**SOFTWARE ENGENEERING**

**5 BCA B**

**"Practical - LAB 3"**

***BY***

**"Tharan" (23215134)**

**SUBMITTED TO**

**meghavi**

****

**SCHOOL OF SCIENCES**

**2025-2026**

## **QUESTION 1**

Task  
1. Traverse the tree using DFS  
2. Showcase the process of searching element in a graph.  
3. Take input from the user for searching an element.

### **Code Solution**

class Node:  
 def \_\_init\_\_(self, value):  
 self.value = value  
 self.left = None  
 self.right = None  
  
def dfs(node, target, path=None):  
 if path is None:  
 path = []  
   
 if node is None:  
 return None, []  
   
 path.append(node.value)  
   
 if node.value == target:  
 return node, path  
   
 left\_result, left\_path = dfs(node.left, target, path.copy())  
 if left\_result:  
 return left\_result, left\_path  
   
 right\_result, right\_path = dfs(node.right, target, path.copy())  
 if right\_result:  
 return right\_result, right\_path  
   
 return None, []  
  
root = Node(1)  
root.left = Node(2)  
root.right = Node(3)  
root.left.left = Node(4)  
root.left.right = Node(5)  
root.right.left = Node(6)  
root.right.right = Node(7)  
  
target = 3  
found\_node, search\_path = dfs(root, target)  
  
if found\_node:  
 print(f"Path to {target}: {search\_path}")  
else:  
 print(f"Element {target} not found in the tree")

### **FINAL Output**

