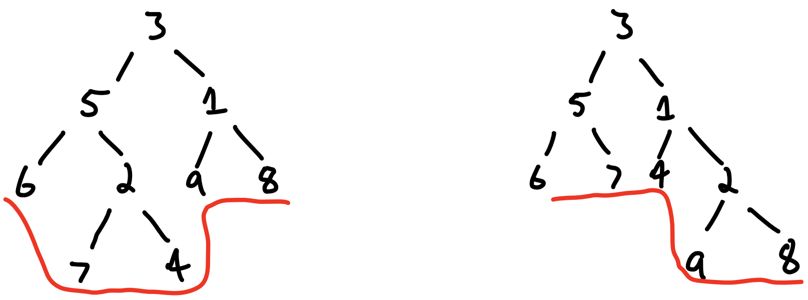
872. Leaf Similar Trees

Consider all the leaves of a binary tree.

Given two binary trees, return true if both have the same sequence of leaves.

tree 1 tree 2



1.) Create 2 lists to store respective leaf nodes of tree1 and tree2.

2.) Enumerate through each tree. If right & left branches are empty, append to respective list.

Could be combined

3.) If size is different, return false.

4.) If two lists are not equal, return false. Otherwise, true.

```
bool similar( TreeNode* root1, TreeNode* root2) {

    std::vector<int> values1;
    std::vector<int> values2;

    recurse( root1, values1);
    recurse( root2, values2);

    if ( values1.size() != values2.size()) {
        return false;
    }

    return std::equal( values1.begin(), value1.end(), values2.begin());

}

void recurse ( TreeNode* node , std::vector<int>& values) {

    if ( node == nullptr ) {
        return;
    }

    recurse( node ->left, values);
    recurse( node-> right, values);
    if ( node-> left == nullptr && node-> right == nullptr) {
        values.push_back( node->val);
    }

}
```