**Grupo 12 Sección 07**

**Juan Andres Bernal – 202110848**

**Juan Esteban López – 202021417**

**Análisis laboratorio 9**

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

Se utiliza la librería de Python sys, y para obtener el valor del limite de recursiones por medio del método getrecursionlimit(), y de acuerdo a como avance el programa, este irá cambiando y estableciendo cierto limite.

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

Se debe hacer este cambio porque depende de cuanta memoria haya disponible para realizar la ejecución del programa en el momento, debido a que si se excede, el programa se deja de ejecutar y esto conlleva a que se use el método setrecursionlimit() para establecer un límite de recursión fijo y el computador no colapse en caso de exceder cierto limite.

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

Tras las pruebas realizadas en Visual Studio Code en Python 3.7.7 e investigación por Internet, concluimos que el valor inicial del límite de recursión de Python es de valor 1000.

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

A mayor número de vértices, obtendremos un número de arcos mayor, teniendo en cuenta esto, notamos que el número de arcos depende del número de vértices, el número de arcos posee un crecimiento exponencial, por lo tanto, se trabajará sobre más datos, por lo que el tiempo de ejecución aumentaría dependiendo de su complejidad y de las condiciones de ejecución de las pruebas

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

Consideramos que el grafo es denso, debido a que el número de arcos es notablemente superior al número de vértices, por lo que afirmaríamos que a cada vértice le corresponde más de un arco.

El grafo es dirigido, debido a que las estaciones pueden llegar a tener más de un arco, y consideramos que este un posee un orden dependiendo del número de la estación y ubicación de esta.

Teniendo en cuenta que un grafo fuertemente conectado, es un grado donde para cada par de vértices u y v existe un camino de u hacia v y de v hacia u. Por lo que teniendo en cuenta que este grafo posee datos de estaciones de buses, consideramos que existe una forma de cada estación llega a cualquier otra y viceversa, por lo que consideramos que el grafo si está fuertemente conectado.

f. ¿Cuál es el tamaño inicial del grafo?

El grafo tiene un tamaño inicial de 14000

g. ¿Cuál es la Estructura de datos utilizada?

Para la construcción del grafo la estructura de datos implementada es lista de adyacencia.

h. ¿Cuál es la función de comparación utilizada?

La función compareStopIds, que se encarga de ordenar el grafo de acuerdo a menor o mayor StopId.

**Graficas de comparación Vertices, Arcos y Tiempo.**

|  |  |  |  |
| --- | --- | --- | --- |
| PRUEBAS Funcion # 4 | | | |
| Tamaño | # Vertices | # Arcos | Tiempo ejecucion |
| 50 | 74 | 73 | 15,625 |
| 150 | 146 | 146 | 15,625 |
| 300 | 295 | 382 | 46,875 |
| 1000 | 984 | 1633 | 218,75 |
| 2000 | 1954 | 3560 | 718 |
| 3000 | 2925 | 5773 | 1328,125 |
| 7000 | 6829 | 15334 | 3500 |
| 10000 | 9767 | 22758 | 10453 |
| 14000 | 13535 | 32270 | 18546 |

|  |  |  |  |
| --- | --- | --- | --- |
| PRUEBAS Funcion # 6 | | | |
| Tamaño | # Vertices | # Arcos | Tiempo ejecucion |
| 50 | 59 | 58 | 0 |
| 150 | 59 | 58 | 0 |
| 300 | 65 | 64 | 0 |
| 1000 | 63 | 62 | 0 |
| 2000 | 80 | 79 | 0 |
| 3000 | 81 | 80 | 15,625 |
| 7000 | 95 | 94 | 15,625 |
| 10000 | 112 | 111 | 0 |
| 14000 | 136 | 135 | 0 |
|  |  |  |  |