Full Binary Tree

A full binary tree is also known as 2-tree in which every node other than the leaf nodes has two child nodes. It means all the leaf nodes should be at the same level and all other internal nodes should contain two child nodes each.

Complete Binary Tree

It follows the definition of a binary tree with some additional conditions. **In a complete binary tree, all the levels of a tree are filled entirely except the last level. In the last level, nodes might or might not be filled fully. Also, let's note that all the nodes should be filled from the left.**

Almost Complete Binary Tree

An almost complete binary tree is a special kind of binary tree where insertion takes place level by level and from left to right order at each level and the last level is not filled fully always. It also contains   nodes at each level except the last level.

The main point here is that if we want to insert any node at the lowest level of the tree, it should be inserted from the left. All the internal nodes should be completely filled.

Max-heap

A max-heap is **a complete binary tree in which the value in each internal node is greater than or equal to the values in the children of that node**. Mapping the elements of a heap into an array is trivial: if a node is stored an index k, then its left child is stored at index 2k + 1 and its right child at index 2k + 2.

Min-heap

A Min Heap Binary Tree is **a Binary Tree where the root node has the minimum key in the tree**. The above definition holds true for all sub-trees in the tree. This is called the Min Heap property. Almost every node other than the last two layers must have two children.