BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY
Department of Computer Science and Engineering

July 2021 CSE 208 Offline Assignment on Advanced Data Structures-II

AVL Tree

In this assignment, you will have to implement the AVL Tree data structure. You will read the input from a text file. Each line of the input file will specify one of the following operations: Insert (I), Delete (D), or Find (F), followed by an integer specifying the key the operation will work on. After insertion or deletion[1], you need to check if the height balance property is maintained or not and report if the height balance property needs to be fixed. You will have to print the tree in nested parenthesis format[2] as well. For example, the sample binary search tree in Figure 1 will be written as **8(3(1)(6(4)(7)))(10()(14(13)()))**. For the Find operation, print only **True** or **False** based on the query and state of the tree.

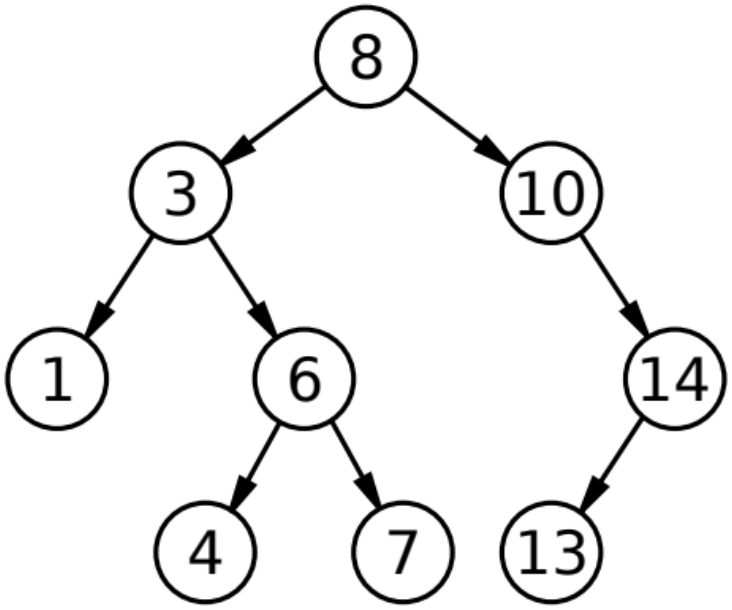

Figure 1: Sample Binary Search Tree

A sample input sequence and its corresponding output are given in table 1.

While you are encouraged to talk to your peers, ask help from teachers, and search relevant resources from online, under no circumstances should you copy code from any source. If found out, you will receive a full 100% penalty.

[1] You may assume you will not be given a delete operation on a node that is not present in the tree.
[2] Root node followed by the left sub-tree and right sub-tree encapsulated in parentheses.

Input[3]		**Output**
F	1	False
I	1	1
I	2	1()(2)
I	3	Height invariant violated. After rebalancing: 2(1)(3)
F	2	True
I	4	2(1)(3()(4))
I	5	Height invariant violated. After rebalancing: 2(1)(4(3)(5))
I	6	Height invariant violated. After rebalancing: 4(2(1)(3))(5()(6))
D	5	4(2(1)(3))(6)
F	5	False
D	6	Height invariant violated. After rebalancing: 2(1)(4(3)())
I	0	2(1(0)())(4(3)())

Table 1: A sample input and corresponding output

[3] Note that, in Table 1, there are some blank lines in the sample input to match the corresponding output lines. However, there will not be any blank lines in the test input files.

Submission Guideline:

1. Create a directory with your 7 digit student id as its name
2. Put the source files only into the directory created in 1
3. Zip the directory
4. Upload the zip into moodle

For example, if your student id is 1805123, create a directory named 1805123. Put your source files(.c, .cpp, .java, .py, .h, .hpp etc) only into 1805123. Zip 1805123 into 1805123.zip and upload the 1805123.zip into moodle.

Failure to follow the above-mentioned submission guideline will result in up to 10% penalty.

Submission Deadline:

February 22, 2022, 11:45 PM