

A. Escape from Stones

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Squirrel Liss lived in a forest peacefully, but unexpected trouble happens. Stones fall from a mountain. Initially Squirrel Liss occupies an interval $[0, 1]$. Next, n stones will fall and Liss will escape from the stones. The stones are numbered from 1 to n in order.

The stones always fall to the center of Liss's interval. When Liss occupies the interval $[k - d, k + d]$ and a stone falls to k , she will escape to the left or to the right. If she escapes to the left, her new interval will be $[k - d, k]$. If she escapes to the right, her new interval will be $[k, k + d]$.

You are given a string S of length n . If the i -th character of S is "l" or "r", when the i -th stone falls Liss will escape to the left or to the right, respectively. Find the sequence of stones' numbers from left to right after all the n stones falls.

Input

The input consists of only one line. The only line contains the string S ($1 \leq |S| \leq 10^6$). Each character in S will be either "l" or "r".

Output

Output n lines — on the i -th line you should print the i -th stone's number from the left.

Examples

input
llrlr
output
3
5
4
2
1

input
rrlll
output
1
2
5
4
3

input
lrlrr
output
2
4
5
3
1

Note

In the first example, the positions of stones 1, 2, 3, 4, 5 will be , respectively. So you should print the sequence: 3, 5, 4, 2, 1.

Codeforces Round #162 (Div. 1)

Finished

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ACM-ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Problem tags

constructive algorithms



data structures

implementation

two pointers

No tag edit access

→ Contest materials

- Announcement 
- Tutorial #1 
- Tutorial #2 