Introduction

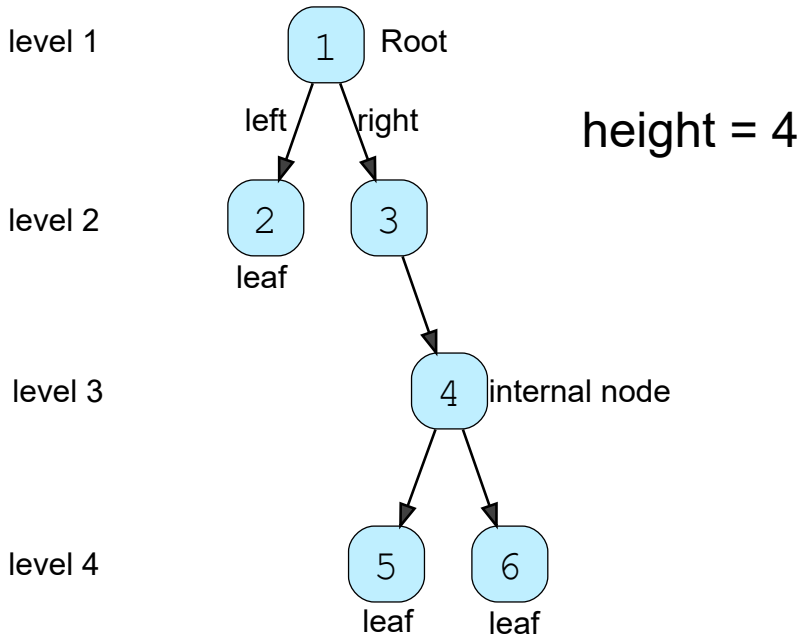
In this lesson, we'll use binary trees, a tree based structure that is the basis for many advanced data structures.

We'll cover the following

Vocabulary

A **binary tree** is a hierarchical data structure.

- The topmost node is called the **root**.
- Each node can have **up to two children**, hence the name Binary tree.
- The two children of a node are the **left** and **right** children.
- The node above a node is called its **parent**. Root has no parent.
- Nodes with no children are called leaves or external nodes.
- Non-Leaf nodes are called **internal nodes**.
- The number of levels is the **height** of the tree.
- **Siblings:** Nodes with the same parent.
- **Descendant:** Node reachable by traversing children (Nodes in the Subtree).
- **Ancestor:** Nodes reachable by traversing parents (Nodes on the path from the node to root).



Node structure

Each node needs two pointers along with a value. The structure would look something like this.

```
struct Node {
  int val;
  Node *left, *right;
}
```

Properties

- The maximum possible nodes at level l are 2^{l-1} as the number of nodes double in the next level.
- The maximum possible nodes in binary tree of height h is $1 + 2 + 4 + ... + 2^{h-1} = 2^h - 1$

In the next lesson, we'll cover some theory on binary trees.