Problem 1610: Maximum Number of Visible Points

Problem Information

Difficulty: Hard
Acceptance Rate: 37.80%
Paid Only: No
Tags: Array, Math, Geometry, Sliding Window, Sorting

Problem Description

You are given an array `points`, an integer `angle`, and your `location`, where `location = [posx, posy]` and `points[i] = [xi, yi]` both denote **integral coordinates** on the X-Y plane.

Initially, you are facing directly east from your position. You **cannot move** from your position, but you can **rotate**. In other words, `posx` and `posy` cannot be changed. Your field of view in **degrees** is represented by `angle`, determining how wide you can see from any given view direction. Let `d` be the amount in degrees that you rotate counterclockwise. Then, your field of view is the **inclusive** range of angles `[d - angle/2, d + angle/2]`.

Your browser does not support the video tag or this video format.

You can **see** some set of points if, for each point, the **angle** formed by the point, your position, and the immediate east direction from your position is **in your field of view**.

There can be multiple points at one coordinate. There may be points at your location, and you can always see these points regardless of your rotation. Points do not obstruct your vision to other points.

Return _the maximum number of points you can see_.

Example 1:

![](https://assets.leetcode.com/uploads/2020/09/30/89a07e9b-00ab-4967-976a-c723b2aa8656.png)

Input: points = [[2,1],[2,2],[3,3]], angle = 90, location = [1,1] **Output:** 3 **Explanation:** The shaded region represents your field of view. All points can be made visible in your field of view, including [3,3] even though [2,2] is in front and in the same line of sight.

Example 2:

Input: points = [[2,1],[2,2],[3,4],[1,1]], angle = 90, location = [1,1] **Output:** 4 **Explanation:** All points can be made visible in your field of view, including the one at your location.

Example 3:

![](https://assets.leetcode.com/uploads/2020/09/30/5010bfd3-86e6-465f-ac64-e9df941d2e49.png)

Input: points = [[1,0],[2,1]], angle = 13, location = [1,1] **Output:** 1 **Explanation:** You can only see one of the two points, as shown above.

Constraints:

* `1 <= points.length <= 105` * `points[i].length == 2` * `location.length == 2` * `0 <= angle < 360` * `0 <= posx, posy, xi, yi <= 100`

Code Snippets

C++:

```
class Solution {
public:
int visiblePoints(vector<vector<int>>& points, int angle, vector<int>&
location) {

}
};
```

Java:

```
class Solution {
public int visiblePoints(List<List<Integer>> points, int angle, List<Integer>
location) {
}
}
```

Python3:

```
class Solution:
def visiblePoints(self, points: List[List[int]], angle: int, location:
List[int]) -> int:
```