

INFORME ACTIVIDAD 2 TP 2:

Contexto del problema:

Para esta actividad implementamos una base de datos de temperaturas sobre un árbol AVL, el cual trabaja con las mediciones, conformadas por dos datos: la temperatura y la fecha. En el funcionamiento interno del árbol AVL, su criterio de ordenamiento va a ser las fechas, y el peso o clave va a ser la temperatura.

Tabla para los órdenes de complejidad de los métodos utilizados en "Temperaturas_DB":

MÉTODO	ORDEN DE COMPLEJIDAD	EXPLICACIÓN
leer_archivo(nombre_archivo)	$O(m \log n)$	Lee la cantidad de líneas m y por cada línea inserta en el árbol AVL. Cada inserción en AVL es $O(\log n)$
guardar_temperatura(temperatura, fecha)	$O(\log n)$	Inserción en el árbol AVL, actualización y posible rebalanceo.
devolver_temperatura(fecha)	$O(\log n)$	Búsqueda en el árbol AVL.
max_temp_rango(fecha1, fecha2)	$O(k + \log n)$	Recorre el árbol en un rango k de fechas y busca la máxima entre los nodos visitados, el peor caso es que sea el último.
min_temp_rango(fecha1, fecha2)	$O(k + \log n)$	Busca el mínimo entre los nodos visitados, el peor caso es que sea el último.
temp_extremos_rango(fecha1, fecha2)	$O(k + \log n)$	se implementa usando max_temp_rango(fecha1, fecha2) y min_temp_rango(fecha1, fecha2), por lo que su orden de complejidad es el mismo que los mencionados.
borrar_temperatura(fecha)	$O(\log n)$	Búsqueda, eliminación y rebalanceo en el árbol AVL.
devolver_temperaturas(fecha1, fecha2)	$O(k + \log n)$	Recorrido inorden filtrado, en el peor de los casos, recorre todos los nodos.
cantidad_muestras()	$O(1)$	Devuelve el tamaño almacenado. Es $O(1)$ porque tenemos un parámetro de la

		instancia del objeto que almacena este dato
--	--	--

Cabe mencionar que el recorrido del árbol AVL en un rango de temperaturas es $O(k)$ donde k sería la cantidad de fechas en el rango