Indian Institute of technology, Guwahati
Department of Computer Science and Engineering
Data Structure Lab: (CS210)

Lab Assignment: 2

Date: 14th August, 2016 **Total Marks: 20**

1. (a) Create a binary tree using **linked list** with inputs given as below format: <**number1 L/R number2**>. You need to insert the number1 in L/R position of number2. All numbers are unique and non-negative. If L/R position of number2 is not available, give a warning as **"number1 is ignored"** without double quotes. Initially tree is empty, you need to make root node containing number1 of first tuple <number1 L/R number2> you encounter ignoring L/R and number2. **[6]**

   1 0 L 10: Initially tree is empty. So, node containing 0 will become root.
   2 1 L 0: insert 1 at the left of 0.
   3 2 R 0: insert 2 at the right of 0.
   4 3 L 0: Since left of 0 is not available, give warning "3 is ignored".
   5 4 L 2: Insert 4 at the left of 2.
   6 -1: end of insertion.

   Except for the first tuple, number2 will always be existing member of tree. You may need to traverse the tree to find the appropriate position for the new node.

   (b) Write a function for **Inorder, Preorder and Postorder** traversal of tree. **[6]**
   (c) Print the **leaf nodes** of the tree. **[8]**

   Input: sequence of <**number1 L/R number2**> .

   Output: Inorder traversal, Preorder traversal, Postorder traversal

   Test1:
   1 L 100
   2 R 1
   3 L 1
   4 L 3
   5 L 4
   7 R 2
   -1

   Output:
   Inorder: 5 4 3 1 2 7
   Preorder: 1 3 4 5 2 7
   Postorder: 5 4 3 7 2 1
   Leaf Nodes: 5 7

Test2:
1 L 0
2 R 1
3 L 1
4 L 2
6 R 2
7 L 3
8 R 3
9 L 3
10 L 4
-1

Output:
9 is ignored
Inorder: 7 3 8 1 10 4 2 6
Preorder: 1 3 7 8 2 4 10 6
Postorder: 7 8 3 10 4 6 2 1
Leaf Nodes: 7 8 10 6

Test3:
2 L 10
11 L 2
12 L 2
15 R 2
17 R 11
18 L 17
19 L 15
13 L 15
21 L 11
-1

Output:
12 is ignored
13 is ignored
Inorder: 21 11 18 17 2 19 15
Preorder: 2 11 21 17 18 15 19
Postorder: 21 18 17 11 19 15 2
Leaf Nodes: 21 18 19

Evaluation Criteria:
1. Do not use any global variable. **Penalty: -4**.
2. All components (a, b, c) MUST be implemented as function**: Penalty: -6 (-2** for each violation)