A. Hotelier

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Amugae has a hotel consisting of 10 rooms. The rooms are numbered from 0 to 9 from left to right.

The hotel has two entrances — one from the left end, and another from the right end. When a customer arrives to the hotel through the left entrance, they are assigned to an empty room closest to the left entrance. Similarly, when a customer arrives at the hotel through the right entrance, they are assigned to an empty room closest to the right entrance.

One day, Amugae lost the room assignment list. Thankfully Amugae's memory is perfect, and he remembers all of the customers: when a customer arrived, from which entrance, and when they left the hotel. Initially the hotel was empty. Write a program that recovers the room assignment list from Amugae's memory.

Input

The first line consists of an integer n ($1 \le n \le 10^5$), the number of events in Amugae's memory.

The second line consists of a string of length n describing the events in chronological order. Each character represents:

- 'L': A customer arrives from the left entrance.
- 'R': A customer arrives from the right entrance.
- '0', '1', ..., '9': The customer in room x (0, 1, ..., 9 respectively) leaves.

It is guaranteed that there is at least one empty room when a customer arrives, and there is a customer in the room x when x (0, 1, ..., 9) is given. Also, all the rooms are initially empty.

Output

In the only line, output the hotel room's assignment status, from room 0 to room 9. Represent an empty room as '0', and an occupied room as '1', without spaces.

Examples

input	Copy
8 LLRL1RL1	

output	Copy
1010000011	

input	Copy
9 L0L0LLRR9	

output	Copy
1100000010	

Note

In the first example, hotel room's assignment status after each action is as follows.

- First of all, all rooms are empty. Assignment status is `0000000000`.
- L: a customer arrives to the hotel through the left entrance. Assignment status is `1000000000`.
- L: one more customer from the left entrance. Assignment status is `1100000000`.
- R: one more customer from the right entrance. Assignment status is `1100000001`.
- L: one more customer from the left entrance. Assignment status is `1110000001`.

- 1: the customer in room 1 leaves. Assignment status is `1010000001`.
- R: one more customer from the right entrance. Assignment status is `1010000011`.
- L: one more customer from the left entrance. Assignment status is `1110000011`.
- 1: the customer in room 1 leaves. Assignment status is `1010000011`.

So after all, hotel room's final assignment status is `1010000011`.

In the second example, hotel room's assignment status after each action is as follows.

- L: a customer arrives to the hotel through the left entrance. Assignment status is `1000000000`.
- 0: the customer in room 0 leaves. Assignment status is `0000000000`.
- L: a customer arrives to the hotel through the left entrance. Assignment status is `1000000000` again.
- 0: the customer in room 0 leaves. Assignment status is `0000000000`.
- L: a customer arrives to the hotel through the left entrance. Assignment status is `1000000000`.
- L: one more customer from the left entrance. Assignment status is `1100000000`.
- R: one more customer from the right entrance. Assignment status is `1100000001`.
- R: one more customer from the right entrance. Assignment status is `1100000011`.
- 9: the customer in room 9 leaves. Assignment status is `1100000010`.

So after all, hotel room's final assignment status is `1100000010`.