Do Not Drink Too Much in Cubbes 425 Points

ISM is not a very cheerful place to live. I have a friend who was not so happy with his life, also he was sad because “C For Code” was about to end. Let me tell you his story. He went to Cubbes to have some drinks. He was happy till he had 8 pegs. But after drinking few more pegs he went mad and started beating everyone. To stop him several guards of ISM were called. They saw that he occupied an interval of length 2d, i.e. [x - d, x + d], guard threw a stone at point x (center of the interval) to stop him, but he moved either to his left or right to escape from it. If he moves left, his new interval will be [x - d, x] and if he moves to the right, his new interval will be [x, x + d]. Again guard threw one more stone to the center of his new interval and my friend kept on escaping from it by moving either to the left or to the right and occupying a new interval. This process continued until the guards threw “N” stones. All the stones that the guards threw were numbered from 1 to N in the order they threw them.

You will be given a string of length “S” that will contain either character “L” or “R”. If the i-th character in “L”, it means that my friend moved to his left after the i-th stone was thrown to him, otherwise he moved to his right.

Can you find the sequence of stone numbers from left to right after all the n stones were thrown.

For example, if the sequence of stones would be LRRLR, then the positions at which stones 1, 2, 3, 4 and 5 were thrown will be x, x-(d/2), x-(d/4), x-(d/8), x-(3*d/16) respectively.

So the sequence from left to right will be 2 3 5 4 1

Input

The only line of input will contain a string containing characters “L” and “R”.

Output

If the input string is of length “N”, then output “N” integers in separate lines denoting the sequence of thrown stone numbers from left to right after all the “N” stones were thrown.

Constraints

1 <= N <= 10^4

Example

Input:

```
LRRLR
```

Output:

```
2
3
5
4
```

```
1
```

Input:

```
LLRRL
```

Output:

```
3
4
5
2
1
```

Input:

```
LRLR
```

Output:

```
2
4
3
1
```