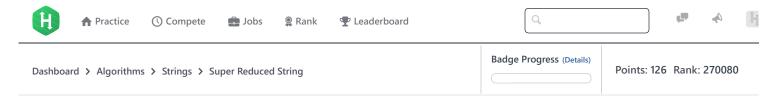
15/11/2017 HackerRank



Super Reduced String **■**





Steve has a string, **s**, consisting of **n** lowercase English alphabetic letters. In one operation, he can delete any *pair of adjacent letters* with same value. For example, string " aabcc " would become either " aab " or " bcc " after **1** operation.

Steve wants to reduce s as much as possible. To do this, he will repeat the above operation as many times as it can be performed. Help Steve out by finding and printing s's non-reducible form!

Note: If the final string is empty, print Empty String .

Input Format

A single string, s.

Constraints

• $1 \le n \le 100$

Output Format

If the final string is empty, print Empty String; otherwise, print the final non-reducible string.

Sample Input 0

aaabccddd

Sample Output 0

abd

Sample Case 0

Steve can perform the following sequence of operations to get the final string:

- 1. aaabccddd → abccddd
- 2. abccddd → abddd
- 3. $abddd \rightarrow abd$

Thus, we print abd.

Sample Input 1

baab

Sample Output 1

Empty String

15/11/2017 HackerRank

Explanation 1

Steve can perform the following sequence of operations to get the final string:

```
1. baab \rightarrow bb
```

```
2. bb → Empty String
```

Thus, we print Empty String.

Sample Input 2

aa

Sample Output 2

Empty String

Explanation 2

Steve can perform the following sequence of operations to get the final string:

```
1. aa \rightarrow Empty String
```

Thus, we print Empty String.

```
f y in
Submissions:<u>60334</u>
Max Score:10
Difficulty: Easy
Rate This Challenge:
☆☆☆☆☆
```

```
Current Buffer (saved locally, editable) & 40
                                                                                           Java 7
 1 ▼ import java.io.*;
 2 import java.util.*;
 3
    import java.text.*;
    import java.math.*;
 5
    import java.util.regex.*;
 6
 7 ▼ public class Solution {
 8
        static String super_reduced_string(String s){
 9 ▼
10
            // Complete this function
11
12
13 🔻
        public static void main(String[] args) {
14
            Scanner in = new Scanner(System.in);
            String s = in.next();
15
16
            String result = super_reduced_string(s);
17
            System.out.println(result);
18
19
    }
20
                                                                                                                    Line: 1 Col: 1
```

Run Code Submit Code

15/11/2017 HackerRank

Join us on IRC at #hackerrank on freenode for hugs or bugs.

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