

## Árboles balanceados

En muchas ocasiones, las inserciones y eliminaciones de elementos en el árbol no ocurren en un orden predecible, es decir, los datos no están organizados jerárquicamente. Para eso existen tipos de árboles distintos, como los árboles equilibrados o árboles AVL, que ayudan eficientemente a resolver estas situaciones.

La eficiencia para una búsqueda de una clave es un árbol binario de búsqueda varía entre  $O(n)$  y  $O(\log(n))$ , dependiendo de la estructura que represente el árbol.

Si los elementos se añaden en el árbol mediante un algoritmo de inserción como el de un árbol binario común, la estructura resultante del árbol dependerá del orden en que sean añadidos. Así, si todos los elementos se insertan en orden creciente o decreciente, el árbol va a tener todas las ramas izquierda o derecha, respectivamente, vacías. Entonces las búsquedas de dicho árbol serán totalmente secuencial.

Sin embargo, si la mitad de los elementos insertados después de otro con clave  $k$  tiene claves menores de  $k$  y la otra mitad claves mayores de  $k$ , se obtiene un árbol equilibrado o balanceado.

El fin de crear a los árboles balanceados reside en mejorar el rendimiento de búsqueda, si bien, en árboles pequeños no produce tanta diferencia, en árboles inmensos ahorra mucho tiempo y recursos para dichas búsquedas.

**Para las inserciones de nuevos nodos, se debe ocurrir que se cumplan los criterios de balanceo.**

Los recorridos de un árbol equilibrado son los mismos que un árbol binario, con la diferencia de que los equilibrados tienen un orden establecido desde su creación y dichos recorridos son más rápidos.

Los recorridos son:

- Recorrido inorden:
  - ❖ Recorrer el subárbol izquierdo.
  - ❖ Visitar el nodo raíz.
  - ❖ Recorrer el subárbol derecho.

- Recorrido preorden:
  - ❖ Visitar la raíz.
  - ❖ Recorrer el subárbol izquierdo.
  - ❖ Recorrer el subárbol derecho.
  
- Recorrido postorden:
  - ❖ Recorrer el subárbol izquierdo.
  - ❖ Recorrer el subárbol derecho.
  - ❖ Visitar la raíz.

**Referencia:**

*Luis Joyane Aguilar, Ignacio Zahonero Martínez. (1998). Estructura de datos. España: McGraw-Hill.*