

Problem 1478: Allocate Mailboxes

Problem Information

Difficulty: Hard

Acceptance Rate: 56.12%

Paid Only: No

Tags: Array, Math, Dynamic Programming, Sorting

Problem Description

Given the array `houses` where `houses[i]` is the location of the `i`th house along a street and an integer `k`, allocate `k` mailboxes in the street.

Return the **minimum** total distance between each house and its nearest mailbox.

The test cases are generated so that the answer fits in a 32-bit integer.

Example 1.

 (https://assets.leetcode.com/uploads/2020/05/07/sample_11_1816.png)

Input: houses = [1,4,8,10,20], k = 3 **Output:** 5 **Explanation:** Allocate mailboxes in position 3, 9 and 20. Minimum total distance from each houses to nearest mailboxes is $|3-1| + |4-3| + |9-8| + |10-9| + |20-20| = 5$

Example 2.

 (https://assets.leetcode.com/uploads/2020/05/07/sample_2_1816.png)

Input: houses = [2,3,5,12,18], k = 2 **Output:** 9 **Explanation:** Allocate mailboxes in position 3 and 14. Minimum total distance from each houses to nearest mailboxes is $|2-3| + |3-3| + |5-3| + |12-14| + |18-14| = 9$.

Constraints:

*`1 <= k <= houses.length <= 100` *`1 <= houses[i] <= 104` * All the integers of `houses` are
unique.

Code Snippets

C++:

```
class Solution {  
public:  
    int minDistance(vector<int>& houses, int k) {  
  
    }  
};
```

Java:

```
class Solution {  
    public int minDistance(int[] houses, int k) {  
  
    }  
}
```

Python3:

```
class Solution:  
    def minDistance(self, houses: List[int], k: int) -> int:
```