You are given two **0-indexed** integer permutations A and B of length n.

A **prefix common array** of A and B is an array C such that C[i] is equal to the count of numbers that are present at or before the index i in both A and B.

Return *the* ***prefix common array*** *of* A *and* B.

A sequence of n integers is called a **permutation** if it contains all integers from 1 to n exactly once.

**Example 1:**

Input: A = [1,3,2,4], B = [3,1,2,4]  
Output: [0,2,3,4]  
Explanation: At i = 0: no number is common, so C[0] = 0.  
At i = 1: 1 and 3 are common in A and B, so C[1] = 2.  
At i = 2: 1, 2, and 3 are common in A and B, so C[2] = 3.  
At i = 3: 1, 2, 3, and 4 are common in A and B, so C[3] = 4.

**Example 2:**

Input: A = [2,3,1], B = [3,1,2]  
Output: [0,1,3]  
Explanation: At i = 0: no number is common, so C[0] = 0.  
At i = 1: only 3 is common in A and B, so C[1] = 1.  
At i = 2: 1, 2, and 3 are common in A and B, so C[2] = 3.

**Constraints:**

* 1 <= A.length == B.length == n <= 50
* 1 <= A[i], B[i] <= n
* It is guaranteed that A and B are both a permutation of n integers.