/\*\*

\* Definition for a binary tree node.

\* struct TreeNode {

\* int val;

\* TreeNode \*left;

\* TreeNode \*right;

\* TreeNode(int x) : val(x), left(NULL), right(NULL) {}

\* };

\*/

class Solution {

public:

int findIdx(vector<int>&inorder, int is, int ie, int key){

int i;

for(i=is;i<=ie;i++){

if(inorder[i] == key)

break;

}

return i;

}

TreeNode\* solve(vector<int>&inorder, int is, int ie, vector<int>&preorder,int ps){

//Terminating case -

if(ps<0 || is>ie)return NULL;

TreeNode\* root = new TreeNode(preorder[ps]);

int idx = findIdx(inorder, is, ie, preorder[ps]);

root->left = solve(inorder, is, idx-1, preorder, ps+1);

root->right= solve(inorder, idx+1, ie, preorder, ps+idx-is+1);

return root;

}

TreeNode\* buildTree(vector<int>& preorder, vector<int>& inorder) {

//Corner cases -

if(preorder.size() == 0)return NULL;

//General cases -

return solve(inorder, 0, inorder.size()-1, preorder, 0);

}

};