Problem 1499: Max Value of Equation

Problem Information

Difficulty: Hard
Acceptance Rate: 44.82%
Paid Only: No
Tags: Array, Queue, Sliding Window, Heap (Priority Queue), Monotonic Queue

Problem Description

You are given an array `points` containing the coordinates of points on a 2D plane, sorted by the x-values, where `points[i] = [xi, yi]` such that `xi < xj` for all `1 <= i < j <= points.length`. You are also given an integer `k`.

Return _the maximum value of the equation_`yi + yj + |xi - xj|` where `|xi - xj| <= k` and `1 <= i < j <= points.length`.

It is guaranteed that there exists at least one pair of points that satisfy the constraint `|xi - xj| <= k`.

Example 1:

Input: points = [[1,3],[2,0],[5,10],[6,-10]], k = 1 **Output:** 4 **Explanation:** The first two points satisfy the condition |xi - xj| <= 1 and if we calculate the equation we get 3 + 0 + |1 - 2| = 4. Third and fourth points also satisfy the condition and give a value of 10 + -10 + |5 - 6| = 1. No other pairs satisfy the condition, so we return the max of 4 and 1.

Example 2:

Input: points = [[0,0],[3,0],[9,2]], k = 3 **Output:** 3 **Explanation:** Only the first two points have an absolute difference of 3 or less in the x-values, and give the value of 0 + 0 + |0 - 3| = 3.

Constraints:

* `2 <= points.length <= 105` * `points[i].length == 2` * `-108 <= xi, yi <= 108` * `0 <= k <= 2 * 108` * `xi < xj` for all `1 <= i < j <= points.length` * `xi` form a strictly increasing sequence.

Code Snippets

C++:

```
class Solution {
public:
int findMaxValueOfEquation(vector<vector<int>>& points, int k) {

}
};
```

Java:

```
class Solution {
public int findMaxValueOfEquation(int[][] points, int k) {

}
}
```

Python3:

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