Course: Data Structures and Algorithms Assignment 4

Binary Tree

Dr. Belal Al-Fuhaidi

1- Mark the following statements as true or false.

a. A binary tree must be nonempty.

False. A binary tree can be empty, meaning it has no nodes at all

b. The level of the root node is 0.

True. In a binary tree, the level of the root node is typically considered to be 0. Level numbering starts from 0 for the root node, 1 for its children, and so on.

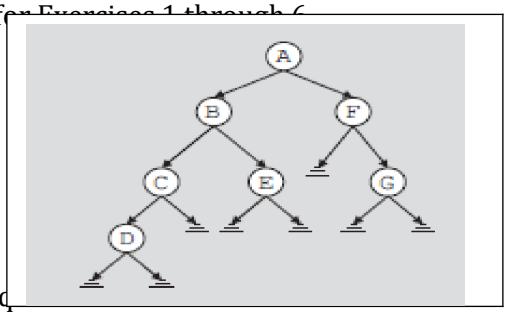
c. If a tree has only one node, the height of this tree is 0 because the number of levels is 0.

True. If a tree has only one node, it can be considered as having a height of 0. The height of a tree is defined as the maximum number of edges in the longest path from the root to a leaf node, and in this case, there are no edges as there is only one node.

d. The inorder traversal of a binary tree always outputs the data in ascending order.

False. The inorder traversal of a binary tree does not necessarily output the data in ascending order. In an inorder traversal, the left subtree is visited first, followed by the root node, and then the right subtree. The order of the nodes depends on the arrangement of the nodes in the tree, and it may or may not result in ascending order.

2- The binary tree of the following Figure is to be used for Exercises 1 through 6



1. Find L_A, the node in the left subtree of A.

- **LA:** هي العقدة في الفرع الأيسر للعقدة A.

2. Find R_A, the node in the right subtree of A.

- **RA:** هي العقدة في الفرع الأيمن للعقدة A.

3. Find R_B, the node in the right subtree of B.

- **Rs:** هي العقدة في الفرع الأيمن للعقدة B.

4. List the nodes of this binary tree in an inorder sequence.

- الترتيب التنازلي: C, A, B, D, E.

5. List the nodes of this binary tree in a preorder sequence.

- الترتيب السابق (Preorder Traversal): A, C, B, D, E.

6. List the nodes of this binary tree in a postorder sequence.

- الترتيب اللاحق (Postorder Traversal): C, B, D, A, E.

3- The binary search tree of the following Figure is to be used for Exercises 1through 4.

1. List the path from the node with info 80 to the node with info 79.

40 52 58 60 79 80

2. A node with info 35 is to be inserted in the tree. List the nodes that are visited by the function insert to insert 35. Redraw the tree after inserting 35.

40 52 58

40 / \ 52 58 / \ / \ 35 60 79 80

3. Delete node 52 and redraw the binary tree.

4. Delete node 40 and redraw the binary tree.

52 / \ 35 58 / \ 60 79 / 80

5. Delete nodes 80 and 58 in that order. Redraw the binary tree after each deletion.

52 / \ 35 60 / \ 79

4- Write the definition of the function, **nodeCount**, that returns the number of nodes in a binary tree.

```
public static int nodeCount(TreeNode root) {
// إذا كانت الشجرة فارغة
  if (root == null) {
    return 0;
  } else {
// عدد العقد في الشجرة الفرعية اليسرى
    int leftCount = nodeCount(root.left);
// عدد العقد في الشجرة الفرعية اليمنى
    int rightCount = nodeCount(root.right);
// عدد العقد في الشجرة بأكملها
    return 1 + leftCount + rightCount;
  }
}
```

5- Write the definition of the function, **leavesCount**, that takes as a parameter a reference to the root node of a binary tree and returns the number of leaves in a binary tree.

```
TreeNode root = new TreeNode(1);
root.left = new TreeNode(2);
root.right = new TreeNode(3);
root.left.left = new TreeNode(4);
root.right.left = new TreeNode(5);

int leaves = leavesCount(root);
System.out.println("Number of leaves: " + leaves);
```

6- Draw the binary tree representation of the following arithmetic expression:
"(((5+2) * (2−1))/((2+9)+((7−2)−1)) *8)".

* / \ / \ * * / \ / \ + 2 / - / \ / \ / \ 5 2 + 9 7 2 / - 1

Binary Tree *Dr. Belal Al-Fuhaidi*

7- Insert, into an empty binary search tree, entries with keys 30, 40, 24, 58, 48, 26, 11, 13 (in this order). Draw the tree after each insertion.

30 / \ 24 40 / \ \ - 48 58 / \ 26 13 / 11

يمكن حل أسئلة من الكتاب , بالإضافة الى المحاضرات

Good Luck