

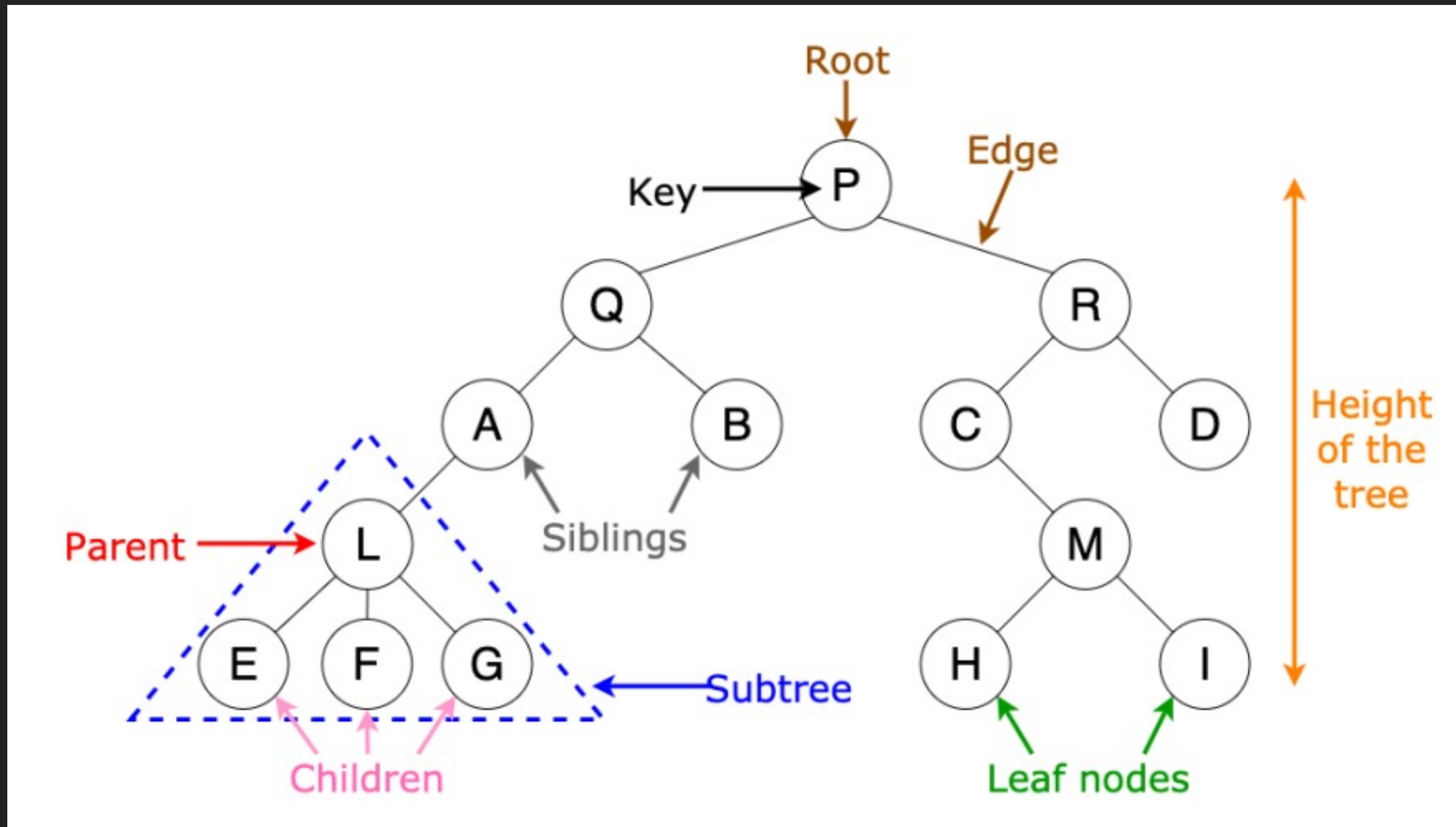
Data Structure & Algorithm

Data Structure

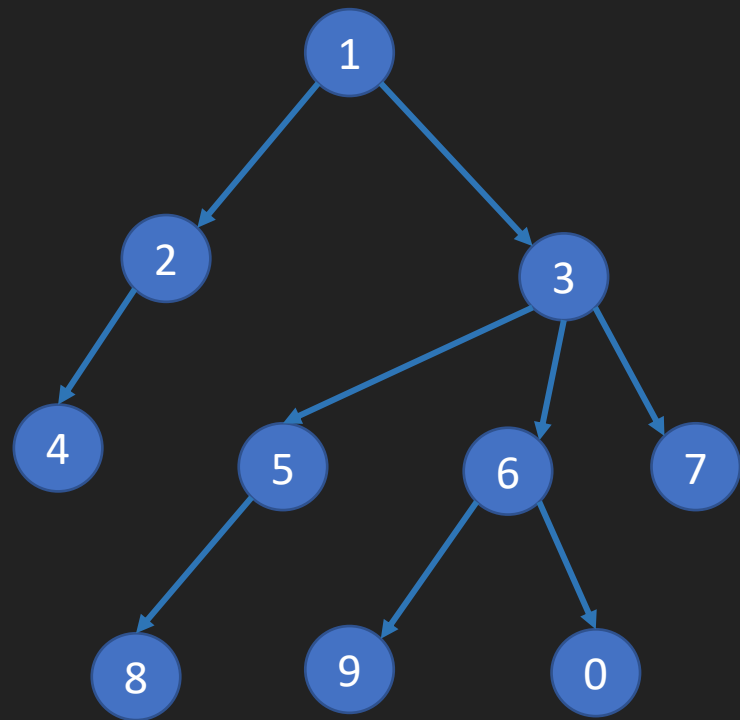
Tree - Binary Tree -
Binary Search Tree

1. Định nghĩa

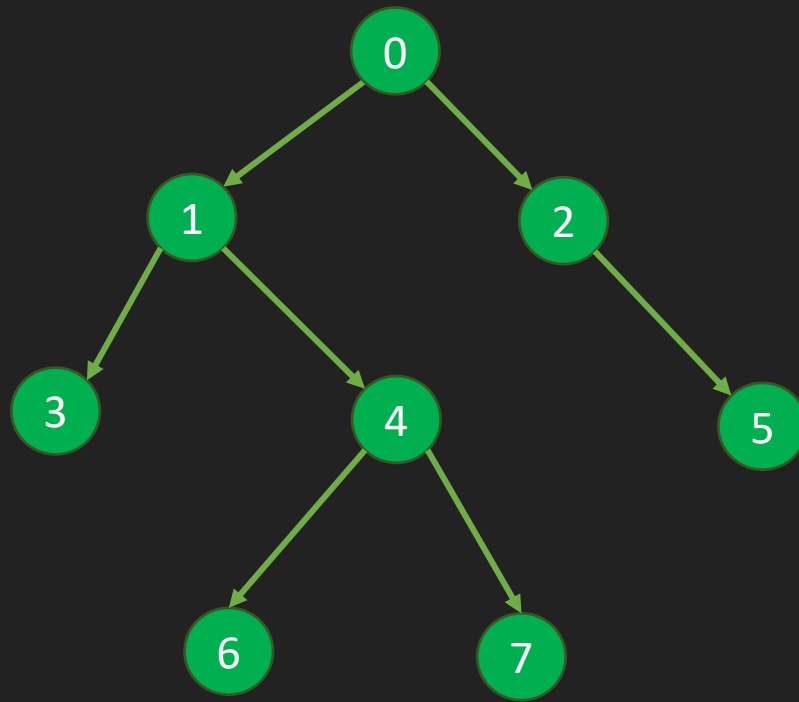
➤ Tree



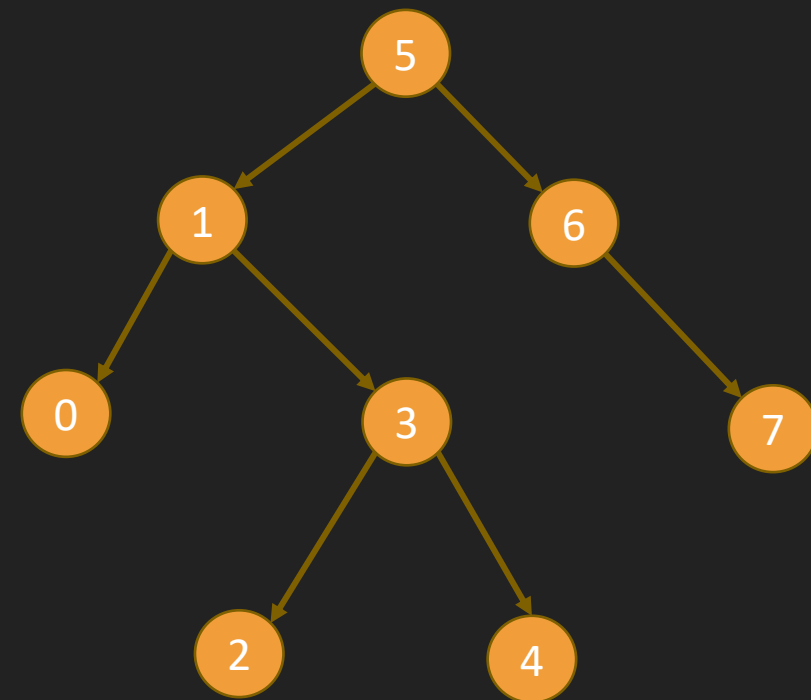
2. Phân loại



➤ Tree



➤ Binary Tree*



➤ Binary Search Tree*

3. Ứng dụng

➤ Tree

- ✓ Cây thư mục
- ✓ Mục lục

➤ Binary Tree*

➤ Binary Search Tree*

- ✓ Thực hiện các thao tác tìm kiếm

4. Tree Structure

```
public class TreeNode {  
    int val;  
    TreeNode left;  
    TreeNode right;  
  
    TreeNode() {}  
  
    TreeNode(int val) {  
        this.val = val;  
    }  
  
    TreeNode(int val, TreeNode left, TreeNode right) {  
        this.val = val;  
        this.left = left;  
        this.right = right;  
    }  
}
```

5. Traverse a Tree

➤ There are 3 types of traverse:

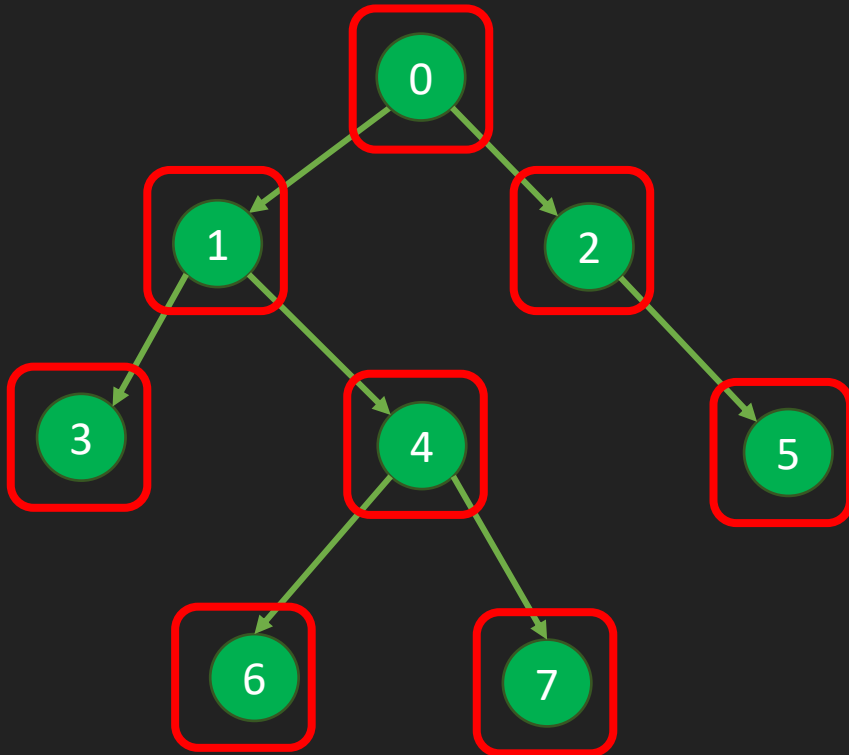
❖ Pre-order Traversal \Rightarrow CurentNode \rightarrow L \rightarrow R

❖ In-order Traversal \Rightarrow L \rightarrow CurrrentNode \rightarrow R

❖ Post-order Traversal \Rightarrow L \rightarrow R \rightarrow CurrentNode

5. Traverse a Tree

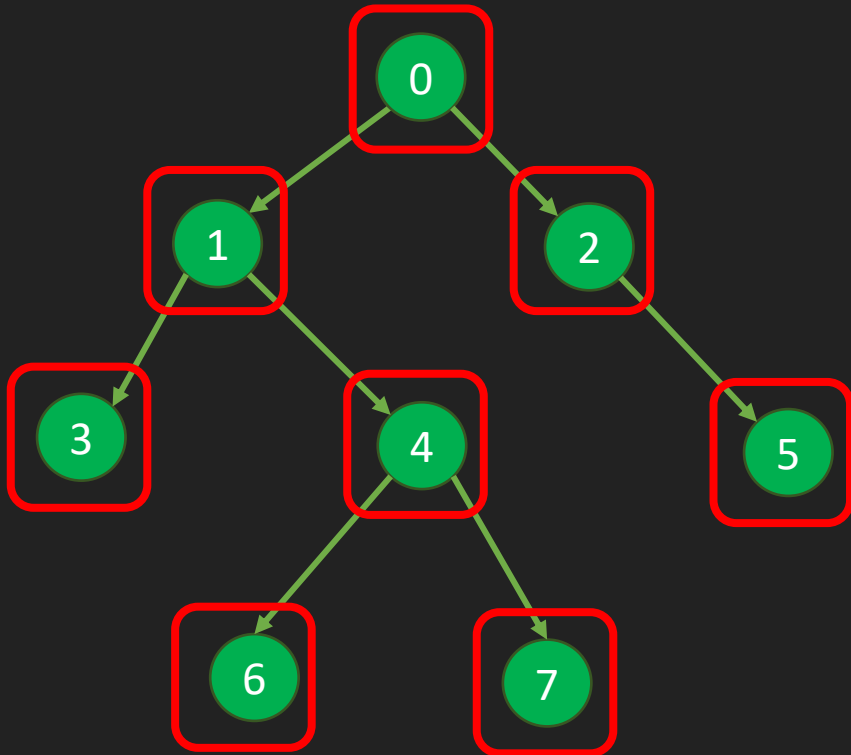
❖ Pre-order Traversal (N-L-R)



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 3 | 4 | 6 | 7 | 2 | 5 |
|---|---|---|---|---|---|---|---|

5. Traverse a Tree

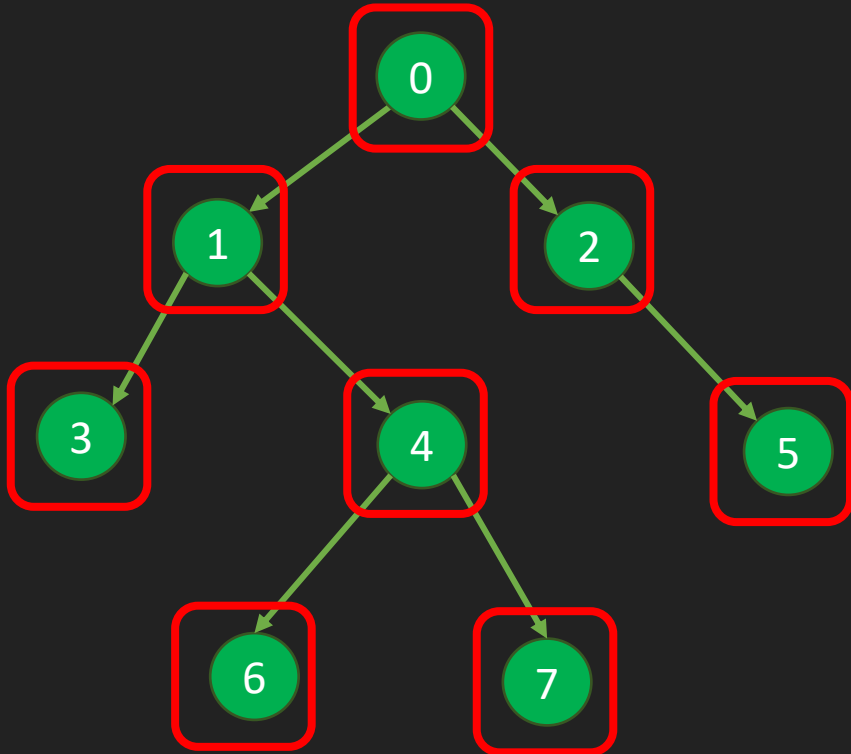
❖ In-order Traversal (L-N-R)



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 3 | 1 | 6 | 4 | 7 | 0 | 2 | 5 |
|---|---|---|---|---|---|---|---|

5. Traverse a Tree

❖ Post-order Traversal (L-R-N)



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 3 | 6 | 7 | 4 | 1 | 5 | 2 | 0 |
|---|---|---|---|---|---|---|---|

5. Traverse a Tree

❖ Practice

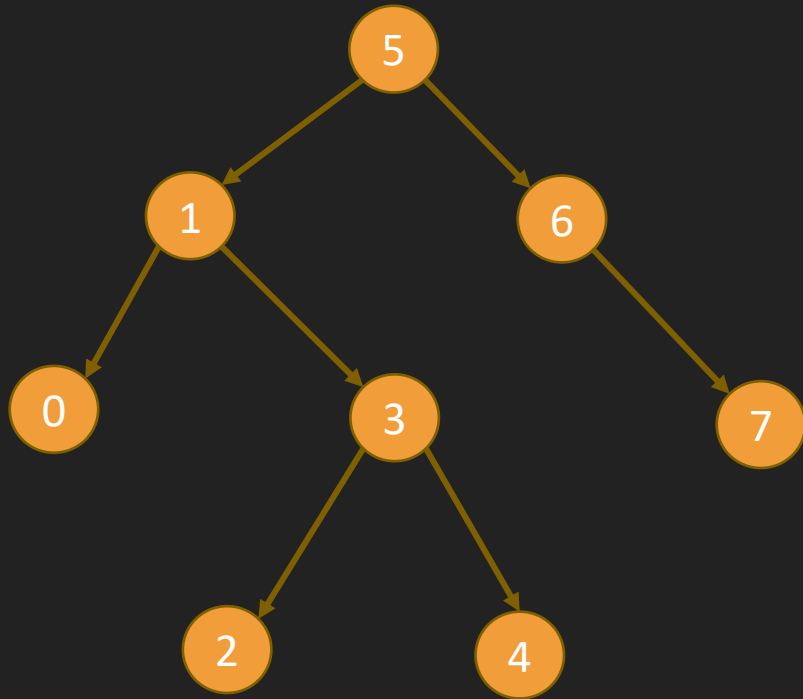
- ✓ 144. Binary Tree Preorder Traversal
- ✓ 94. Binary Tree Inorder Traversal (**Homework**)
- ✓ 145. Binary Tree Postorder Traversal (**Homework**)
- ✓ 102. Binary Tree Level Order Traversal (**Homework**)

6. Working with Binary Tree

✓ **104.** Maximum Depth of Binary Tree

✓ **112.** Path Sum

7. Binary Search Tree



❖ **Question:** Print in ascending order?

❖ **Practice:**

✓ **700.** Search in a Binary Search Tree

✓ **701.** Insert into a Binary Search Tree (**Homework**)

✓ **450.** Delete Node in a BST (**Homework**)

✓ **173.** Binary Search Tree Iterator (**Homework**)

✓ **98.** Validate Binary Search Tree (**Homework**)

Data Structure & Algorithm



Please Like and Subscribe