**OBSERVACIONES DEL LA PRACTICA**

Jairo Nicolás Gómez Cod 202020414

Santiago Andrés Vela Cod 202026767

bus\_routes\_14000.csv

Numero de vértices: 13535

Numero de arcos: 32270

El límite de recursión actual: 1048576

Tiempo [ms]: 5998.539

Tiempo Opción 4 - Tiempo [ms]: 132573.498

bus\_routes\_10000.csv

Numero de vértices: 9767

Numero de arcos: 22758

El límite de recursión actual: 1048576

Tiempo [ms]: 4387.457

Tiempo Opción 4 - Tiempo [ms]: 42561.165

bus\_routes\_7000.csv

Numero de vértices: 6829

Numero de arcos: 15334

El límite de recursión actual: 1048576

Tiempo [ms]: 3506.393

Tiempo Opción 4 - Tiempo [ms]: 18895.592

bus\_routes\_3000.csv

Numero de vértices: 2922

Numero de arcos: 5773

El límite de recursión actual: 1048576

Tiempo [ms]: 1315.431

Tiempo Opción 4 - Tiempo [ms]: 4600.504

bus\_routes\_2000.csv

Numero de vértices: 1954

Numero de arcos: 3560

El límite de recursión actual: 1048576

Tiempo [ms]: 1209.473

Tiempo Opción 4 - Tiempo [ms]: 2475.709

bus\_routes\_1000.csv

Numero de vértices: 984

Numero de arcos: 1633

El límite de recursión actual: 1048576

Tiempo [ms]: 738.459

Tiempo Opción 4 - Tiempo [ms]: 676.995

bus\_routes\_300.csv

Numero de vértices: 295

Numero de arcos: 382

El límite de recursión actual: 1048576

Tiempo [ms]: 335.159

Tiempo Opción 4 - Tiempo [ms]: 245.385

bus\_routes\_150.csv

Numero de vértices: 146

Numero de arcos: 146

El límite de recursión actual: 1048576

Tiempo [ms]: 183.962

Tiempo Opción 4 - Tiempo [ms]: 235.919

bus\_routes\_50.csv

Numero de vértices: 74

Numero de arcos: 73

El límite de recursión actual: 1048576

Tiempo [ms]: 173.020

Tiempo Opción 4 - Tiempo [ms]: 116.011

Punto 6

bus\_routes\_14000.csv

Numero de vértices: 136

El número de arcos es: 136

Tiempo [ms]: 17.692

bus\_routes\_10000.csv

Numero de vértices: 112

El número de arcos es: 112

Tiempo [ms]: 12.857

bus\_routes\_7000.csv

Numero de vértices: 95

El número de arcos es: 95

Tiempo [ms]: 9.244

bus\_routes\_3000.csv

Numero de vértices: 81

El número de arcos es: 81

Tiempo [ms]: 5.125

bus\_routes\_2000.csv

bus\_routes\_1000.csv

bus\_routes\_300.csv

bus\_routes\_150.csv

bus\_routes\_50.csv

# **Preguntas de análisis**

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

Se utiliza setrecursionlimit y se eleva el 2 \*\*20 para cambiar el limite de recursión a ese valor

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

Consideramos que se necesita cambiar el limite de recursión porque los algoritmos son recursivos entonces al procesar tanta información va llegar un punto que el valor inicial de recursión no va a alcanzar por lo que hay que aumentarlo. Eso debido a que la funcion se va a llamar muchas veces cuando se este ejecuntando un algoritmo recursivo, entonces por eso hay que ampliarlo.

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

El valor inicial de limite de recursión en python es 1000

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

La relacion que existe entre el numero de vertices y arcos es: Primero, entre archivos claramente se disminuye el numero de vertices y arcos. Segundo, la relación entre vertices y arcos en la mayoria de archivos es que existe una mayor cantidad de arcos que vertices. En algunos casos son el doble y hasta el triple de arcos. El tiempo que toma en ejecutar la opcion 4 se disminuye cuando se disminuye la cantidad de archivos

1. ¿Qué características tiene el grafo definido?

Un grafo definido tiene un tamaño que es el numero de vertices. Tambien tiene una estructura de datos el cual puede ser una matriz o lista de adyancecnias. Se le indica si es dirigida o no. Finalmente tiene una función de compración.

1. ¿Cuál es el tamaño inicial del grafo?

El tamaño inicial del grafo es 14000, es decir, el numero de vertices

1. ¿Cuál es la Estructura de datos utilizada?

La estructura de datos utilizada es un ADJ\_LIST, es decir una lista de adyacencias

1. ¿Cuál es la función de comparación utilizada?

La función de comparacion utilizada es la compareStopsIds que compara los ids de las paradas