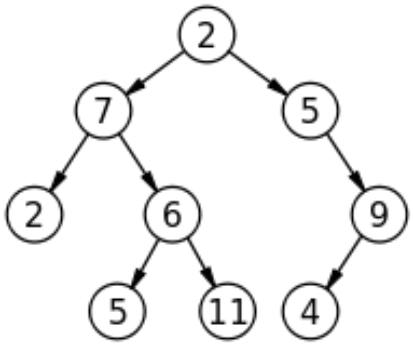
A **binary tree** is a tree data structure in which each node has at most two children, which are referred to as the left child and the right child.

From the recursive definition, a binary tree is a tuple (L, S, R), where L and R are binary trees or the empty set and S is a singleton set.



A labeled binary tree of size 9 and height 3, with a root node whose value is 2. The above tree is unbalanced and not sorted.

2, 5, 11, and 4 are leaf nodes, because they don't have children.

From a graph theory perspective, binary trees are arborescences. It is possible to interpret a binary tree as a undirected, rather than a directed graph, in which case a binary tree is an ordered, rooted tree.

Some authors use rooted binary tree instead of binary tree to emphasize the fact that the tree is rooted, but as defined above, a binary tree is always rooted. A binary tree is a special case of an ordered K-ary tree, where k is 2.

In mathematics, what is termed binary tree can vary significantly from author to author. Some use the definition commonly used in computer science, but others define it as every non-leaf having exactly two children and don't necessarily order the children either.

In computing, binary trees are used in two very different ways:

- First, as a means of accessing nodes based on some value or label associated with each node. Binary trees labelled this way are used to implement binary search trees and binary heaps, and are used for efficient searching and sorting. The designation of non-root nodes as left or right child even when there is only one child present matters in some of these applications, in particular it is significant in binary search trees.
- Second, as a representation of data with a relevant bifurcating structure. In such cases the particular arrangement of nodes under and/or to the left or right of other nodes is part of the information. Common examples occur with Huffman coding and cladograms. The everyday division of documents into chapters, sections, paragraphs, and so on is an analogous example with n-ary rather than binary trees.