

## Problema 2

Para analizar los órdenes de complejidad de los métodos implementados primero se deben de reconocer la complejidad de las funciones dentro del AVL auxiliar, pues todas las implementaciones se construyen a partir de estas.

Función	Complejidad Big-O	Análisis
Agregar	$O(\log n)$	Al agregar un elemento, el árbol realiza un reequilibrio; para ello se hace una actualización de factor de equilibrio; dicho esto, se debe recorrer cada nivel, es decir que la complejidad es $O(\log n)$ , en caso de que la actualización requiera rotaciones, estas son de $O(1)$ , por ende no afecta a la complejidad final.
Obtener/Buscar	$O(\log n)$	La obtención recorre de manera dirigida, se compara la clave buscada con los hijos de cada nodo y se selecciona una ruta a seguir pues el árbol se encuentra ordenado, con cada elemento teniendo un valor más pequeño a la izquierda y un mayor a la derecha. De esta manera no se requiere recorrer todo el árbol, sino un camino logarítmico.
Máx/Mín	$O(n)$	En la búsqueda de valores extremos, se deben recorrer todos los elementos para compararlos y guardar los mayores o menores. Como se analizan todos los valores, la complejidad de la función es $O(n)$ . Para la búsqueda en rangos, la complejidad seguirá siendo lineal ya que el peor caso será el rango completo.
Eliminar	$O(\log n)$	La eliminación de un elemento se basa en buscarlo y una vez encontrado, se extrae; luego el árbol debe de empalmar el nodo reemplazante con el sucesor elegido según el caso; posteriormente se reequilibra en caso de ser necesario.

Se realizó un análisis de los órdenes de complejidad para los métodos implementados para la clase “Temperatura\_DB”.

Función	Complejidad Big-O	Análisis
convertir_fecha	$O(1)$	Convierte un string a un objeto “datetime”.
guardar_temperatura	$O(\log n)$	Convierte la fecha y utiliza la función “agregar” del AVL auxiliar.
devolver_temperatura	$O(\log n)$	Convierte la fecha y utiliza la función “obtener” del AVL auxiliar.
max_temp_rango	$O(n)$	En el peor caso recorre todos los nodos utilizando la función de máximo del AVL auxiliar.

Función	Complejidad Big-O	Análisis
<code>min_temp_rango</code>	$O(n)$	En el peor caso recorre todos los nodos utilizando la función de mínimo del AVL auxiliar.
<code>temp_extemos_rango</code>	$O(n)$	Utiliza las dos funciones anteriores.
<code>devolver_todas_temperaturas</code>	$O(n)$	Recorre siempre todos los nodos iterando a través de un inorden en el AVL auxiliar y obteniendo cada valor.
<code>devolver_temperaturas</code>	$O(n)$	En el peor caso (rango completo) recorre todos los nodos al igual que la función anterior.
<code>cantidad_muestras</code>	$O(1)$	Retorna el valor de tamaño almacenado.
<code>borrar_temperatura</code>	$O(\log n)$	Convierte la fecha y utiliza la función eliminar en el AVL auxiliar.

---

En el módulo “Test\_implementation” se verifica a través de pruebas el correcto rendimiento de cada algoritmo desarrollado previamente. Para ello se introducen fechas secuenciales con temperaturas arbitrarias y se ejecutan las distintas funciones.