**OBSERVACIONES DEL LA PRACTICA**

# Nicolas Saavedra -20212963

# Ernesto Perez– 202112530

# **Preguntas de análisis**

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

El limite de la recursion se obtiene mediante la funcion ‘sys.getrecursionlimit()’, si se quiere cambiar el limite de recursion se deben poner la siguiente linea de codigo: ‘sys.setrecursionlimit()’ y se pone dentro de los parentesis el valor que se quiere tener como limite de recursion.

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

Este cambio es muy pesado en terminos de memoria pues le da el ‘permiso’ a python de poder llamar a una misma funcion mas de el valor por defecto y una vez este valor es sobrepasado el codigo se detiene. Este cambio puede ser importante cuando se trabajan con datos muy grandes y se usan funciones recursivas muy grandes.

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

El límite de recursión por defecto de Python es de 1000. Lo que significa que una función no se puede llamar a si misma más de 1000 veces.

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

El numero de arcos en un grafo siempre sera mayor al numero de vertices, sin embargo no es un valor completamente fijo ya que depende de que tan denso es el grafo. El numero de arcos siempre sera mayor o igual a n y menor a en el peor caso cuando todo el grafo esta conectado. En cuanto al tiempo, este siempre sera proporcional al tamaño del grafo que se le esta cargando informacion y que tan denso este sea.

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

* Es un grafo bastante disperso pues para el archivo de mayor tamaño, la cantidad de vértices es de 13535 y la cantidad de arcos es de 32270 lo que se aproxima más a algo como 2n o 3n y está muy lejos de ser que sería el caso de ser muy denso.
* El grafo es dirigido, es decir que de un aeropuerto se puede llegar a otro, pero no viceversa.
* El número de componentes conectados para el grafo con mayor cantidad de datos es de 30, teniendo en cuenta que la totalidad de vértices es de 13535 por lo que si podemos decir que está fuertemente conectado para tener tantos vértices.

1. ¿Cuál es el tamaño inicial del grafo?

Según la funcion en donde se crea el grafo, este tiene un tamaño de 14000.

1. ¿Cuál es la Estructura de datos utilizada?

Se utiliza un lista de adyacencia para implementar el grafo, esto es bueno cuando el grafo no es muy denso, sin embargo ocupa mas memoria cuando el grafo es muy denso.

1. ¿Cuál es la función de comparación utilizada?

Se usa la funcion llamada ‘compareStopIds( )’, esta es una funcion muy comun y no cambia nada pues simplemente retorna -1 cuando el valor 1 es menor al 2, retorna 0 si son iguales y 1 si el valor 1 es mayor al 2.