

https://leetcode.com/problems/minimum-cost-to-make-array-equal/description/

LeetCode

Problem List

2448. Minimum Cost to Make Array Equal

Hard

516

10

Companies

You are given two 0-indexed arrays `nums` and `cost` consisting each of `n` positive integers.

You can do the following operation **any** number of times:

- Increase or decrease **any** element of the array `nums` by 1.

The cost of doing one operation on the  $i^{\text{th}}$  element is `cost[i]`.

Return the **minimum** total cost such that all the elements of the array `nums` become **equal**.

**Example 1:**

Input: `nums = [1,3,5,2], cost = [2,3,1,14]`  
Output: 8  
Explanation: We can make all the elements equal to 2 in the

```

1 class Solution:
2     def minCost(self, nums: List[int], cost: List[int]) -> int:
3         l,r=min(nums),max(nums)
4         def f(m):
5             return sum([abs(i-m)*j for i,j in zip(nums,cost)])
6         while l<r:
7             m=(l+r)//2
8             a,b,c=f(m-1),f(m),f(m+1)
9             if a>b<c:
10                return b
11             elif a>b:
12                 l=m+1
13             else:
14                 r=m-1

```

Console

Run

Submit

leetcode.com/problems/minimum-number-of-operations-to-make-array-continuous/

LeetCode

Problem List

2099. Minimum Number of Operations to Make Array Continuous

Hard

643

7

Companies

You are given an integer array `nums`. In one operation, you can replace **any** element in `nums` with **any** integer.

`nums` is considered **continuous** if both of the following conditions are fulfilled:

- All elements in `nums` are **unique**.
- The difference between the **maximum** element and the **minimum** element in `nums` equals `nums.length - 1`.

For example, `nums = [4, 2, 5, 3]` is **continuous**, but `nums = [1, 2, 3, 5, 6]` is **not continuous**.

Return the **minimum** number of operations to make `nums` **continuous**.

```

1 import sys
2 class Solution:
3     def minOperations(self, nums: List[int]) -> int:
4         u=len(nums)
5         arr=sorted(set(nums))
6         n=len(arr)
7         out=sys.maxsize
8         for i,j in enumerate(arr):
9             l=i
10            r=n-1
11            se=j+u-1
12            ans=i
13            while l<r:
14                m=(l+r)//2
15                if arr[m]<=se:
16                    ans=m
17                    l=m+1
18                else:
19                    r=m-1
20            out=min(out,u-(ans-i+1))
21        return out

```

Console

Run

Submit