

Problem 1370: Increasing Decreasing String

Problem Information

Difficulty: Easy

Acceptance Rate: 77.07%

Paid Only: No

Tags: Hash Table, String, Counting

Problem Description

You are given a string `s`. Reorder the string using the following algorithm:

1. Remove the **smallest** character from `s` and **append** it to the result.
2. Remove the **smallest** character from `s` that is greater than the last appended character, and **append** it to the result.
3. Repeat step 2 until no more characters can be removed.
4. Remove the **largest** character from `s` and **append** it to the result.
5. Remove the **largest** character from `s` that is smaller than the last appended character, and **append** it to the result.
6. Repeat step 5 until no more characters can be removed.
7. Repeat steps 1 through 6 until all characters from `s` have been removed.

If the smallest or largest character appears more than once, you may choose any occurrence to append to the result.

Return the resulting string after reordering `s` using this algorithm.

Example 1:

Input: s = "aaaabbbbcccc" **Output:** "abccbaabccba" **Explanation:** After steps 1, 2 and 3 of the first iteration, result = "abc". After steps 4, 5 and 6 of the first iteration, result = "abccba". First iteration is done. Now s = "aabbcc" and we go back to step 1. After steps 1, 2 and 3 of the second iteration, result = "abccbaabc". After steps 4, 5 and 6 of the second iteration, result = "abccbaabccba".

Example 2:

****Input:**** s = "rat" ****Output:**** "art" ****Explanation:**** The word "rat" becomes "art" after re-ordering it with the mentioned algorithm.

****Constraints:****

* `1 <= s.length <= 500` * `s` consists of only lowercase English letters.

Code Snippets

C++:

```
class Solution {  
public:  
    string sortString(string s) {  
  
    }  
};
```

Java:

```
class Solution {  
public String sortString(String s) {  
  
}  
}
```

Python3:

```
class Solution:  
    def sortString(self, s: str) -> str:
```