Problem 910: Smallest Range II

Problem Information

Difficulty: Medium
Acceptance Rate: 37.56%
Paid Only: No
Tags: Array, Math, Greedy, Sorting

Problem Description

You are given an integer array `nums` and an integer `k`.

For each index `i` where `0 <= i < nums.length`, change `nums[i]` to be either `nums[i] + k` or `nums[i] - k`.

The **score** of `nums` is the difference between the maximum and minimum elements in `nums`.

Return _the minimum**score** of _`nums` _after changing the values at each index_.

Example 1:

Input: nums = [1], k = 0 **Output:** 0 **Explanation:** The score is max(nums) - min(nums) = 1 - 1 = 0.

Example 2:

Input: nums = [0,10], k = 2 **Output:** 6 **Explanation:** Change nums to be [2, 8]. The score is max(nums) - min(nums) = 8 - 2 = 6.

Example 3:

Input: nums = [1,3,6], k = 3 **Output:** 3 **Explanation:** Change nums to be [4, 6, 3]. The score is max(nums) - min(nums) = 6 - 3 = 3.

Constraints:

* `1 <= nums.length <= 104` * `0 <= nums[i] <= 104` * `0 <= k <= 104`

Code Snippets

C++:

```
class Solution {
public:
int smallestRangeII(vector<int>& nums, int k) {

}
};
```

Java:

```
class Solution {
public int smallestRangeII(int[] nums, int k) {

}
}
```

Python3:

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