static Node lca(Node root, int v1, int v2) {

LinkedList<Node> path1 = findPath(root, v1);

if (v1 == v2) { return path1.getLast(); }

LinkedList<Node> path2 = findPath(root, v2);

pSize = (path1.size() < path2.size()) ? path1.size() : path2.size();

for (int i = 0; i < pSize; i++) {

if (path1.get(i) != path2.get(i)) {

return path1.get(i-1);

}

}

return path1.get(pSize - 1);

}

static LinkedList<Node> findPath (Node root, int value) {

LinkedList<Node> path = new LinkedList<Node>();

Node curr = root;

while (curr != null) {

path.add(curr);

if (value == curr.data) { return path; }

else if (value < curr.data) { curr = curr.left; }

else { curr = curr.right; }

}

return null;

}