

Lab II

29. Jan. 2016

Implement the class AVLNode which represents a binary tree node having an integer value, references to the left and right child, and the height. Using AVLNode, implement the class AVL representing an AVL tree that will be composed of multiple AVLNodes.

In the AVL class, implement the insert method, which takes as input an integer value and adds it to the tree maintaining the AVL tree structure. The insert method should handle balancing the tree by rotations based on the left left case, left right case, right right case, and right left case that were discussed in class.

Solve the following problems after finishing the AVL insert implementation.

Problem 1

Given a sequence of integers, insert them into a binary search tree then print the tree using pre-order traversal.

Input

Your program will be tested against multiple test cases. Each test case is made up of two lines. The first line contains an integer N representing the number of integers. The second line has N integers to be inserted into the tree.

Output

For each test case, print the tree using pre-order traversal.

Sample Input

7
25 13 10 30 15 27 37

4
6 7 8 9

6

Sample Output

25 12 10 15 30 27 37

7 6 8 9

10 6 4 7 15 13

10 7 15 13 4 6

Problem 2

Given a sequence of integers and a value k , insert them into an AVL tree then search for k in the tree. If k is in the tree, print “found”, else print “not found”.

Input

Your program will be tested against multiple test cases. Each test case is made up of three lines. The first line contains an integer N representing the number of integers. The second line has N integers to be inserted into the tree. The third line contains an integer k representing the value to search for.

Output

For each test case, print “found” if k is in the tree, else print “not found”.

Sample Input

7
25 13 10 30 15 27 37
52

4
6 7 8 9
9

6
10 7 15 13 4 6
15

Sample Output

not found

found

found