

670. 最大交换 (M)

给定一个非负整数，你至多可以交换一次数字中的任意两位。返回你能得到的最大值。

输入：2736 输出：7236

解释：交换数字2和数字7。

输入：9973 输出：9973

解释：不需要交换。

解题思路：

需要交换一个较大的低位到一个较小的高位上，且高位需要尽可能高；

将数字转化为字符串，从右向左按位遍历，记录已遍历的最大值；

当前位上的值小于已遍历的最大值时，说明找到了可以交换的位，记录下来；

遍历到最高位后结束，交换记录的两个位置（如果没有记录，则不交换任何位置）。

//

<https://leetcode-cn.com/problems/maximum-swap/solution/js-tan-xin-by-marswiz-zugh/>

```
var maximumSwap = function(num) {
    num = (num + '').split('').map(i => +i);
    let last = -1, lastMax = -1, max = -1;
    for (let i = num.length-1; i >= 0; i--) {
        if (max !== -1 && num[i] < num[max]) {
            last = i;
            lastMax = max;
        }
        if (max === -1 || num[i] > num[max]) max = i;
    }
    if (last !== -1) {
        [num[last], num[lastMax]] = [num[lastMax], num[last]];
    }
    return +num.join('');
};
```

//

<https://leetcode-cn.com/problems/maximum-swap/solution/js-tan-xin-by-marswiz-zugh/>

```
var maximumSwap = function (num) {
    let last = new Array(10).fill(-1);
    num = Array.from(num.toString());
    //找到相同值最后出现的位置
```

```
for (let i = 0; i < num.length; i++) {  
    last[num[i] - '0'] = i;  
}  
//原数组从左到右遍历，索引数组从后往前遍历  
//遇到比当前位值大的，交换，因为索引数组从后往前遍历的，所以保证了值为最大  
for (let i = 0; i < num.length; i++) {  
    for (let d = 9; d > (num[i] - '0'); d--) {  
        if (last[d] > i) {  
            let temp = num[last[d]];  
            num[last[d]] = num[i];  
            num[i] = temp;  
            return Number(num.join(""));  
        }  
    }  
}  
return Number(num.join(""));  
};  
  
//https://leetcode-cn.com/problems/maximum-swap/solution/2021316-zui-da-jia  
o-huan-quan-chang-zui-ery0x/  
// https://www.youtube.com/watch?v=6hFyHU7BNlI
```

本题的核心要义就是 让最高位的小数和后面的大数交换