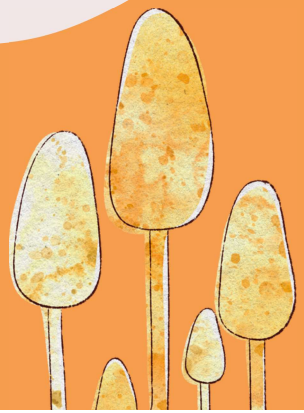
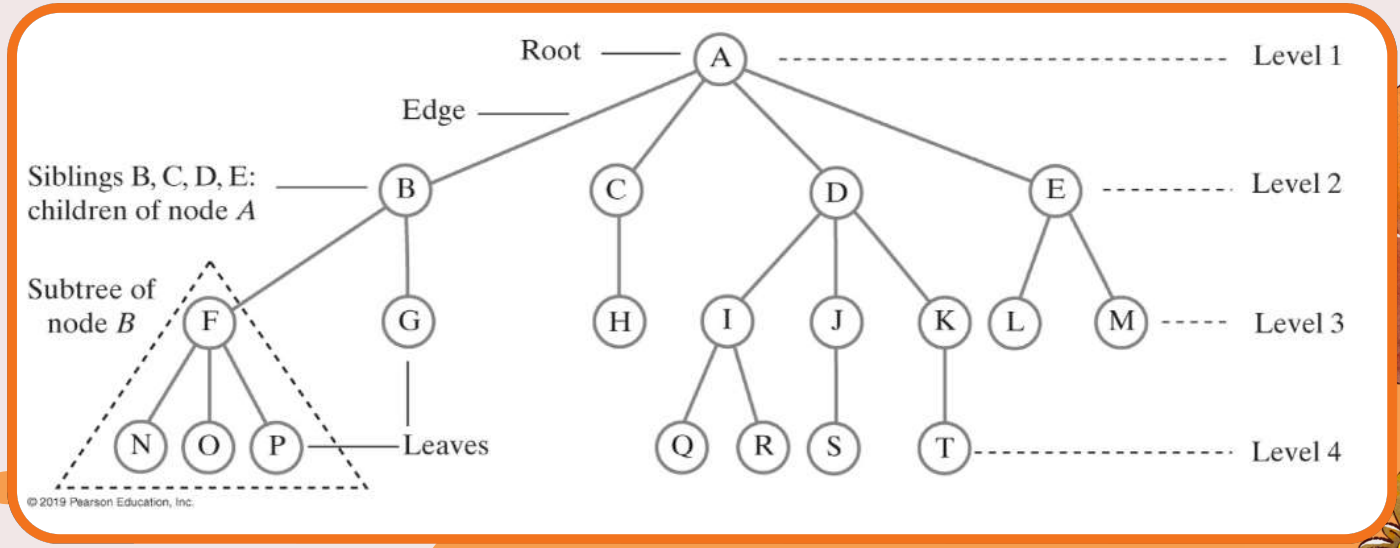
# TREE TRAVERSAL TECHNIQUES

KariAnn Harjo



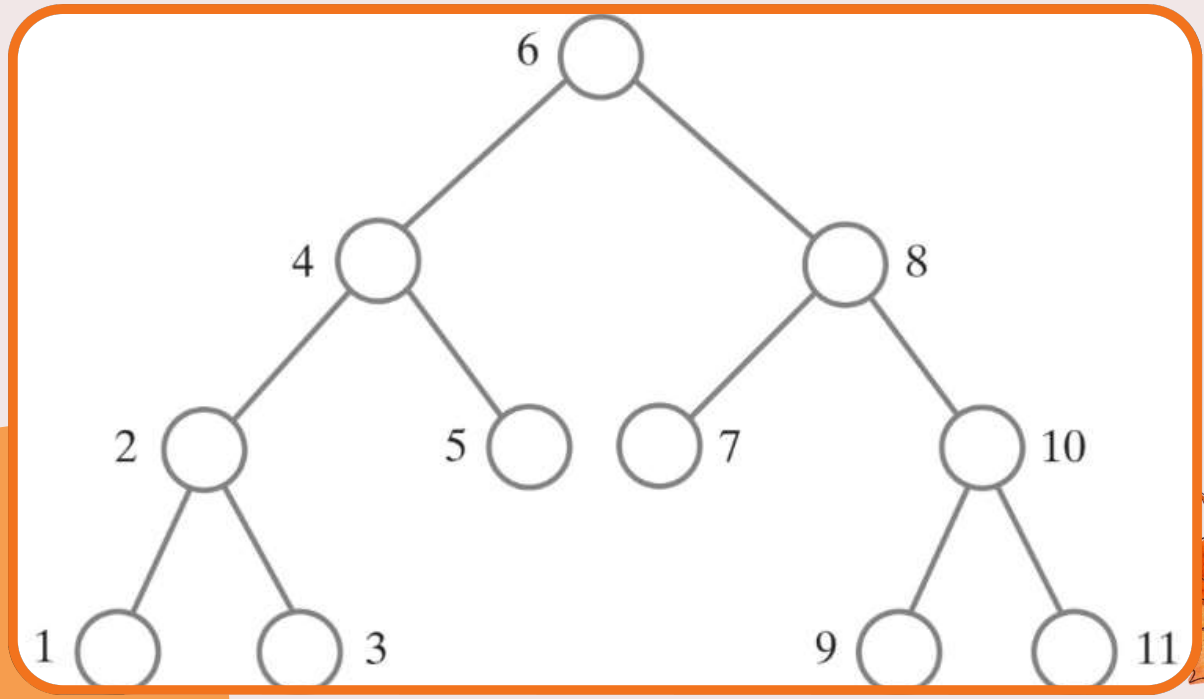
# WHAT IS A BINARY TREE?

A Binary Tree is a data structure made up of nodes connected by edges, where each node contains a value and two references to two other nodes, representing the subtree to the left and right.



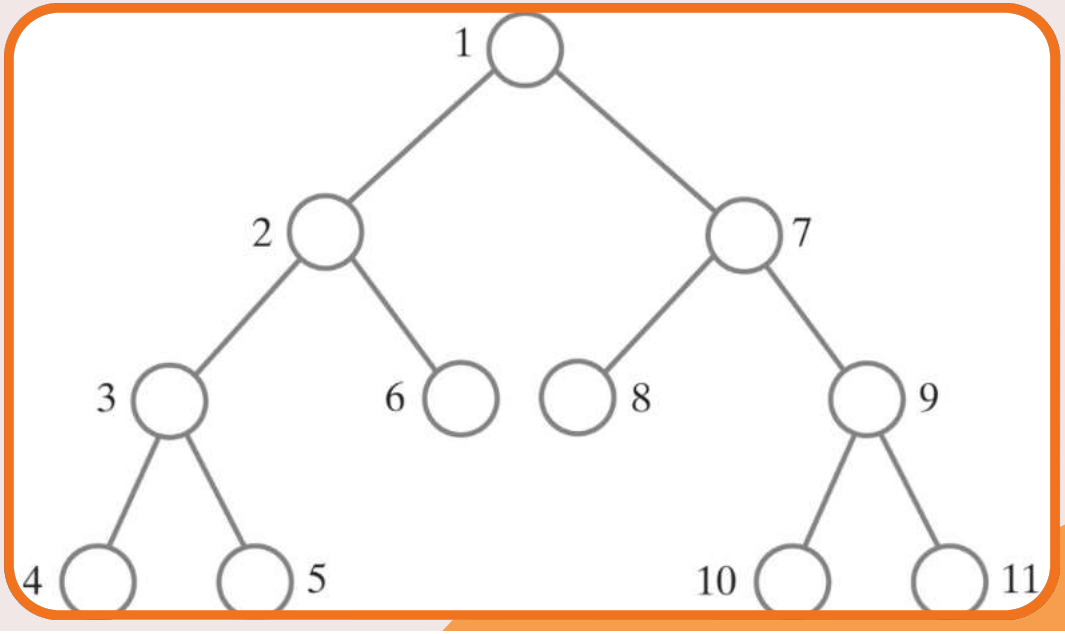
# IN-ORDER TRAVERSAL

In-Order Traversal means that we will visit the root of a binary tree between visiting nodes in root's subtrees.



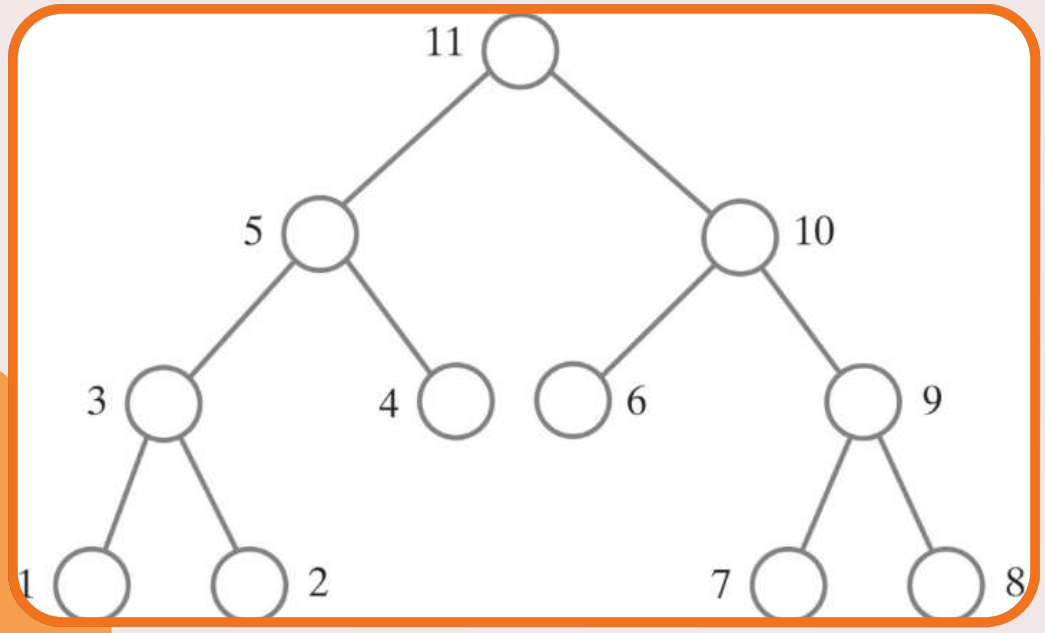
# PRE-ORDER TRAVERSAL

Pre-Order Traversal means that we will visit the root before we visit root's subtrees.



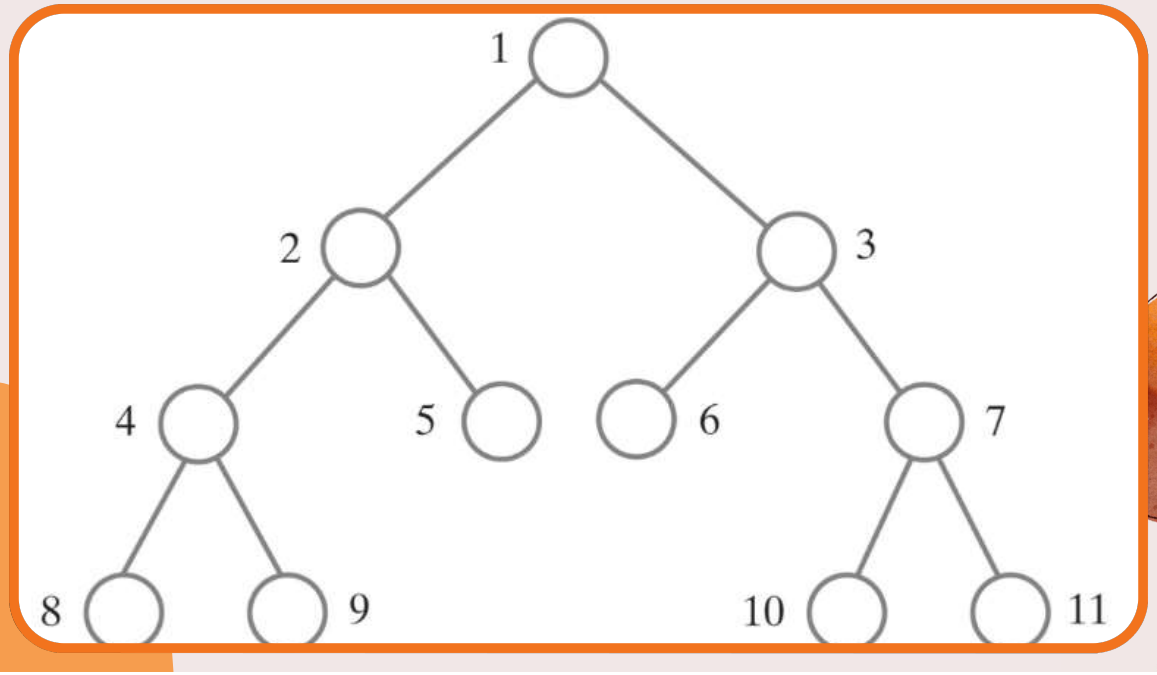
# POST-ORDER TRAVERSAL

Post-Order Traversal means that we will visit the root of a binary tree after visiting nodes in the root's subtrees.



# LEVEL-ORDER TRAVERSAL

Level-Order Traversal means that we will begin at the root and visit nodes one level at a time.



# COMPARING TRAVERSALS

**In-**

Reveals the nodes in a non-decreasing order in Binary Search Trees. It provides a natural or sorted sequence of the elements in the tree.

**Pre-**

Starts from the root and explores as far as possible along each branch before backtracking. It reflects the structure of the tree and shows the hierarchy clearly.

**Post-**

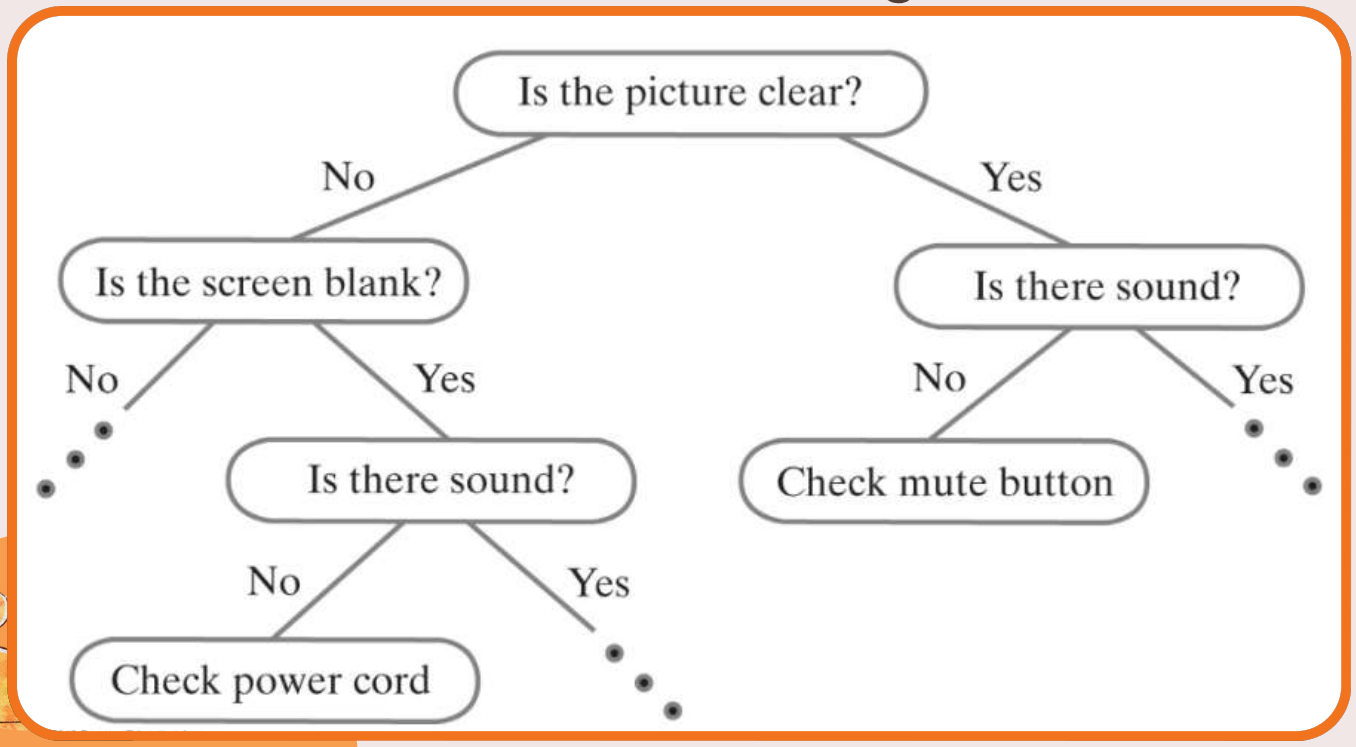
Processes all the children of a node before the node itself. It provides a way to traverse from the leaves up to the root, which is useful for recursive computations and deletions.

**Level-**

Traverses the tree level by level, starting from the root and moving to the next levels sequentially. It gives a clear view of the tree structure at each depth or level.

# PRACTICAL APPLICATION

## Troubleshooting



# THANK YOU