Topics

Reverse Shuffle Merge ☆

Problem Editorial △ Submissions Leaderboard Given a string, \boldsymbol{A} , we define some operations on the string as follows: a. reverse(A) denotes the string obtained by reversing string A. Example: reverse("abc") = "cba"b. shuffle(A) denotes any string that's a permutation of string A. Example: shuffle("god") ∈ ['god', 'gdo', 'ogd', 'odg', 'dgo', 'dog'] c. merge(A1, A2) denotes any string that's obtained by interspersing the two strings A1 & A2, maintaining the order of characters in both. For example, A1 = "abc" & A2 = "def", one possible result of merge(A1, A2) could be "abcdef", another could be "abdecf", another could be "abdecf" Given a string s such that $s \in merge(reverse(A), shuffle(A))$ for some string A, find the lexicographically smallest A. For example, s = abab. We can split it into two strings of ab. The reverse is ba and we need to find a string to shuffle in to get abab. The middle two characters match our reverse string, leaving the a and b at the ends. Our shuffle string needs to be ab. Lexicographically ab < ba, so our answer is ab. **Function Description** Complete the reverseShuffleMerge function in the editor below. It must return the lexicographically smallest string fitting the criteria. reverseShuffleMerge has the following parameter(s): • s: a string

Input Format

A single line containing the string 8.

Constraints

- s contains only lower-case English letters, ascii[a-z]
- $1 \le |s| \le 10000$

Output Format

Find and return the string which is the lexicographically smallest valid $m{A}$.

Sample Input 0

eggegg

Sample Output 0

egg

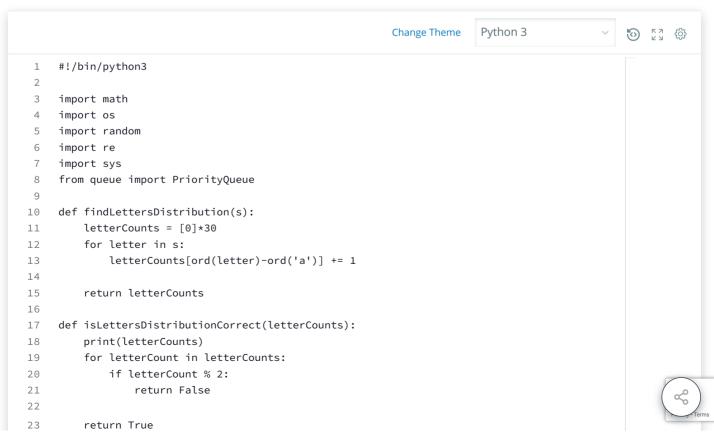
Explanation 0

Split "eggegg" into strings of like character counts: "egg", "egg" reverse("egg") = "gge" shuffle("egg") can be "egg" "eggegg" belongs to the merge of ("gge", "egg")



The merge is: eggegg.

```
'egg' < 'gge'
Sample Input 1
  abcdefgabcdefg
Sample Output 1
  agfedcb
Explanation 1
Split the string into two strings with like characters: abcdefg and abcdefg.
Reverse abcdefg = gfedcba
Shuffle agfedcb can be bcdefga
Merge to abcdefgabcdefg
Sample Input 2
  aeiouuoiea
Sample Output 2
  aeiou
Explanation 2
Split the string into groups of like characters: aeiou
Reverse aeiou = uoiea
These merge to aeiouuoiea
```



```
13/05/2020
         def constructStringFromLetterDistribution(s, letterCounts):
     25
             a = ''
     26
     27
               q = PriorityQueue()
               letters = {}
     28
     29
     30
               for c in s:
                   if (not c in letters) or (letters[c] == 0):
     31
     32
                       letters[c] = 1
     33
                       q.put(c)
     34
                   elif letters[c] == 1:
                                                                                                        Line: 47 Col: 1
                        ☐ Test against custom input
                                                                                          Run Code
                                                                                                         Submit Code
   <u>↑ Upload Code as File</u>
```



Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature

