

Análisis de Complejidad Big-O

Tabla con análisis de complejidad para cada método de la clase **Temperaturas_DB**.

- **N**: Número total de muestras (nodos) en el árbol.
- **K**: Número de *días* en el rango consultado (es decir, **fecha2 - fecha1**).
- **M**: Número de líneas en el archivo de muestras.

Método	Complejidad	Análisis
guardar_temperatura	$O(\log N)$	Corresponde a una operación de inserción (agregar) en un árbol AVL, que incluye la búsqueda del lugar y las posibles rotaciones para rebalancear.
devolver_temperatura	$O(\log N)$	Corresponde a una operación de búsqueda (obtener o <code>__getitem__</code>) en un árbol AVL.
borrar_temperatura	$O(\log N)$	Corresponde a una operación de eliminación (eliminar) en un árbol AVL, que incluye la búsqueda y el rebalanceo.
cantidad_muestras	$O(1)$	Simplemente devuelve el atributo <code>self.tamano</code> del árbol, que se mantiene actualizado en $O(1)$ tras cada inserción/eliminación.
max_temp_rango	$O(K \cdot \log N)$	Ver explicación abajo.
min_temp_rango	$O(K \cdot \log N)$	Ver explicación abajo.
devolver_temperaturas	$O(K \cdot \log N)$	Ver explicación abajo.
temp_extremos_rango	$O(K \cdot \log N)$	Llama a <code>min_temp_rango</code> y <code>max_temp_rango</code> , por lo que su costo es $2 \cdot O(K \cdot \log N)$, que se simplifica a $O(K \cdot \log N)$.
cargar_muestras	$O(M \cdot \log N)$	Itera M veces (una por cada línea del archivo). En cada iteración, realiza un <code>guardar_temperatura</code> , que cuesta $O(\log N)$. El costo total es M inserciones. (Donde N crece hasta M).