

Reverse Shuffle Merge ★

45 more points to get your gold badge!

Rank: 196305 | Points: 805/850



- Problem
- Submissions
- Leaderboard
- Editorial
- Topics

RATE THIS CHALLENGE



Given a string, 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: $shuffle("god") \in ['god', 'gdo', 'ogd', 'odg', 'dgo', 'dog']$
- 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 $"adbecf"$ and so on.

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 s .

Constraints

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

Output Format

Find and return the string which is the lexicographically smallest valid 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 *gfedcba* 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*

Change Theme

Language

C#



```
1 using System.CodeDom.Compiler;
2 using System.Collections.Generic;
3 using System.Collections;
4 using System.ComponentModel;
5 using System.Diagnostics.CodeAnalysis;
6 using System.Globalization;
7 using System.IO;
8 using System.Linq;
9 using System.Reflection;
10 using System.Runtime.Serialization;
11 using System.Text.RegularExpressions;
12 using System.Text;
13 using System;
14
15 class Result
16 {
17
18     /*
19      * This class is used to store the result of the merge operation.
20      * It contains a list of strings, which are the merged strings.
21      * The list is initialized with the input strings.
22      * The merge operation is performed by repeatedly taking the
23      * minimum character from the input strings and adding it to the
24      * result string.
25      * The process continues until all characters from all input strings
26      * are included in the result string.
27      * The result string is then returned as a list of strings.
28      */
29 }
```

```
19      * Complete the 'reverseShuffleMerge' function below.
20      *
21      * The function is expected to return a STRING.
22      * The function accepts STRING s as parameter.
23      */
24
25      public static string reverseShuffleMerge(string s)
26      {
27          var result = new List<char>();
```

Line: 102 Col: 1

 Upload Code as File

☐ Test against custom input

Run Code

Submit Code

[Contest Calendar](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) | [Request a Feature](#)