private static boolean satisfiesHeapOrdering(BinaryTree<Integer> t) {

BinaryTree<Integer> left = t.newInstance();

BinaryTree<Integer> right = t.newInstance();

boolean result = true;

if (t.size() != 0) {

int root = t.disassemble(left, right);

if (left.size() > 0 && root < left.root()) {

result = false;

}

if (right.size() > 0 && root < right.root()) {

result = false;

}

if (result) {

result = satisfiesHeapOrdering(left) && satisfiesHeapOrdering(right);

}

t.assemble(root, left, right);

}

return result;

}