**OBSERVACIONES DEL LA PRÁCTICA**

# Juana María Morales Ramos - 202021591

# Mateo López Tobón - 202021416

# **Preguntas de análisis**

1. **¿Qué instrucción se usa para cambiar el límite de recursión de Python?**

Es posible modificar el tamaño máximo (límite de recursión) de la pila de recursión mediante la instrucción **sys.setrecursionlimit(n)**, la cual en el ejercicio esta ubicado en la vista.

1. **¿Por qué considera que se debe hacer este cambio?**

Porque en Python, para evitar que la memoria se termine, la pila de ejecución de funciones tiene un límite. De lo contrario, como cada llamada recursiva agrega un elemento a la pila de llamadas a funciones y la memoria de nuestras computadoras no es infinita, el ciclo deberá terminarse cuando se agote la memoria disponible.

En este caso, se podría considerar realizar un cambio, ya que para el ejercicio el limite de recursión actual es de 1048576 y se debe tener en cuenta que en **python** el **límite** máximo de llamadas de recursividad es de 1000.

1. **¿Cuál es el valor inicial que tiene Python cómo límite de recursión?**

En python el límite máximo de llamadas de recursividad es de 1000.

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre del archivo** | **Número de vértices** | **Número de arcos** | **Tiempo de ejecución (mseg)** |
| bus\_routes\_50.csv | 74 | 73 | 45.495 |
| bus\_routes\_150.csv | 146 | 146 | 69.280 |
| bus\_routes\_300.csv | 295 | 382 | 125.488 |
| bus\_routes\_1000.csv | 984 | 1633 | 533.761 |
| bus\_routes\_2000.csv | 1954 | 3560 | 2307.380 |
| bus\_routes\_3000.csv | 2922 | 5773 | 4091.510 |
| bus\_routes\_7000.csv | 6829 | 15334 | 13019.950 |
| bus\_routes\_10000.csv | 9767 | 22758 | 37195.500 |
| bus\_routes\_14000.csv | 13535 | 32270 | 57956.590 |

La relación observada es que medida de los archivos de datos son más pequeños, es decir, disminuyen su contenido, asi mismo se observa un decrecimiento en la cantidad de vértices (nodo) y de arcos (pares ordenados de un conjunto de vértices), por consiguiente esto también demuestra que se necesita un menor de tiempo de ejecución. Por otro lado, por obvias razones se observa que hay menor cantidad de vértices y mayor cantidad de arcos, ya que estos últimos al ser las posibles relaciones que se pueden establecer entre los nodos, pueden ser multiples.

1. **¿El grafo definido es denso o disperso?, ¿El grafo es dirigido o no dirigido?, ¿El grafo está fuertemente conectado?**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre del archivo** | **Número de vértices** | **Número de arcos** | **Tiempo de ejecución (mseg)** |
| bus\_routes\_50.csv | 74 | 73 | 0.7159 |
| bus\_routes\_150.csv | 146 | 146 | 1.8590 |
| bus\_routes\_300.csv | 295 | 382 | 1.0440 |
| bus\_routes\_1000.csv | 984 | 1633 | 1.6350 |
| bus\_routes\_2000.csv | 1954 | 3560 | 1.9219 |
| bus\_routes\_3000.csv | 2922 | 5773 | 1.2629 |
| bus\_routes\_7000.csv | 6829 | 15334 | 6.4549 |
| bus\_routes\_10000.csv | 9767 | 22758 | 2.0659 |
| bus\_routes\_14000.csv | 13535 | 32270 | 11.727 |

El grafo es dirigido, ya que las rutas tienen una dirección específica entre las estaciones.

El grafo es disperso, y no esta fuertemente conectado.

1. **¿Cuál es el tamaño inicial del grafo?**

El tamaño inicial configurado del grafo es de 14000.

1. **¿Cuál es la Estructura de datos utilizada?**

Según el model.py, vemos que en el newAnalyzer se configura como datastructure adjlist.py, la cual contiene la librería de todas las funciones necesarias en grafos.

1. **¿Cuál es la función de comparación utilizada?**

En el model hay dos funciones de comparación. Una de ellas es “compareroutes”, la cual compara dos rutas y se utiliza en una de las funciones para agregar información al grafo llamada addRouteStop, esta última, agrega a una estacion, una ruta que es servida en ese paradero. Y la segunda función de comparación es “compareStopsIds”, la cual compara dos estaciones y se utiliza en la parte superior del model.py en el newAnalyzer.