Problem 3285: Find Indices of Stable Mountains

Problem Information

Difficulty: Easy
Acceptance Rate: 86.51%
Paid Only: No
Tags: Array

Problem Description

There are `n` mountains in a row, and each mountain has a height. You are given an integer array `height` where `height[i]` represents the height of mountain `i`, and an integer `threshold`.

A mountain is called **stable** if the mountain just before it (**if it exists**) has a height **strictly greater** than `threshold`. **Note** that mountain 0 is **not** stable.

Return an array containing the indices of _all_ **stable** mountains in **any** order.

Example 1:

Input: height = [1,2,3,4,5], threshold = 2

Output: [3,4]

Explanation:

* Mountain 3 is stable because `height[2] == 3` is greater than `threshold == 2`. * Mountain 4 is stable because `height[3] == 4` is greater than `threshold == 2`.

Example 2:

Input: height = [10,1,10,1,10], threshold = 3

Output: [1,3]

Example 3:

Input: height = [10,1,10,1,10], threshold = 10

Output: []

Constraints:

* `2 <= n == height.length <= 100` * `1 <= height[i] <= 100` * `1 <= threshold <= 100`

Code Snippets

C++:

```
class Solution {
public:
vector<int> stableMountains(vector<int>& height, int threshold) {

}
};
```

Java:

```
class Solution {
public List<Integer> stableMountains(int[] height, int threshold) {

}
}
```

Python3:

```
class Solution:
def stableMountains(self, height: List[int], threshold: int) -> List[int]:
```