

有序数组的平方

给你一个按 **非递减顺序** 排序的整数数组 `nums`，返回 **每个数字的平方** 组成的新数组，要求也按 **非递减顺序** 排序。

示例 1:

输入: `nums = [-4,-1,0,3,10]`

输出: `[0,1,9,16,100]`

解释: 平方后, 数组变为 `[16,1,0,9,100]`

排序后, 数组变为 `[0,1,9,16,100]`

示例 2:

输入: `nums = [-7,-3,2,3,11]`

输出: `[4,9,9,49,121]`

```
class Solution:
    def sortedSquares(self, nums: List[int]) -> List[int]:
        ...
        new_num=[]
        for i in nums:
            new_num.append(i*i)
        new_num.sort()
        return new_num
        ...
        for i in range(len(nums)):
            nums[i]=nums[i]**2
        nums.sort()
        return nums
```