Problem 2940: Find Building Where Alice and Bob Can Meet

Problem Information

Difficulty: Hard
Acceptance Rate: 52.16%
Paid Only: No
Tags: Array, Binary Search, Stack, Binary Indexed Tree, Segment Tree, Heap (Priority Queue), Monotonic Stack

Problem Description

You are given a **0-indexed** array `heights` of positive integers, where `heights[i]` represents the height of the `ith` building.

If a person is in building `i`, they can move to any other building `j` if and only if `i < j` and `heights[i] < heights[j]`.

You are also given another array `queries` where `queries[i] = [ai, bi]`. On the `ith` query, Alice is in building `ai` while Bob is in building `bi`.

Return _an array_ `ans` _where_ `ans[i]` _is**the index of the leftmost building** where Alice and Bob can meet on the_ `ith` _query_. _If Alice and Bob cannot move to a common building on query_ `i`, _set_ `ans[i]` _to_ `-1`.

Example 1:

Input: heights = [6,4,8,5,2,7], queries = [[0,1],[0,3],[2,4],[3,4],[2,2]] **Output:** [2,5,-1,5,2] **Explanation:** In the first query, Alice and Bob can move to building 2 since heights[0] < heights[2] and heights[1] < heights[2]. In the second query, Alice and Bob can move to building 5 since heights[0] < heights[5] and heights[3] < heights[5]. In the third query, Alice cannot meet Bob since Alice cannot move to any other building. In the fourth query, Alice and Bob can move to building 5 since heights[3] < heights[5] and heights[4] < heights[5]. In the fifth query, Alice and Bob are already in the same building. For ans[i] != -1, It can be shown that ans[i] is the leftmost building where Alice and Bob can meet. For ans[i] == -1, It can be shown that there is no building where Alice and Bob can meet.

Example 2:

Input: heights = [5,3,8,2,6,1,4,6], queries = [[0,7],[3,5],[5,2],[3,0],[1,6]] **Output:** [7,6,-1,4,6] **Explanation:** In the first query, Alice can directly move to Bob's building since heights[0] < heights[7]. In the second query, Alice and Bob can move to building 6 since heights[3] < heights[6] and heights[5] < heights[6]. In the third query, Alice cannot meet Bob since Bob cannot move to any other building. In the fourth query, Alice and Bob can move to building 4 since heights[3] < heights[4] and heights[0] < heights[4]. In the fifth query, Alice can directly move to Bob's building since heights[1] < heights[6]. For ans[i] != -1, It can be shown that ans[i] is the leftmost building where Alice and Bob can meet. For ans[i] == -1, It can be shown that there is no building where Alice and Bob can meet.

Constraints:

* `1 <= heights.length <= 5 * 104` * `1 <= heights[i] <= 109` * `1 <= queries.length <= 5 * 104` * `queries[i] = [ai, bi]` * `0 <= ai, bi <= heights.length - 1`

Code Snippets

C++:

```
class Solution {
public:
vector<int> leftmostBuildingQueries(vector<int>& heights,
vector<vector<int>>& queries) {

}
};
```

Java:

```
class Solution {
public int[] leftmostBuildingQueries(int[] heights, int[][] queries) {

}
}
```

Python3:

```
class Solution:
def leftmostBuildingQueries(self, heights: List[int], queries:
List[List[int]]) -> List[int]:
```