CIS 263 Introduction to Data Structures and Algorithms

Binary Search Tree (BST)

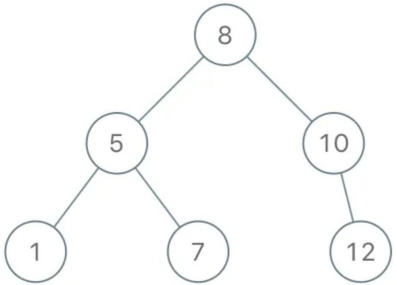
Outline

- BST Traversal
- Deletion

BST - Traversal

- Pre Order
- In Order
- Post Order

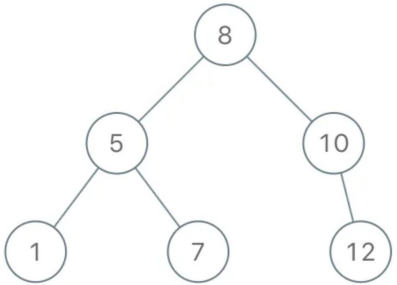
BST - Traversal




```python
# Pre Order Traversal
def pre_order(self, root):
  if root:
    print(root.data)
    self.pre_order(root.left_child)
    self.pre_order(root.right_child)
```

8, 5, 1, 7, 10, 12

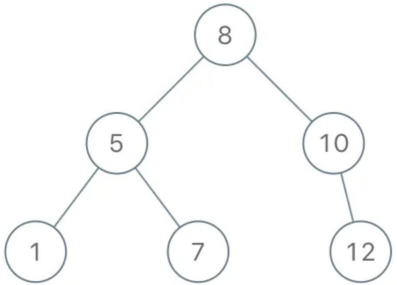
BST - Traversal




```
# In Order Traversal
def in_order(self, root):
  if root:
    self.in_order(root.left_child)
    print(root.data)
    self.in_order(root.right_child)
```

1, 5, 7, 8, 10, 12

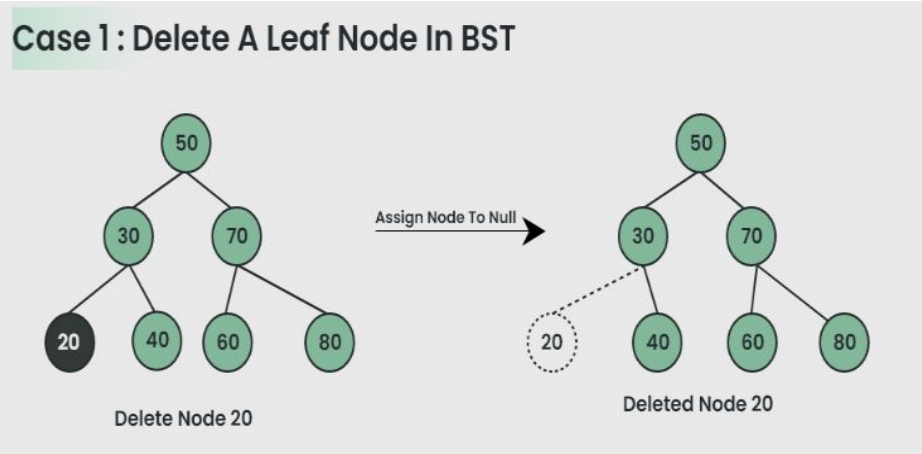
BST - Traversal




```python
# Post Order Traversal
def post_order(self, root):
  if root:
    self.post_order(root.left_child)
    self.post_order(root.right_child)
    print(root.data)
```

1, 7, 5, 12, 10, 8

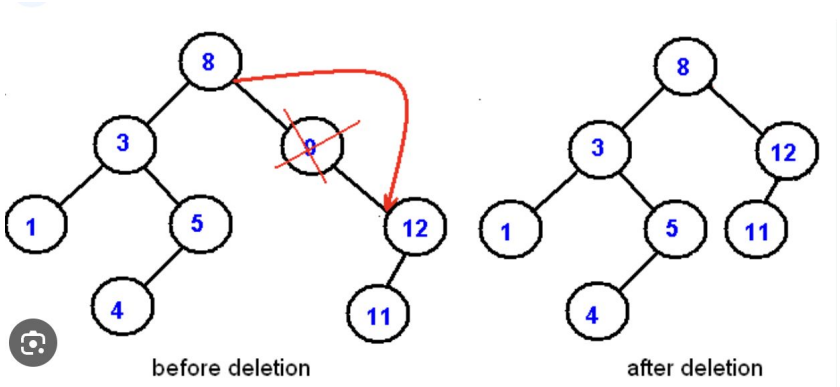
BST - Deletion



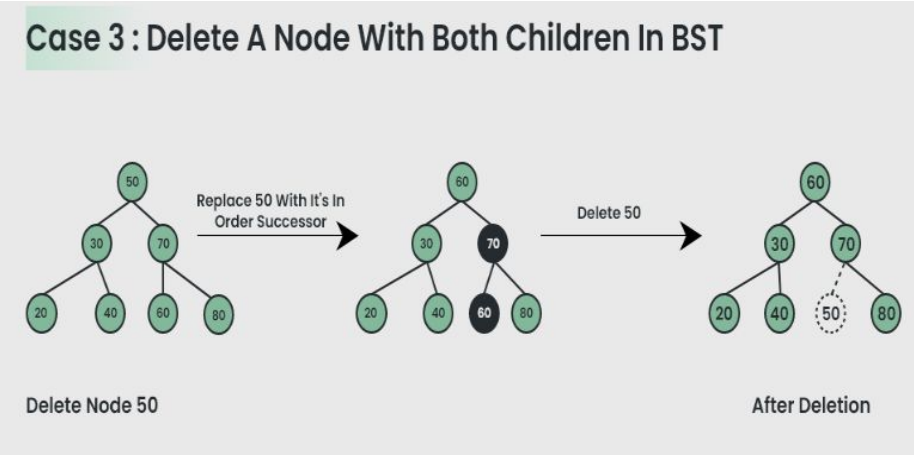
https://www.geeksforgeeks.org/

BST - Deletion

A node with a single child



BST - Deletion



Find out the minimum value in the right child of the node

https://www.geeksforgeeks.org/