

# Árboles B+

## Descripción General

Los árboles B+ son una de estructura de datos derivada de los árboles B. La principal diferencia se encuentra en que los árboles B+ guardan toda la información en sus hojas, mientras que en el resto del árbol se guardan “llaves” para llegar a dicha información.

Un árbol B+ tiene un nivel de ramificación u orden que indica la manera de estructurarlo. En un árbol de orden  $n$ , la raíz puede tener desde 1 hasta  $n$  datos y al menos dos descendientes. Las ramas deben tener entre  $n/2$  y  $n$  datos y como se ha indicado anteriormente, estos son las llaves para llegar a la información y por ello pueden estar repetidos. Las hojas deben tener todas la misma altura y en ellas deben estar guardados todos los datos, además, estos deben estar ordenados. De este modo, si tomamos un  $M$  que será el número máximo de de hijos que un nodo puede tener, entonces la altura que el árbol B+ tendrá, será  $\log M(n)$  en el peor de los casos y  $\log[M/2](n)$  en el mejor.

Finalmente, los árboles B+ suponen una mejora para los árboles B pues conservan la propiedad de acceso aleatorio rápido y permiten además un recorrido secuencial rápido. Por otro lado, debido a la repetición de datos entre ramas, para guiar el camino de acceso a la información, estos ocupan significativamente un poco más de memoria.

## Métodos de un Árbol B+

*Búsqueda:* Las búsquedas en un árbol B+ se realizan de manera similar a las búsquedas realizadas en un árbol B, con la diferencia de tener que asegurarnos de que nos encontramos en una hoja y no en una rama.

*Inserción:* La inserción también es parecida a la inserción en un árbol B, pero cuando se inserta un nuevo dato en un nodo lleno, este se divide en dos, pero el primero contendrá  $n/2$  datos y el segundo  $1+n/2$ , además lo que subirá al nodo anterior será la clave central del nodo original.

*Borrar:* Al igual que las otros dos métodos, el borrar es similar al de un árbol B, este, sin embargo tiene dos características nuevas:

- Si al eliminar un dato, que debe ser siempre de una hoja, el número de datos es mayor o igual a  $n/2$  el proceso ha terminado. Los datos de los nodos internos no se modifican aunque sean una copia del eliminado, pues siguen constituyendo un separador válido entre los datos de los nodos descendientes.
- Si al eliminar el dato, el número de datos en el nodo es menor que  $n/2$ , entonces será necesaria una fusión y redistribución de los mismos tanto en los nodos hojas como en el índice.

## Usos de los Árboles B+

Los árboles B+ se utilizan principalmente en bases de datos, por su rápido acceso a los datos guardados y su capacidad de crecer indefinidamente. Otro uso es en los sistemas de archivos, por la misma razón, además de que con el uso de ellos se reduce el número de accesos al disco. Finalmente, los árboles B+ son usados por algunos dispositivos como los usb, para el acomodo y acceso de los archivos que guardan. Como se puede ver, la mayoría de los usos de este tipo de árboles, está en el almacenamiento de largos volúmenes de archivos a los que se quiere acceder rápidamente, asumiendo un pequeño costo de memoria.

## Bibliografía

- ❖ <https://www.javatpoint.com/b-plus-tree>
- ❖ <https://www.geeksforgeeks.org/introduction-of-b-tree/>
- ❖ <https://sites.google.com/site/tutoriasarboles/arboles-b-y-b>
- ❖ [https://en.wikipedia.org/wiki/B%2B\\_tree](https://en.wikipedia.org/wiki/B%2B_tree)