

145. Binary Tree Postorder Traversal

/*

Morris: adapted to post order traversal

Time complexity: $O(n)$

Space complexity: $O(1)$

n: #nodes in the binary tree

*/

typedef std::vector<int> vi;

class Solution {

public:

vi postorderTraversal(TreeNode* root) {

vi ans;

TreeNode* cur=root;

while (cur){

TreeNode* ptr = cur->right;

if (!ptr) {

ans.push_back(cur->val);

cur=cur->left;

}

else {

while (ptr->left && ptr->left!=cur) ptr=ptr->left;

if (!ptr->left) {

ans.push_back(cur->val);

ptr->left = cur;

cur=cur->right;

}

else {

ptr->left = NULL;

cur=cur->left;

}

}

}

reverse(ans.begin(), ans.end());

return ans;

}

};