

Modification Question I

Binary Tree Operations

Problem

Given an inorder and a preorder traversal of a binary tree, construct the unique binary tree T corresponding to these traversals. The binary tree contains unique non-negative integers as keys. The program should include the following functions:

- (a) **main()**: Repeatedly reads an input character from the menu list through the terminal and executes menu driven operations accordingly.
- (b) **PostOrder(T)**: Prints the post order traversal of the binary tree T .
- (c) **LeafSum(T)**: Prints the sum of the values of leaf nodes. A leaf node is a node with no children.

Input Format

1. The first line contains an integer $n \in [1, 10^6]$ indicating the number of nodes in the tree.
2. The second line contains a space-separated sequence of n integers representing the PREORDER traversal of the tree T with key values $\in [1, 10^6]$.
3. The third line contains a space-separated sequence of n integers representing the INORDER traversal of the tree T with key values $\in [1, 10^6]$.
4. Each subsequent line contains a character from the set $\{'p', 's', 'e'\}$.
 - Character 'p' calls **Postorder(T)** - to print the postorder traversal of the tree.
 - Character 's' calls **LeafSum(T)** - to print the sum of the values of all leaf nodes.
 - Input 'e' terminates the execution of the program.

Output Format

The output of each command should be printed on a separate line.

- For option 'p', print the postorder traversal of T . Each node's value is separated by a space.
- For option 's', print the sum of the values of all leaf nodes.

Testcase

Input1:

5
1 2 3 4 5
5 4 3 2 1
p
s
e

Output:

5 4 3 2 1
5

Input2:

6
4 2 1 3 6 5
1 2 4 3 5 6
p
s
e

Output:

1 2 5 6 3 4
6