

高度检查器

学校在拍年度纪念照时，一般要求学生按照 **非递减** 的高度顺序排列。

请你返回能让所有学生以 **非递减** 高度排列的最小必要移动人数。

注意，当一组学生被选中时，他们之间可以以任何可能的方式重新排序，而未被选中的学生应该保持不动。

示例：

输入：heights = [1,1,4,2,1,3]

输出：3

解释：

当前数组：[1,1,4,2,1,3]

目标数组：[1,1,1,2,3,4]

在下标 2 处（从 0 开始计数）出现 4 vs 1，所以必须移动这名学生。

在下标 4 处（从 0 开始计数）出现 1 vs 3，所以必须移动这名学生。

在下标 5 处（从 0 开始计数）出现 3 vs 4，所以必须移动这名学生。

示例 2：

输入：heights = [5,1,2,3,4]

输出：5

示例 3：

输入：heights = [1,2,3,4,5]

输出：0

```
class Solution {
public:
    int heightChecker(vector<int>& heights) {
        vector<int> tmp=heights;
        sort(tmp.begin(),tmp.end());
        int sum=0;
        for(int i=0;i<heights.size();i++)
        {
            if(tmp[i]!=heights[i])
```

```
        {  
            sum++;  
        }  
    }  
    return sum;  
}  
};
```