



B Tree & B+ Tree

+ B Tree

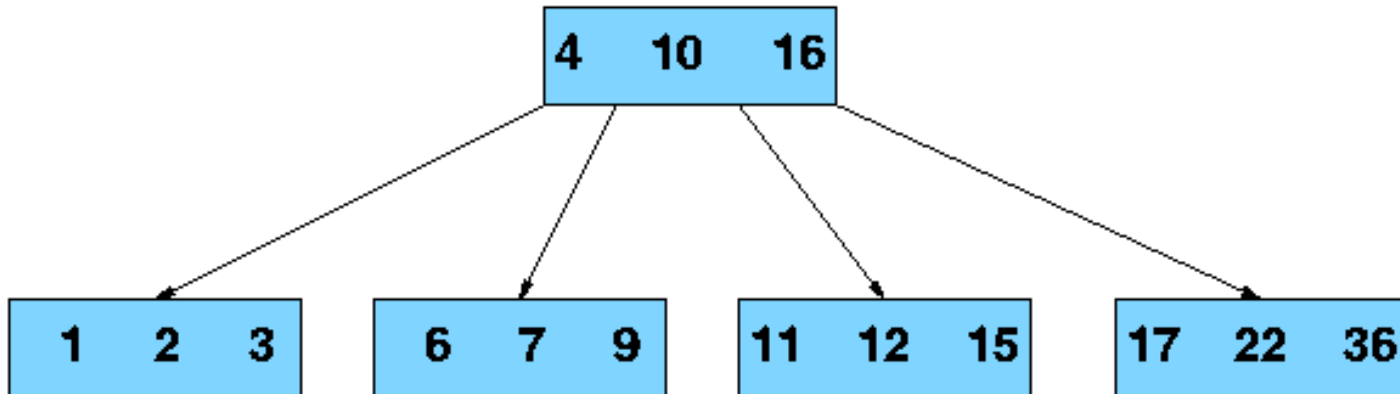
- In computer science, a **B-tree** is a tree data structure that keeps data sorted and allows searches, insertions, and deletions in logarithmic amortized time. It is most commonly used in databases and filesystems.
- Each node of a b-tree may have a variable number of keys and children.
- Each key has an associated child that is the root of a subtree containing all nodes with keys less than or equal to the key but greater than the preceeding key.
- A node also has an additional rightmost child that is the root for a subtree containing all keys greater than any keys in the node.

+ B Tree

- A b-tree has a minimum number of allowable children for each node known as the *minimization factor*. If t is this *minimization factor*, every node must have at least $t - 1$ keys.
- Since each node tends to have a large branching factor (a large number of children), it is typically necessary to traverse relatively few nodes before locating the desired key.
- If access to each node requires a disk access, then a b-tree will minimize the number of disk accesses required.
- The minimization factor is usually chosen so that the total size of each node corresponds to a multiple of the block size of the underlying storage device.

B Tree

- This choice simplifies and optimizes disk access. Consequently, a b-tree is an ideal data structure for situations where all data cannot reside in primary storage and accesses to secondary storage are comparatively expensive (or time consuming).



+ B+ Tree

- a **B+ tree** (also known as a **Quaternary Tree**) is a type of tree, which represents sorted data in a way that allows for efficient insertion, retrieval and removal of records, each of which is identified by a *key*.
- In a B+ tree, in contrast to a B-tree, all records are stored at the lowest level of the tree; only keys are stored in interior blocks.
- The [ReiserFS](#) filesystem (for [Unix](#) and [Linux](#)), [XFS](#) filesystem (for [IRIX](#) and [Linux](#)), [JFS2](#) filesystem (for [AIX](#), [OS/2](#) and [Linux](#)) and [NTFS](#) filesystem (for [Microsoft Windows](#)) all use this type of tree for block indexing. [Relational databases](#) also often use this type of tree for table indices.

+ B+ Tree

