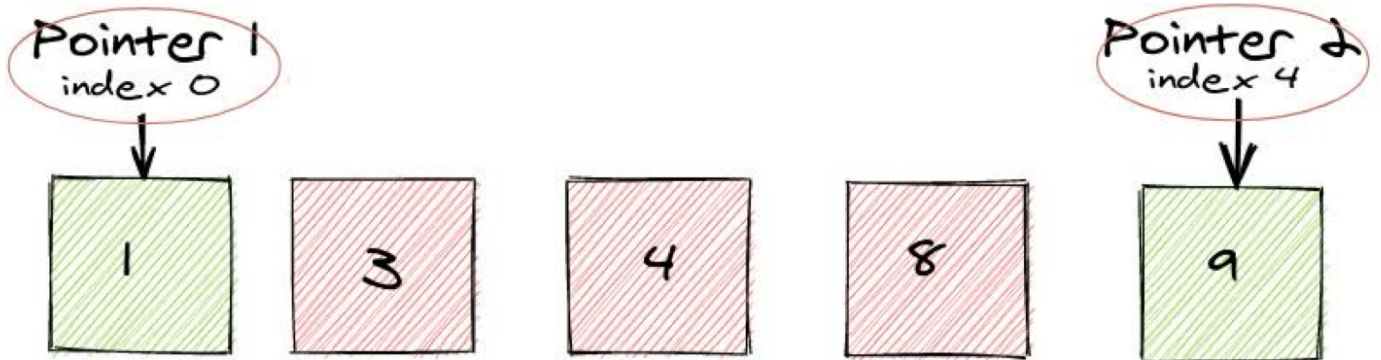


Two-Pointers Technique 10



Leetcode
- 838

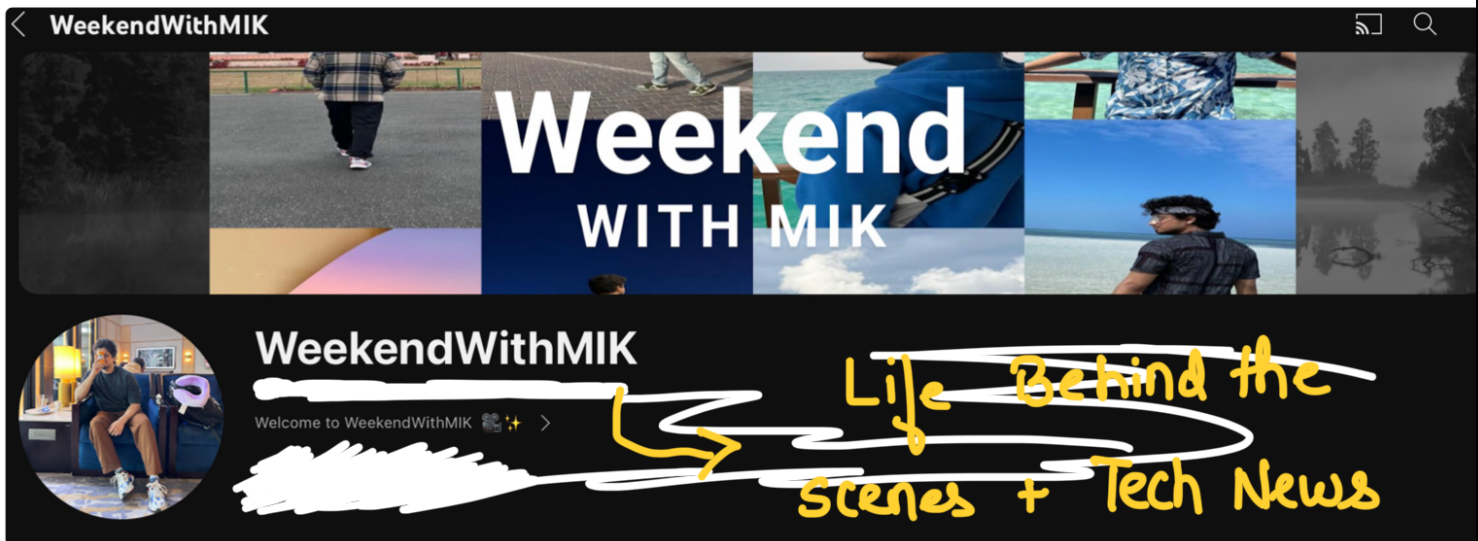
Medium



∞   codestorywithmik

  CSwithMIK

  codestorywithMIK



MOTIVATION...

Your future is built in the
hours no one is watching.
ee Comfort never created legacy"



MIK...

838. Push Dominoes

Medium

Topics

Companies



Solved ✓

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.


You are given a string `dominoes` representing the initial state where:

- `dominoes[i] = 'L'`, if the i^{th} domino has been pushed to the left,
- `dominoes[i] = 'R'`, if the i^{th} domino has been pushed to the right, and
- `dominoes[i] = '.'`, if the i^{th} domino has not been pushed.

Return a string representing the final state.

Example:- dominoes = "RR.L"

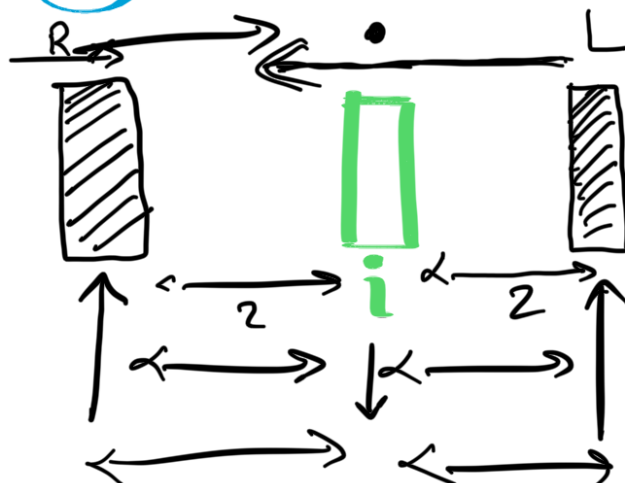
output = "RR.L"



dominoes = ".L.R...LR..L.."

Output = "LL.RR.LLRRLL.."

Thought Process



physics.

• L • R • • • L R • • L • •



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	
leftClosest R	-1	-1	-1	3	3	3	3	-1	8	8	8	-1	-1	-1	R →

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	
rightClosest L	1	1	-1	-1	7	7	7	7	-1	11	11	11	-1	-1	← L

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	
result =	L	L	•	R	R	•	L	L	•	•	•	•	•	•	
							i								

1st st $\Rightarrow \text{distLeftR} = \text{abs}(i - \text{leftClosestR}[i])$; $6 - 3 = 3$

2nd $\Rightarrow \text{distRightL} = \text{abs}(i - \text{rightClosestL}[i])$; $6 - 7 = 1$

if (rightClosestL[i] == leftClosestR[i]) {

result[i] = '•';

} else if (rightClosestL[i] == '-1') { ←

result[i] = 'R';

} else if (leftClosestR[i] == '-1') {

result[i] = 'L';

```

} else if (distR == distL) {
    res[i] = '0';

```

```

} else {

```

```

    res[i] = distRightL < distLeftR ? 'L' : 'R';

```

```

}

```

return res;

