670. Maximum Swap

You are given an integer `num`. You can swap two digits at most once to get the maximum valued number.

Return *the maximum valued number you can get*.

Example 1:

```
Input: num = 2736
Output: 7236
Explanation: Swap the number 2 and the number 7.
```

Example 2:

```
Input: num = 9973
Output: 9973
Explanation: No swap.
```

Constraints:

- `0 <= num <= 10<sup>8</sup>`
- ```
  class Solution:
      def maximumSwap(self, num: int) -> int:

          number = str(num)
          if len(number)==1:
              return num

          idx = 0
          for i in range(1,len(number)):
              if int(number[i-1])>=int(number[i]):
                  continue
              else:
                  idx = i
                  break

          if i==len(number):
              return num
          maxDigit = 0
          swapId = 0
          for j in range(idx,len(number)):
  ```
  ```

```
            if int(number[j])>=maxDigit:
                maxDigit =int(number[j])
                swapId = j
        number = list(number)
        for i in range(0,idx):
            if int(number[i])<maxDigit:
                number[i], number[swapId] = number[swapId],number[i]
                break
        number = ''.join(number)
        return int(number)
```