

# ECE 220 Computer Systems & Programming

Trees: traversal and search



# Tree Data Structure

Array, linked list, stack, queue – linear data structures

**Tree:** a collection of nodes connected by edges. It's a *nonlinear* data structure.

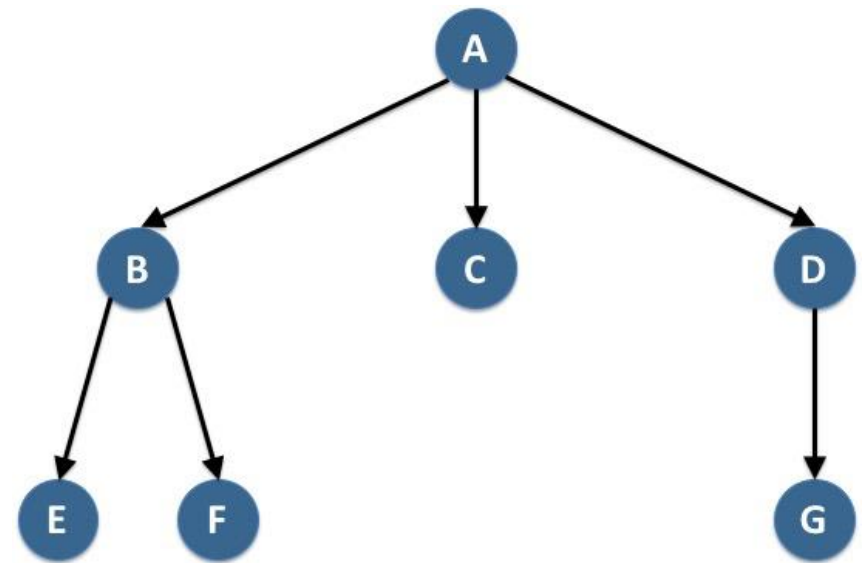
## Tree Terminology:

root, internal node, external node (leaf)

parent, child, sibling

height, depth, level

- Elements are stored hierarchically
- Tree has at most \_\_\_\_\_ root
- Besides the root, each element has \_\_\_\_\_ parent
- Each element has \_\_\_\_\_ children



# Binary Tree

- Each node has at most 2 children – left child and right child

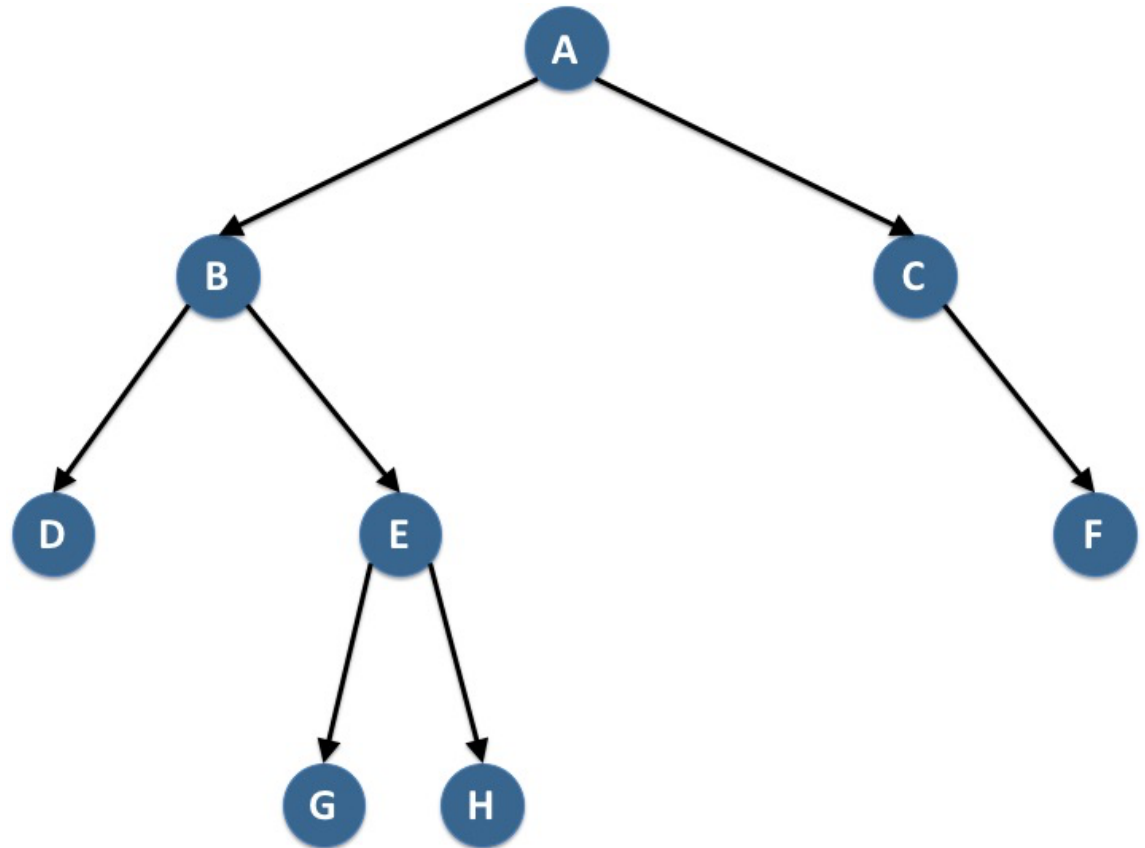
What is the height of the tree?

What is the depth of node E?

What is the height of node E?

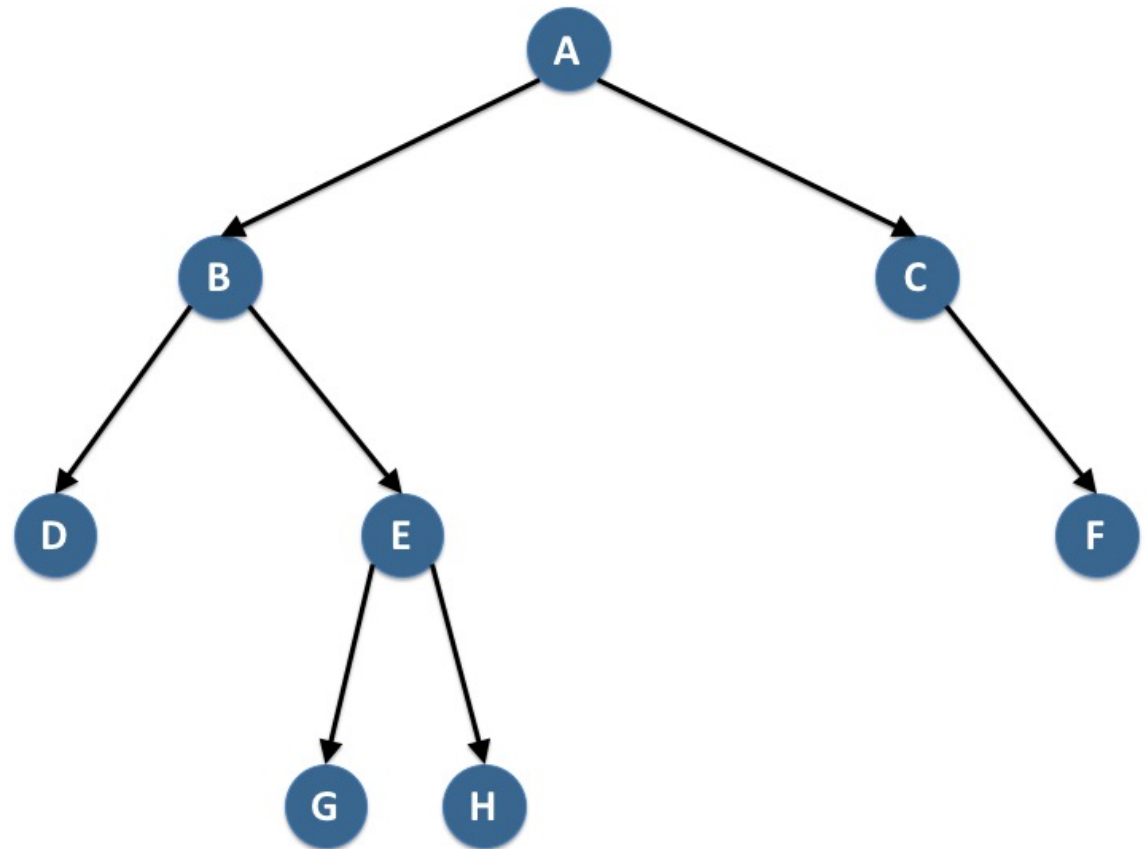
Which nodes are internal nodes?

Which nodes are leaves?



# Tree Traversal – BFS & DFS

## Breadth-First Search (level-order)



## Depth-First Search

1. Pre-order: **root**, left, right
2. In-order: left, **root**, right
3. Post-order: left, right, **root**

# Binary Search Tree

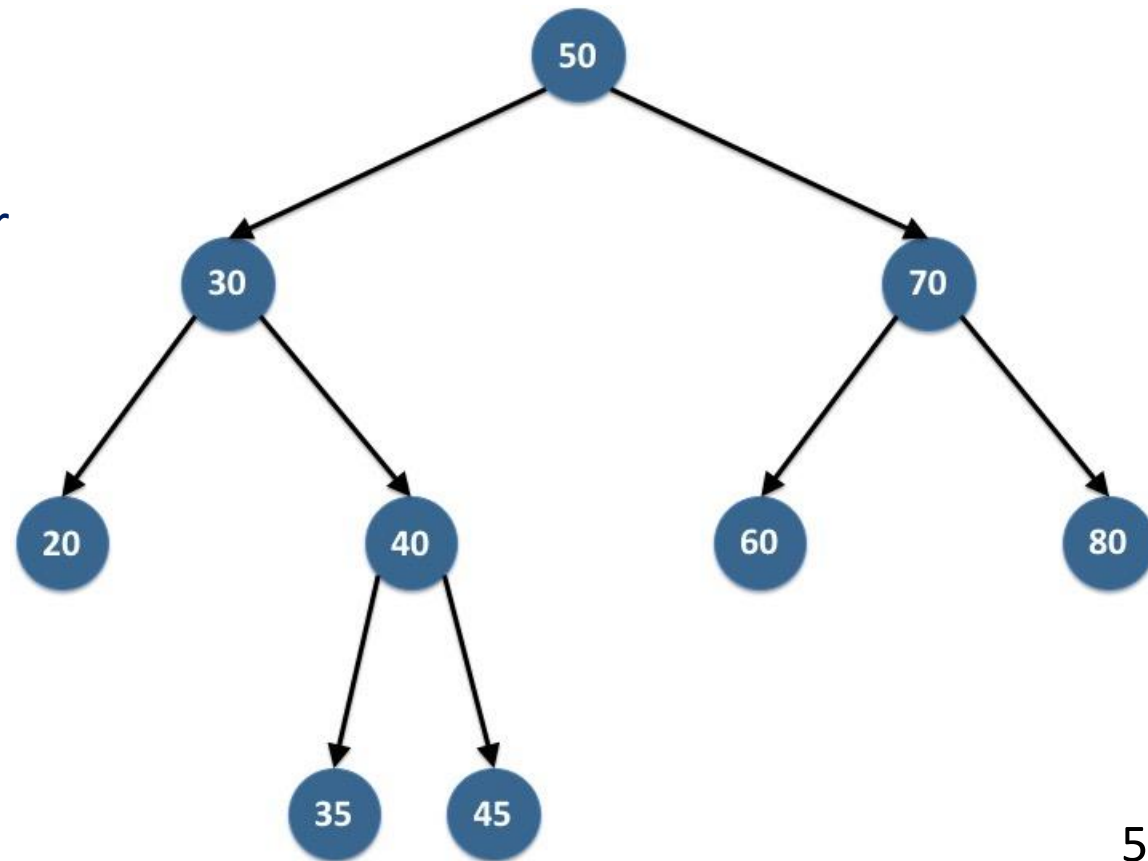
- Data of nodes on the **left subtree** is **smaller** than the data of parent node
- Data of nodes on the **right subtree** is **larger** than the data of parent node
- Both left and right subtrees must also be BST
- Data in each node is unique

What is the sequence of access for

1. pre-order traversal?

2. in-order traversal?

3. post-order traversal?



<http://visualgo.net/bst.html>