# 324. Wiggle Sort II

Given an integer array `nums`, reorder it such that `nums[0] < nums[1] > nums[2] < nums[3]...`.

You may assume the input array always has a valid answer.

**Example 1:**

```
Input: nums = [1,5,1,1,6,4]
Output: [1,6,1,5,1,4]
Explanation: [1,4,1,5,1,6] is also accepted.
```

**Example 2:**

```
Input: nums = [1,3,2,2,3,1]
Output: [2,3,1,3,1,2]
```

**Constraints:**

- `1 <= nums.length <= 5 * 10<sup>4</sup>`
- `0 <= nums[i] <= 5000`
- It is guaranteed that there will be an answer for the given input `nums`.

**Follow Up:** Can you do it in `O(n)` time and/or **in-place** with `O(1)` extra space?

: NO=====

```python
class Solution:
    def wiggleSort(self, nums: List[int]) -> None:
        """
        Do not return anything, modify nums in-place instead.
        """
        ans = [0]*len(nums)
        nums.sort()
        n = len(nums)
        i = 1
        j = len(nums)-1
        while i<n:
            ans[i] = nums[j]
            i = i+2
            j = j-1
        i = 0
        while i<n:
```

```
            ans[i] = nums[j]
            i = i+2
            j = j-1
        for i in range(len(nums)):
            nums[i] = ans[i]
        return nums
```