**OBSERVACIONES DEL LA PRACTICA**

David Leonardo Almanza - 202011293

Laura Daniela Arias Flórez - 202020621

# **Preguntas de análisis**

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

Se usa la última instrucción que sale en el archivo del view. El que es con un condicional if.

if \_\_name\_\_ == "\_\_main\_\_":

    threading.stack\_size(67108864)  # 64MB stack

    sys.setrecursionlimit(2 \*\* 20)

    thread = threading.Thread(target=thread\_cycle)

    thread.start()

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

Porque por la forma en la que se construyen los grafos, si no hiciéramos este cambio, Python no permitiría crear la estructura de datos, ya que sobrepasaría el límite inicial de recursión.

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

El valor inicial que tiene Python como límite de recursión es de 1000.

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

14.000 datos: 31.39 segs

10.000 datos: 13.29 segs

7.000 datos: 6.58 segs

3.000 datos: 2.31 segs

Es bastante claro que a medida que crece el numero de datos, y por lo tanto el número de vértices y arcos, así mismo lo hace el tiempo de ejecución de la operación. Parece crecer de forma cuadrática.

1. ¿Qué características tiene el grafo definido?

El grafo definido es un grafo dirigido, además arcos tienen peso.

1. ¿Cuál es el tamaño inicial del grafo?

El tamaño inicial del grafo es de 14.000 elementos

1. ¿Cuál es la Estructura de datos utilizada?

Para las paradas se usan mapas y para las conexiones se usa un grafo dirigido con arcos con peso.

1. ¿Cuál es la función de comparación utilizada?

La funcion de comparacion es la de compareStopIds, esta ordena de modo descendente tanto los nombres de las paradas como los nombres de las conecciones, que estas representadas mediante el codigo de la estacion y el nombre de la ruta, separadas por un guion.