3rd year Divorce String

locked

Problem | Submissions | Leaderboard | Discussions

Given a string s of lower and upper case English letters.

A divorsed string is a string which doesn't have two adjacent characters **s[i]** and **s[i + 1]** where:

- 0 <= i <= s.length - 2
- s[i] is a lower-case letter and s[i + 1] is the same letter but in upper-case or **vice-versa**.

To make the string divorsed, you can choose two adjacent characters that make the string couple and remove them. You can keep doing this until the string becomes divorsed.

Print the string after making it divorsed. The answer is guaranteed to be unique under the given constraints.

Notice that an empty string is also divorsed.

Input Format

Single line cointaing the string.

Constraints

- 1 <= s.length <= 100
- s contains only lower and upper case English letters.

Output Format

Print the divorsed string in single line.

Sample Input 0

```
abBAcC
```

Sample Output 0

```
-1
```

Explanation 0

We have many possible scenarios, and all lead to the same answer. For example:

- "abBAcC" --> "aAcC" --> "cC" --> ""
- "abBAcC" --> "abBA" --> "aA" --> ""
- return -1 if it is empty string

Sample Input 1

```
GgiIetu
```

Sample Output 1

```
etu
```

Submissions: 2
Max Score: 100
Difficulty: Easy

Rate This Challenge:

More

Python 3

1

Upload Code as File | Test against custom input | Run Code | Submit Code