838. Push Dominoes

Medium | Topics | Companies

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*.

Python3 • Auto

```python
class Solution:
    def pushDominoes(self, dominoes: str) -> str:
        l = [0] * len(dominoes) # -1 left; 0 still; 1 right
        prev = donimoes[0]

        # want to scan twice: one for the force to the right, one for the left, calculaye the force on the '.', and it will fall on the direction with greater force
        for i in range(1, len(dominoes)):
            if donimoes[i] == '.':


            elif donimoes[i] == 'R':

            else:

        return ''.append('L' if e ==-1 else '.' if e == 0 else 'R' for e in l)
```

one line code for 'if, elif, else'

✅ Syntax:

python Copy

```python
value = result1 if condition1 else result2 if condition2 else result3
```

two-way scan:
greedy + simulation method and runs in O(n) time and O(n) space

All Submissions

Accepted 44 / 44 testcases passed

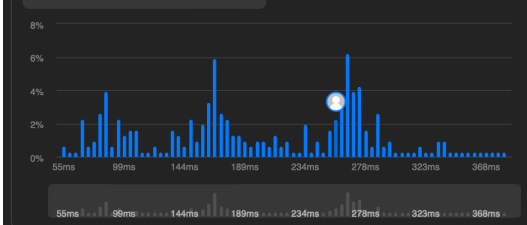
AndrewC275 submitted at May 02, 2025 12:06

Editorial | Solution

Runtime 259 ms | Beats 39.48%

Analyze Complexity

Memory 22.80 MB | Beats 45.40%



Code | Python3

```python
class Solution:
    def pushDominoes(self, dominoes: str) -> str:

        # want to scan twice: one for the force to the right, one for the l
        force = [0] * len(dominoes)

        # scan right-wards, from head to tail
        f = 0
```

View more

Python3 • Auto

```python
class Solution:
    def pushDominoes(self, dominoes: str) -> str:
        # want to scan twice: one for the force to the right, one for the left, calculaye the force on the '.', and it will fall on the direction with greater force
        force = [0] * len(dominoes)
        # scan right-wards, from head to tail
        f = 0
        for i in range(len(dominoes)):
            if  dominoes[i]== 'R':
                f = len(dominoes)
            elif dominoes[i] == 'L':
                f = 0
            else:
                f = max(f-1, 0)
            force[i] = f
        # scan left-wards, from tail to head
        f = 0
        for i in range(len(dominoes)-1, -1, -1):
            if dominoes[i] == 'L':
                f = len(dominoes)
            elif dominoes[i] == 'R':
                f = 0
            else:
                f = max(f-1, 0)
            force[i] -= f
        res = []
        for e in force:
            if e > 0:
                res.append('R')
            elif e < 0:
                res.append('L')
            else:
                res.append('.')
        return ''.join(res)
```

```python
class Solution:
    def pushDominoes(self, dominoes: str) -> str:
        temp = ''

        while dominoes != temp:
            temp = dominoes
            dominoes = dominoes.replace('R.L', 'xxx')
            dominoes = dominoes.replace('R.', 'RR')
            dominoes = dominoes.replace('.L', 'LL')

        return  dominoes.replace('xxx', 'R.L')
```

top solution:

- keep running until no further changes could be made
- when the forces from left and right is canceled out