**OBSERVACIONES DEL LA PRACTICA**

# Sergio Avendaño - 201923730

# Juan Jose Sierra - 202013642

# **Preguntas de análisis**

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

R: Se usa la funcion “getrecursionlimit()” la cual se usa a través de la biblioteca “sys”. Esta función devuelve el valor del limite de la recursividad, y este limite se encarga de que no se vaya a usar una recursividad infinita y se bloquee Python.



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

R: Como se mencionó anteriormente, es necesario recurrir a esta función ya que al estar manejando una gran cantidad de datos y al ser la estructura de estos tan compleja, se necesita un función que ayude a controlar el consumo de recursos.

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

R: El valor inicial que tiene el límite de recursión es 1000.

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

R: Creemos que existe una relación binaria entre el numero de vértices y arcos.

1. **¿Qué características tiene el grafo definido?**

R: Un grafo definido tiene como característica principal que sus arcos tienen una relación unidireccional es decir que solo tienen un sentido, un vértice uno y un vértice dos, pero no viceversa. Además, siempre va a existir un camino entre un par de nodos, es decir a pesar de que no sea bidimensional siempre van a haber dos arcos entre dos vértices que garanticen la conectividad.

1. **¿Cuál es el tamaño inicial del grafo?**

R: El tamaño inicial del grafo seria 14000 el cual es asignado en la función del newAnalyzer que se encuentra en el model.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

1. **¿Cuál es la Estructura de datos utilizada?**

R: La estructura de datos utilizada para implementar el TAD es una lista de adyacencias (Probing).

1. **¿Cuál es la función de comparación utilizada?**

R: La función de comparación usada es “compareStopTds”, que compara las dos estaciones que entran por parámetro y compara el código con la llave ingresada por parámetro.

Texto

Descripción generada automáticamente