

BÀI TẬP THỰC HÀNH CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT

LAB MANUAL

Academic Year : 2022 - 2023

Semester : III Semester

Prepared by

MSc. KhangVQH

S. No.	Experiment
1	SEARCHING TECHNIQUES
2	SORTING TECHNIQUES
3	SORTING TECHNIQUES
4	IMPLEMENTATION OF STACK AND QUEUE
5	APPLICATIONS OF STACK
6	IMPLEMENTATION OF SINGLE LINKED LIST
7	IMPLEMENTATION OF DOUBLE LINKED LIST
8	IMPLEMENTATION OF STACK USING LINKED LIST
9	IMPLEMENTATION OF QUEUE USING LINKED LIST
10	IMPLEMENTATION OF BINARY SEARCH TREE

WEEK-10

IMPLEMENTATION OF BINARY SEARCH TREE

10.1 OBJECTIVE:

To write a C program to implement binary search tree creation, traversal and count node.

10.2 PROGRAM LOGIC:

1. The left sub tree of a node contains smaller nodes than a root node.
2. The right sub tree of a node contains greater nodes than a root node.
3. Both the left and right sub trees must also be binary search trees.
4. There are three types of tree traversals: Preorder, Postorder, and Inorder.

Pre-order traversal

Algorithm:

1. Visit the root (we will print it when we visit to show the order of visiting)
2. Traverse the left subtree in pre-order
3. Traverse the right subtree in pre-order

In-order traversal

Visit the root node in between the left and right node (in)

Algorithm:

1. Traverse the left subtree in in-order
2. Visit the root (we will print it when we visit to show the order of visiting)
3. Traverse the right subtree in in-order

Post-order traversal

Visit the root node after (post) visiting the left and right subtree.

Algorithm:

1. Traverse the left subtree in in-order
2. Traverse the right subtree in in-order
3. Visit the root (we will print it when we visit to show the order of visiting)

Maximum depth or Height of a tree

Algorithm:

maxDepth()

1. If tree is empty then return 0
2. Else
 - (a) Get the max depth of left subtree recursively i.e., call maxDepth(tree->left-subtree)
 - (a) Get the max depth of right subtree recursively i.e., call maxDepth(tree->right-subtree)
 - (c) Get the max of

max depths of left and right subtrees and add 1 to it
 for the current node. $\text{max_depth} = \max(\text{max dept of left subtree, max depth of right subtree}) + 1$
 (d) Return max_depth

Count number of leaf nodes in a binary tree

A node is a leaf node if both left and right child nodes of it are NULL.

Algorithm

getLeafCount(node)

1) If node is NULL then return 0.

2) Else If left and right child nodes are NULL return 1.

3) Else recursively calculate leaf count of the tree using below formula.

Leaf count of a tree = Leaf count of left sub tree + leaf count of right sub tree

10.3 IMPLEMENTATION:

Input/Output:

arr = [8,3,1,6,4,7,10,14,13]

Breadth-First Traversal

8

3 10

1 6 14

4 7 13

Inorder Traversal

1

3

4

6

7

8

10

13

14

Preorder Traversal

8

3

1

6

4

7

10

14

13

Postorder Traversal

1
4
7
6
3
13
14
10
8

Count the number of nodes in the binary search tree.

10.4 LAB ASSIGNMENT:

1. Formulate a program to create a Binary Tree of integers?
2. Write a recursive program, for traversing a binary tree in preorder, inorder and postorder?

10.5 POST-LAB VIVA QUESTIONS:

1. Write the balance factor of a Binary Tree?
2. What are the data structures used for Binary Trees?
3. Define a Complete Binary Tree?
4. List out the applications of Binary Tree?
5. Define pre-order traversal.
6. Define post-order traversal.
7. Define in-order traversal.