana de la central que se nos brinda, decidimos utilizar un algoritmo de distancias euclidianas. Estas son sacadas mediante esta fórmula:

$$\sqrt{\left(x_2 - x_1\right)^2 + \left(y_2 - y_1\right)^2}$$

Cada distancia se da en sus coordenadas XY. Gracias a la fórmula euclidiana, podemos definir las distancias entre dos puntos de forma bastante precisa y su complejidad computacional es de O(N).

Conclusión

Esta actividad nos permitió entender cómo podemos utilizar los algoritmos y funciones vistas en clase en situaciones reales, además de esto, nos permitió poder mejorar nuestras aptitudes de trabajo en equipo y nuestras habilidades de programación. Estamos contentos con el resultado y esperamos poder seguir utilizando estos y más algoritmos con el objetivo de dominarlos en su totalidad.