

## Preguntas Lab 8

Nikolas Santacruz Mejia [n.santacruz@uniandes.edu.co](mailto:n.santacruz@uniandes.edu.co) 201821946

Juan David Quintero [jd.quinterog@uniandes.edu.co](mailto:jd.quinterog@uniandes.edu.co) 201910905

**Pregunta 1:** ¿Qué características tiene el grafo definido?, ¿Tamaño inicial, es dirigido?, ¿Estructura de datos utilizada?

El grafo es dirigido y cada arco representa la posibilidad de ir desde una parada con un bus en específico hacia otra parada que también está relacionada con el numero de servicio de algún vehículo. El tamaño del grafo estará determinado por el número de buses y sus correlaciones con cada parada, porque esto es lo que determinara la cantidad de nodos y por tanto su tamaño. Como se está utilizando una lista de adyacencias, la estructura de datos ocupara poco espacio si el grafo no es tan denso, es decir, que cada bus no puede llegar a muchas estaciones porque si es así habrán muchas relaciones expresadas por los arcos y la lista de adyacencia ocupara mucho espacio.

**Pregunta 2:** ¿Qué instrucción se usa para cambiar el límite de recursión de Python? ¿Por qué considera que se debe hacer este cambio?, ¿Cuál es el valor inicial que tiene Python cómo límite de recursión?

`sys.setrecursionlimit(recursionLimit)`, esta es la estructura de datos utilizada para aumentar el numero de llamados recursivos para una sola función, esto se necesita porque por defecto este numero es igual 1000 y Python lo hace para prevenir ciclos infinito. En nuestra ocasión esto nos podría representar una limitación así que tenemos que aumentar el numero de llamados recursivos para poder utilizar nuestro algoritmo de búsqueda DFS, para el cual nunca vamos a llegar a un ciclo infinito si lo adoptamos bien y especificamos que se deben marcar los nodos ya visitados.

**Pregunta 3:** ¿Qué relación creen que existe entre el número de vértices, arcos y el tiempo que toma la operación 4? (Ayuda: ¿es un crecimiento lineal?)

*Archivo 300 -> Vértices: 295, Arcos: 382, establecer estación base: 0.087*

*Archivo 1000 -> Vértices: 984, Arcos: 1633, establecer estación base: 0.295*

*Archivo 7000 -> Vértices: 6829, Arcos: 15342, establecer estación base: 4.145*

*Archivo 14000 -> Vértices: 13535, Arcos: 32301, establecer estación base: 23.27*

Se puede apreciar como el numero de vértices tiene un crecimiento lineal con respecto al numero de datos, mientras que el aumento del número de arcos también tiene una relación positiva pero no tiene un crecimiento en igual proporción que el aumento de la información. El tiempo que toma la operación 4 también crece a medida que hay mayores datos, y por tanto nodos y arcos, este crecimiento temporal pareciera que fuera exponencial y esto intuitivamente si es coherente porque a medida que hay más información sobre distintas rutas, habrá más rutas posibles que podemos elegir dada una estación y un bus también.