Problem 1779: Find Nearest Point That Has the Same X or Y Coordinate

Problem Information

Difficulty: Easy
Acceptance Rate: 69.71%
Paid Only: No
Tags: Array

Problem Description

You are given two integers, `x` and `y`, which represent your current location on a Cartesian grid: `(x, y)`. You are also given an array `points` where each `points[i] = [ai, bi]` represents that a point exists at `(ai, bi)`. A point is **valid** if it shares the same x-coordinate or the same y-coordinate as your location.

Return _the index**(0-indexed)** of the **valid** point with the smallest **Manhattan distance** from your current location_. If there are multiple, return _the valid point with the**smallest** index_. If there are no valid points, return `-1`.

The **Manhattan distance** between two points `(x1, y1)` and `(x2, y2)` is `abs(x1 - x2) + abs(y1 - y2)`.

Example 1:

Input: x = 3, y = 4, points = [[1,2],[3,1],[2,4],[2,3],[4,4]] **Output:** 2 **Explanation:** Of all the points, only [3,1], [2,4] and [4,4] are valid. Of the valid points, [2,4] and [4,4] have the smallest Manhattan distance from your current location, with a distance of 1. [2,4] has the smallest index, so return 2.

Example 2:

Input: x = 3, y = 4, points = [[3,4]] **Output:** 0 **Explanation:** The answer is allowed to be on the same location as your current location.

Example 3:

Input: x = 3, y = 4, points = [[2,3]] **Output:** -1 **Explanation:** There are no valid points.

Constraints:

* `1 <= points.length <= 104` * `points[i].length == 2` * `1 <= x, y, ai, bi <= 104`

Code Snippets

C++:

```
class Solution {
public:
int nearestValidPoint(int x, int y, vector<vector<int>>& points) {

}
};
```

Java:

```
class Solution {
public int nearestValidPoint(int x, int y, int[][] points) {

}
}
```

Python3:

```
class Solution:
def nearestValidPoint(self, x: int, y: int, points: List[List[int]]) -> int:
```