

# Problem 2231: Largest Number After Digit Swaps by Parity

## Problem Information

Difficulty: [Easy](#)

Acceptance Rate: 64.55%

Paid Only: No

Tags: Sorting, Heap (Priority Queue)

## Problem Description

You are given a positive integer `num`. You may swap any two digits of `num` that have the same **parity** (i.e. both odd digits or both even digits).

Return **the largest** possible value of `num` after **any** number of swaps.

**Example 1:**

**Input:** `num = 1234` **Output:** `3412` **Explanation:** Swap the digit 3 with the digit 1, this results in the number 3214. Swap the digit 2 with the digit 4, this results in the number 3412. Note that there may be other sequences of swaps but it can be shown that 3412 is the largest possible number. Also note that we may not swap the digit 4 with the digit 1 since they are of different parities.

**Example 2:**

**Input:** `num = 65875` **Output:** `87655` **Explanation:** Swap the digit 8 with the digit 6, this results in the number 85675. Swap the first digit 5 with the digit 7, this results in the number 87655. Note that there may be other sequences of swaps but it can be shown that 87655 is the largest possible number.

**Constraints:**

`1 <= num <= 10^9`

## Code Snippets

### C++:

```
class Solution {  
public:  
    int largestInteger(int num) {  
  
    }  
};
```

### Java:

```
class Solution {  
    public int largestInteger(int num) {  
  
    }  
}
```

### Python3:

```
class Solution:  
    def largestInteger(self, num: int) -> int:
```