

## Proyecto °2

### Objetivos

- Implementar una base de datos de temperaturas “**Temperaturas\_DB**” que utilice internamente un Árbol AVL. Que permita realizar las operaciones dictadas (interfaz de Temperaturas\_DB).
- Escribir una tabla con el análisis de complejidad O-grande para **cada uno** de los métodos implementados
- Implementar un método de lectura de un archivo de [muestras](#) que permita cargar las mismas a la base de datos de temperaturas.

### Desarrollo

- El presente programa implementa una base de datos de temperaturas con un Árbol AVL incorporado. El programa permite la lectura de archivos de texto con temperaturas (en grados celsius) y fechas (DD/MM/AAAA) en formato [temperatura];[fecha].

### Complejidad temporal:

- **guardar\_temperatura(temperatura, fecha):**
  - Esta función al llamar al método **insertar** del Árbol AVL por lo que tendrá su misma complejidad temporal:  $O(\log(n))$
- **devolver\_temperatura(fecha):**
  - Una que usa el método **obtener** del árbol AVL (método de búsqueda) cuya complejidad temporal es:  $O(\log(n))$ , que en este caso también se mantiene.
  - **max\_temp\_rango(fecha1, fecha2)** y **min\_temp\_rango(fecha1, fecha2):**
    - En este caso el programa utiliza el metodo de búsqueda ( $O(\log(n))$ ) para encontrar **fecha1** (más cercano), una vez lo tiene, el programa continúa visitando todos los **k** nodos que se encuentran después de

**fecha1** e irlos comparando hasta dar con el extremo final **fecha2** ( $O(k)$ ) y determinar máximo o mínimo. Por ende, la complejidad temporal total de esta función es:

- $O(\log(n)) + O(k) = O(\log(n) + k)$

- **temp\_extremos\_rango(feche1, fecha2):**
  - Esta función realiza esencialmente la misma operación que la anterior pero en forma simultánea, por lo tanto la complejidad se mantiene:  $O(\log(n) + k)$
- **borrar\_temperatura(feche):**
  - Hace uso del método eliminar del árbol AVL. Dependiendo el caso, esta operación puede requerir mas o menos rotaciones para reequilibrar el arbol, una por nivel, por lo que la complejidad temporal para todos los casos es de  $O(\log(n))$ .
- **devolver\_temperaturas(feche1, fecha2):**
  - Esta operación también hace uso del método **obtener** del árbol. Y genera el mismo recorrido entre extremos que **max\_temp\_rango**, por lo tanto la complejidad de esta función también será  $O(\log(n) + k)$ .
- **cantidad\_muestras():**
  - La función toma el valor del atributo **longitud** de la clase Arbol AVL. Complejidad temporal:  $O(1)$