665 Non-decreasing Array

2018年4月22日 15:23

Given an array with n integers, your task is to check if it could become non-decreasing by modifying **at most** 1 element.
We define an array is non-decreasing if array[i] <= array[i + 1] holds for every i (1 <= i < n).
Example 1:
Input: [4,2,3]
Output: True
Explanation: You could modify the first 4 to 1 to get a non-decreasing array.
Example 2:
Input: [4,2,1]
Output: False
Explanation: You can't get a non-decreasing array by modify at most one element.
Note: The n belongs to [1, 10,000].

来自 <https://leetcode.com/problems/non-decreasing-array/description/>
给定一个长度为 n 的整数数组，你的任务是判断在**最多**改变 1 个元素的情况下，该数组能否变成一个非递减数列。
我们是这样定义一个非递减数列的： 对于数组中所有的 i (1 <= i < n)，满足 array[i] <= array[i + 1]。

说明： n 的范围为 [1, 10,000]。

Solution for Python3:

```python
class Solution1:
    def checkPossibility(self, nums):
        """
        :type nums: List[int]
        :rtype: bool
        """
        p = None
        for i in range(len(nums) - 1):
           if nums[i] > nums[i + 1] :
               if p is not None:
                   return False
               p = i
        return p == None or p == 0 or p == len(nums) - 2 or nums[p - 1] <= nums[p + 1] or nums[p] <= nums[p + 2]

class Solution2:
    def checkPossibility(self, nums):
        """
        :type nums: List[int]
        :rtype: bool
        """
        cnt = 0
        for i in range(len(nums) - 1):
           if nums[i] > nums[i + 1]:
               cnt += 1
               if cnt > 1:
                   break
               if i - 1 < 0 or nums[i - 1] < nums[i + 1]:
                   nums[i] = nums[i + 1]
               else:
                   nums[i + 1] = nums[i]
        return cnt <= 1
```

Solution for C++:

```cpp
class Solution {
public:
    bool checkPossibility(vector<int>& nums) {
        int cnt = 0;
        for (int i = 1; i < nums.size() && cnt <= 1; i++) {
            if (nums[i - 1] > nums[i]) {
                cnt++;
                if (i - 2 < 0 || nums[i - 2] <= nums[i])
                    nums[i - 1] = nums[i];
                else
                    nums[i] = nums[i - 1];
            }
        }
        return cnt <= 1;
    }
};
```