Homework 12



public static <T> int size(BinaryTree<T> t) {

BinaryTree<T> left = new BinaryTree1L<T>;

BinaryTree<T> right = new BinaryTree1L<T>;

int sizeL = 0;

int sizeR = 0;

t.disassemble(left, right);

if (left != null) {

sizeL = size(left);

}

if (right != null) {

sizeR = size(right);

}

return sizeL + sizeR + 1;

}

Same thing but iterative

public static <T> int size(BinaryTree<T> t) {

BinaryTree<T> left = new BinaryTree1L<T>;

BinaryTree<T> right = new BinaryTree1L<T>;

BinaryTree<T> temp = t;

int sizeL = 0;

int sizeR = 0;

t.disassemble(left, right);

while (left != null) {

sizeL = size(left);

left.disassemble(

}

temp.disassemble(left, right);

while (right != null) {

sizeR = size(right);

}

return sizeL + sizeR + 1;

}