OBSERVACIONES DEL LA PRACTICA

Juan Manuel Pérez - 202021827

Nicolas Camargo - 202020782

Preguntas de análisis

a) ¿Qué instrucción se usa para cambiar el límite de recursión de Python?

= Se utiliza la librería sys con la instrucción sys.setrecursionlimit(n), donde n representa el nuevo valor limite a establecer. Por otro lado, en el view en la función 'OptionTwo' esta una instrucción similar que nos da el límite de recursión actual, la cual es: sys.getrecursionlimit().

b) ¿Por qué considera que se debe hacer este cambio?

= Debido a que, haciendo este cambio se establece un límite apropiado en la recursividad del programa, por lo tanto, gracias a este límite de recursión se evita una recursión infinita que generaría que python crasheara. Sin embargo, se debe tener en cuenta que el límite depende de la capacidad del sistema en el que se ejecute el programa para soportar mayor límite de recursividad y de si se desea ir más a fondo en la recursividad.

c) ¿Cuál es el valor inicial que tiene Python cómo límite de recursión?

= El valor inicial de python en límite de recursión es de 1000.

d) ¿Qué relación creen que existe entre el número de vértices, arcos y el tiempo que toma la operación 4?

= Como muestra la siguiente tabla que representa la relación entre el número de vértices, arcos, el tiempo de ejecución y los diferentes archivos, se evidencia que a mayor número de datos existe un mayor número de vértices y de arcos, y asimismo de tiempo, por tanto, se puede afirmar según los resultados obtenidos que es una relación directamente proporcional, pues a mayor número de vértices y de arcos, mayor tiempo de ejecución.

	Numero Vértices	Numero Arcos	Tiempo
Bus_routes_50.csv	**74**	**73**	**0.1125786304473877**
Bus_routes_150.csv	**146**	**146**	**0.1083395481109619**
Bus_routes_300.csv	**295**	**382**	**0.3511528968811035**
Bus_routes_1000.csv	**984**	**1633**	**1.6838405132293701**
Bus_routes_2000.csv	**1954**	**3560**	**4.825983762741089**
Bus_routes_3000.csv	**2922**	**5773**	**9.182554960250854**
Bus_routes_7000.csv	**6829**	**15334**	**25.709922552108765**
Bus_routes_10000.csv	**9767**	**22758**	**74.26689791679382**

Bus_routes_14000.csv	**13535**	**32270**	**199.43931698799133**

e) ¿El grafo definido es denso o disperso?, ¿El grafo es dirigido o no dirigido?, ¿El grafo está fuertemente conectado?

= Se trata de un grafo denso por el número de arcos que se acerca al máximo, y dirigido pues es la implementación que se importa desde ADT.graph al estar en True. Además, está fuertemente conectado debido a que como indica la función AddRouteConnections, por cada vértice se crean arcos entre ellas para representar el cambio de ruta.

f) ¿Cuál es el tamaño inicial del grafo?

= El tamaño inicial del grafo es de 14000. Aunque si no se asigna tamaño, este seria de 10 por defecto, según la función newGraph en ADT/graph.py.

g) ¿Cuál es la Estructura de datos utilizada?

= La estructura de datos utilizada es 'ADJ_LIST', es decir, una lista de adyacencia, que es la representación de los arcos del grafo en una lista.

h) ¿Cuál es la función de comparación utilizada?

= La función de comparación utilizada se llama 'compareStopIds', que compara el valor de las estaciones.