

Problem 3424: Minimum Cost to Make Arrays Identical

Problem Information

Difficulty: Medium

Acceptance Rate: 37.41%

Paid Only: No

Tags: Array, Greedy, Sorting

Problem Description

You are given two integer arrays `arr` and `brr` of length `n`, and an integer `k`. You can perform the following operations on `arr` _any_ number of times:

- * Split `arr` into _any_ number of **contiguous** subarrays and rearrange these subarrays in _any order_. This operation has a fixed cost of `k`.
- * Choose any element in `arr` and add or subtract a positive integer `x` to it. The cost of this operation is `x`.

Return the **minimum** total cost to make `arr` **equal** to `brr`.

Example 1:

Input: arr = [-7,9,5], brr = [7,-2,-5], k = 2

Output: 13

Explanation:

- * Split `arr` into two contiguous subarrays: `[-7]` and `[9, 5]` and rearrange them as `[9, 5, -7]`, with a cost of 2.
- * Subtract 2 from element `arr[0]`. The array becomes `[7, 5, -7]`. The cost of this operation is 2.
- * Subtract 7 from element `arr[1]`. The array becomes `[7, -2, -7]`. The cost of this operation is 7.
- * Add 2 to element `arr[2]`. The array becomes `[7, -2, -5]`. The cost of this operation is 2.

The total cost to make the arrays equal is `2 + 2 + 7 + 2 = 13`.

****Example 2:****

****Input:**** arr = [2,1], brr = [2,1], k = 0

****Output:**** 0

****Explanation:****

Since the arrays are already equal, no operations are needed, and the total cost is 0.

****Constraints:****

```
* `1 <= arr.length == brr.length <= 105` * `0 <= k <= 2 * 1010` * `-105 <= arr[i] <= 105` * `-105 <= brr[i] <= 105`
```

Code Snippets

C++:

```
class Solution {  
public:  
    long long minCost(vector<int>& arr, vector<int>& brr, long long k) {  
  
    }  
};
```

Java:

```
class Solution {  
public long minCost(int[] arr, int[] brr, long k) {  
  
}  
}
```

Python3:

```
class Solution:  
    def minCost(self, arr: List[int], brr: List[int], k: int) -> int:
```