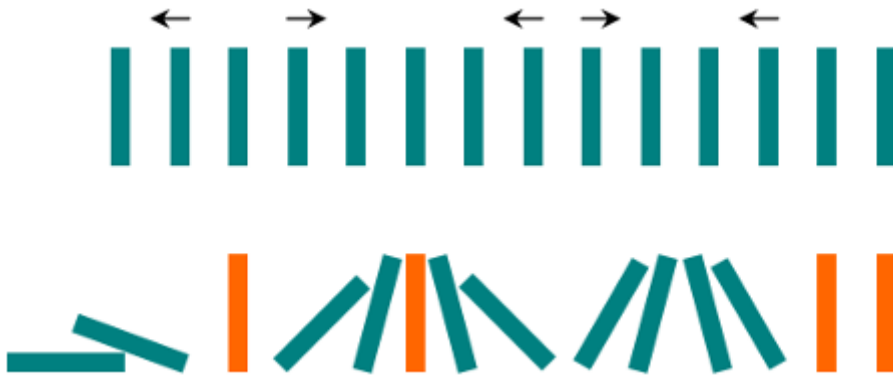


## 838. Push Dominoes

There are  $N$  dominoes in a line, and we place each domino vertically upright.

In the beginning, we simultaneously push some of the dominoes either to the left or to the right.



After each second, each domino that is falling to the left pushes the adjacent domino on the left.

Similarly, the dominoes falling to the right push their adjacent dominoes standing on the right.

When a vertical domino has dominoes falling on it from both sides, it stays still due to the balance of the forces.

For the purposes of this question, we will consider that a falling domino expends no additional force to a falling or already fallen domino.

Given a string "S" representing the initial state.  $S[i] = 'L'$ , if the  $i$ -th domino has been pushed to the left;  $S[i] = 'R'$ , if the  $i$ -th domino has been pushed to the right;  $S[i] = '.'$ , if the  $i$ -th domino has not been pushed.

Return a string representing the final state.

### Example 1:

**Input:** ".L.R...LR..L.."

**Output:** "LL.RR.LLRRLL.."

### Example 2:

**Input:** "RR.L"

**Output:** "RR.L"

**Explanation:** The first domino expends no additional force on the second domino.

### Note:

1.  $0 \leq N \leq 10^5$
2. String dominoes contains only 'L', 'R' and '.'.

**Solution:** We first Add L and R at the first and last pos of the string

For examples .L.R...LR..L.. → L.L.R...LR..L..R

focus on 'L' and 'R'.

'R.....R' => 'RRRRRRRRR'

'R.....L' => 'RRRRLLLLL' or 'RRRR.LLLL'

'L.....R' => 'L.....R'

'L.....L' => 'LLLLLLLLL'

//C++

```
string pushDominoes(string d) {
    d = 'L' + d + 'R';
    string res = "";
    for (int i = 0, j = 1; j < d.length(); ++j) {
        if (d[j] == '.') continue;
        int middle = j - i - 1;
        if (i > 0) res += d[i];
        if (d[i] == d[j]) res += string(middle, d[i]);
        else if (d[i] == 'L' && d[j] == 'R') res += string(middle, '.');
        else res += string(middle / 2, 'R') + string(middle % 2, '.') +
string(middle / 2, 'L');
        i = j;
    }
    return res;
}
```

python

```
def pushDominoes(self, d):
    d = 'L' + d + 'R'
    res = []
    i = 0
    for j in range(1, len(d)):
        if d[j] == '.': continue
        middle = j - i - 1
        if i: res.append(d[i])
        if d[i] == d[j]: res.append(d[i] * middle)
        elif d[i] == 'L' and d[j] == 'R': res.append('.') * middle
        else: res.append('R' * (middle / 2) + '.' * (middle % 2) + 'L' *
(middle / 2))
        i = j
    return ''.join(res)
```