Problem | Editorial | Submissions

Minimize the Heights II

Difficulty: **Medium** Accuracy: **15.06%** Submissions: **649K+** Points: **4**

Given an array **arr[]** denoting heights of **N** towers and a positive integer **K.**

For **each** tower, you must perform **exactly one** of the following operations **exactly once**.

- **Increase** the height of the tower by **K**
- **Decrease** the height of the tower by **K**

Find out the **minimum** possible difference between the height of the shortest and tallest towers after you have modified each tower.

You can find a slight modification of the problem here.
Note: It is **compulsory** to increase or decrease the height by K for each tower. **After** the operation, the resultant array should **not** contain any **negative integers**.

Examples :

Input: k = 2, arr[] = {1, 5, 8, 10}
Output: 5
Explanation: The array can be modified as {1+k, 5-k, 8-k, 10-k} = {3, 3, 6, 8}.The difference between the largest and the smallest is 8-3 = 5

Java (1.8) | Average Time: 25m | Start Timer

```
1  // } Driver Code Ends
34
35
36 // User function Template for Java
37
38 class Solution {
39     int getMinDiff(int[] arr, int k) {
40         int n = arr.length;
41         if (n == 1) return 0;
42         Arrays.sort(arr);
43         int diff = arr[n - 1] - arr[0];
44         int smallest = arr[0] + k;
45         int largest = arr[n - 1] - k;
46         for (int i = 0; i < n - 1; i++) {
47             int minHeight = Math.min(smallest, arr[i + 1] - k);
48             int maxHeight = Math.max(largest, arr[i] + k);
49             if (minHeight < 0) continue;
50             diff = Math.min(diff, maxHeight - minHeight);
51         }
52         return diff;
53     }
54 }
55
56
```

Custom Input | Compile & Run | Submit