Max sum path in two arrays

Given two **sorted** arrays of **distinct** integers **arr1** and **arr2**. Each array may have some elements in common with the other array. Find the **maximum sum of a path** from the beginning of any array to the end of any array. You can switch from one array to another array only at the common elements.

Note: When we switch from one array to other, we need to consider the common element only once in the result.

Examples :

Input: arr1 = [2, 3, 7, 10, 12] , arr2 = [1, 5, 7, 8]
Output: 35
Explanation: The path will be 1+5+7+10+12 = 35, where 1 and 5 come from arr2 and then 7 is common so we switch to arr1 and add 10 and 12.

Input: arr1 = [1, 2, 3] , arr2[] = [3, 4, 5]
Output: 15
Explanation: The path will be 1+2+3+4+5=15.

Expected Time Complexity: O(m + n)
Expected Auxiliary Space: O(1)

Constraints:
1 <= arr1.size(), arr2.size() <= 10^4
1 <= arr1[i], arr2[i] <= 10^5