Insert Nodes into a Binary Tree from a File and Delete nodes from a Binary Tree

Lab 35C

Your task on this assignment will be to insert the items from the text file into a binary tree. Please check in jGrasp that the trees have some degree of balance to them and that they don’t just turn into linked lists.

Once you have a somewhat balanced tree you will need to write the code to delete a node. Please make sure you can delete each of the following nodes (a leave, the root and a node of a branch with several parents and children – test, test and test more).

A data.txt file is provided for your processing. Here is some code to get you started …

import java.io.\*;  
import java.util.Scanner;  
import java.util.Random;  
  
public class Ch\_35\_Lab\_C {  
 static TreeNode root;  
  
 public static void main(String[] args) throws IOException  
 {  
 root = null;  
 Scanner input = new Scanner(System.in);  
 Random rInt = new Random();  
  
 System.out.print("Enter the number of integers you want: ");  
 int numInt = input.nextInt()-1;  
 //There will probably be less  
 //Since the tree can't have duplicates  
 //-1 because the root is assigned before the loop  
 System.out.println();  
  
 System.out.print("Enter the lowest value for a random int: ");  
 int minVal = input.nextInt();  
 System.out.println();  
  
 System.out.print("Enter the highest value for a random int: ");  
 int maxVal = input.nextInt();  
 System.out.println();  
  
 int rVal = rInt.nextInt(maxVal) + minVal;  
 root = new TreeNode(rVal, null, null);  
 for(int i = 0; i < numInt; i++)  
 {  
 rVal = rInt.nextInt(maxVal) + minVal;  
 insertNode(root, new TreeNode(rVal, null, null));  
 }  
  
 System.out.print("Would you like to see the in-order traversal of the list? Y/N: ");  
 String answer = input.nextLine();  
 answer = input.nextLine();// dummy value  
 System.out.println();  
  
 if(answer.equals("Y") || answer.equals("y"))  
 inOrderTraversal(root);  
 System.out.println();  
 System.out.println();  
  
 System.out.print("Enter the value you would like to delete: ");  
 int searchVal = input.nextInt();  
 System.out.println();  
 TreeNode del = searchNode(searchVal, root);  
 if(del == null)  
 {  
 System.out.println("There is no TreeNode with that value.");  
 System.out.println();  
 }  
 else  
 deleteNode(del);  
  
 System.out.print("Would you like to see the in-order traversal of the list? Y/N: ");  
 answer = input.nextLine();  
 answer = input.nextLine();//For the double thingy  
 System.out.println();  
  
 if(answer.equals("Y") || answer.equals("y"))  
 inOrderTraversal(root);  
 }  
  
 public static void getList() throws IOException  
 {//If we want to run it from a data file  
 BufferedReader file = new BufferedReader(new FileReader(new File("data.txt")));  
 String value;  
 while((value = file.readLine()) != null)  
 {  
 if(root == null)  
 root = new TreeNode(Integer.parseInt(value), null, null);  
 else  
 insertNode(root, new TreeNode(Integer.parseInt(value), null, null));  
 }  
 }

public static void insertNode(TreeNode current, TreeNode p)

{

}

public static void deleteNode(TreeNode current)

{

}

public static TreeNode searchNode(int value, TreeNode current)  
 {  
}

public static void inOrderTraversal(TreeNode p)  
 {  
 if(p != null)  
 {  
 inOrderTraversal(p.getLeft());  
 System.out.print(p.getValue() + " ");  
 inOrderTraversal(p.getRight());  
 }  
 }

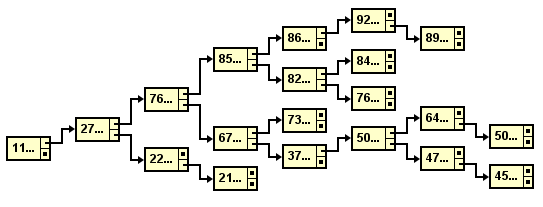
}

class TreeNode  
{  
 private TreeNode left, right;  
 private int value;  
 public TreeNode(int v, TreeNode l, TreeNode r)  
 {  
 value = v;  
 left = l;  
 right = r;  
 }  
  
 public TreeNode getLeft()  
 {  
 return left;  
 }  
  
 public TreeNode getRight()  
 {  
 return right;  
 }  
  
 public int getValue()  
 {  
 return value;  
 }  
  
 public void setLeft(TreeNode p)  
 {  
 left = p;  
 }  
  
 public void setRight(TreeNode p)  
 {  
 right = p;  
 }  
}

Here is the 60 Point version

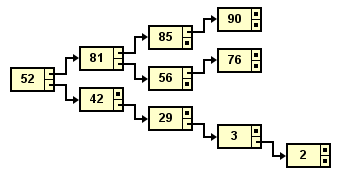
Read in the data.txt file and take a screenshot showing that it is a balanced tree. It must not produce a linked list type of tree. You must include this screenshot with later versions (i.e. 100 pt). \*Note: only the first two digits of the numbers in the text file are displayed.

-------------------------Run 1 ------------------------------



Here is the 70 Point version (Demonstrate an InOrder Traversal with screenshots of your tree)

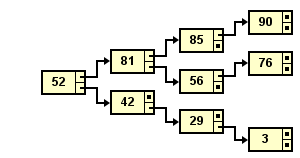
--------------------Run 2 --------------------  
Enter the number of integers you want: 10  
Enter the lowest value for a random int: 1  
Enter the highest value for a random int: 100  
Would you like to see the in-order traversal of the list? Y/N: Y  
2 3 29 42 52 56 76 81 85 90



Here is the 80 Point version (Demonstrate an InOrder Traversal with the leaf “2” removed, include a screenshot of your tree)

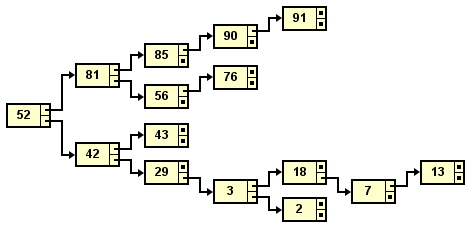
--------------------Run 3--------------------  
Enter the number of integers you want: 10  
Enter the lowest value for a random int: 1  
Enter the highest value for a random int: 100

Would you like to see the in-order traversal of the list? Y/N: Y  
2 3 29 42 52 56 76 81 85 90   
Enter the value you would like to delete: 2  
Would you like to see the in-order traversal of the list? Y/N: Y  
3 29 42 52 56 76 81 85 90

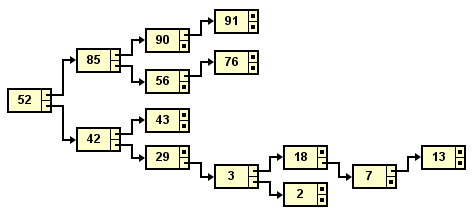


Here is the 90 Point version (Demonstrate an InOrder Traversal with node “81 removed, include a before and after screenshot of your tree)  
--------------------Run 4 --------------------

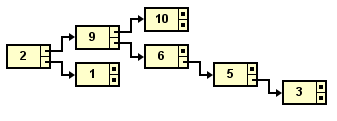
Enter the number of integers you want: 15  
Enter the lowest value for a random int: 1  
Enter the highest value for a random int: 100  
Would you like to see the in-order traversal of the list? Y/N: Y  
2 3 7 13 18 29 42 43 52 56 76 81 85 90 91



Enter the value you would like to delete: 81  
Would you like to see the in-order traversal of the list? Y/N: Y  
2 3 7 13 18 29 42 43 52 56 76 85 90 91



Here is the 100 Point version (Demonstrate an InOrder Traversal with removal of the root node, include a before and after screenshot of your tree)

--------------------Run 5 --------------------  
  
Enter the number of integers you want: 10  
Enter the lowest value for a random int: 1  
Enter the highest value for a random int: 10  
Would you like to see the in-order traversal of the list? Y/N: Y  
1 2 3 5 6 9 10   
  


Enter the value you would like to delete: 2  
Would you like to see the in-order traversal of the list? Y/N: y  
1 3 5 6 9 10

