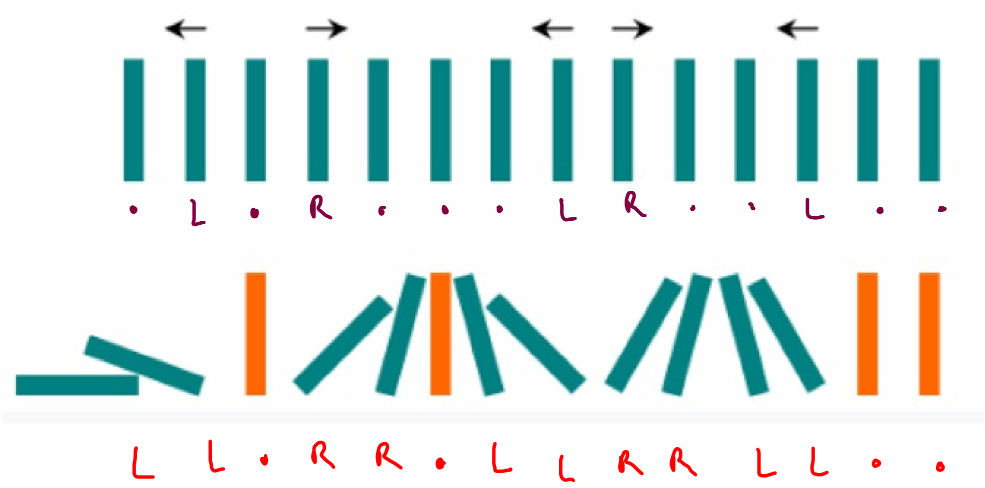
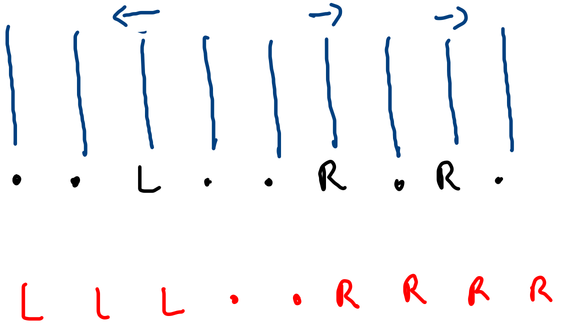
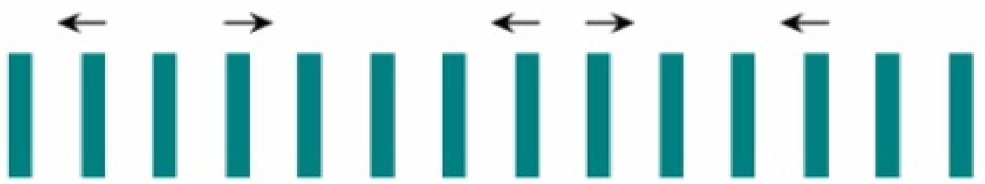
838. Push Dominoes







str : L . L . R . . . L R . . L . . R

case 1 : L . . . . L → L L L L L L

case 2 : L . . . . R → L . . . . R

case 3 : R . . . . R → R R R R R R

case 4 : R L

dots (even)

R R R L L L

dots (odd)

R R R . L L L

```java
public String pushDominoes(String dominoes) {
    String str = 'L' + dominoes + 'R';
    StringBuilder sb = new StringBuilder(str);

    int i = 0;

    while(i < str.length()-1) {
        int j = i+1;

        while(j < str.length() && str.charAt(j) == '.') {
            j++;
        }

        //solve i to j
        solve(sb,i,j);
        i = j;
    }

    //remove extra 'L','R' which were initially added
    sb.deleteCharAt(0);
    sb.deleteCharAt(sb.length()-1);

    return sb.toString();
}
```

```java
public static void solve(StringBuilder sb,int i,int j) {

    if(sb.charAt(i) == 'L' && sb.charAt(j) == 'L') {
        for(int k = i+1; k <= j-1;k++) {
            sb.setCharAt(k,'L');
        }
    }
    else if(sb.charAt(i) == 'L' && sb.charAt(j) == 'R') {
        //no changes
    }
    else if(sb.charAt(i) == 'R' && sb.charAt(j) == 'R') {
        for(int k = i+1; k <= j-1;k++) {
            sb.setCharAt(k,'R');
        }
    }
    else if(sb.charAt(i) == 'R' && sb.charAt(j) == 'L') {
        while(i < j) {
            sb.setCharAt(i,'R');
            sb.setCharAt(j,'L');

            i++;
            j--;
        }
    }
}
```

".L.R...LR..L.."

L R L R L

str : L . L . R . . . L R . . L . . R

ans : ~~L~~ L L . R R . L L R R L L . . ~~R~~

Output: "LL.RR.LLRRLL.."

829. Consecutive Numbers Sum

Hard 921 1111 Add to List Share

Given an integer `n`, return *the number of ways you can write* `n` *as the sum of consecutive positive integers.*

```
Input: n = 15
Output: 4
Explanation: 15 = 8 + 7 = 4 + 5 + 6 = 1 + 2 + 3 + 4 + 5
```

9

→ 9, 4+5, 2+3+4

5

→ 5, 2+3

15 = 1 + 2 + 3 + 4 + 5

15 = 4 + 5 + 6

15 = 7 + 8

15 = 15

$k = 1 \quad 15 = 15$

$k = 2 \quad 15 = 7 + 8$

$k = 3 \quad 15 = 4 + 5 + 6$

$k = 5 \quad 15 = 1 + 2 + 3 + 4 + 5$

starting from x

$$x + (x+1) + (x+2) + (x+3) + \dots (x + k-1) \geq N$$

$$kx + (1 + 2 + 3 + \dots k-1) \geq N$$

$$kx + \frac{k(k-1)}{2} \geq N$$

$$\boxed{x \leq \frac{N - \frac{k(k-1)}{2}}{k}}$$

max value of k
when $x = 1$

$$1 \leq \frac{N - \frac{k(k-1)}{2}}{k}$$

$$k \leq N - \frac{k(k-1)}{2}$$

$$2k \leq 2N - k^2 + k$$

$$\boxed{k^2 + k \leq 2N}$$

$N = 15$

K → no. of terms used to represent N.

x → starting term in a representation

K	x	
1	15	$15 = 15$
2	7	$15 = 7 + 8$
3	4	$15 = 4 + 5 + 6$
5	1	$15 = 1 + 2 + 3 + 4 + 5$

$$x + (x+1) + (x+2) + \dots (x + K - 1) \geq N$$

$$Kx + (1 + 2 + \dots K-1) \geq N$$

$$Kx + \frac{K(K-1)}{2} \geq N$$

$$\boxed{x \leq \left[N - \frac{K(K-1)}{2}\right] / K} \quad \text{— } 1$$

max K can be achieved when $x = 1$, put $x = 1$ in 1.

$$1 \leq \frac{N - \frac{K(K-1)}{2}}{K}$$

$$K \leq N - \frac{K(K-1)}{2}$$

$$2K \leq 2N - K^2 + K$$

$$\boxed{K^2 + K \leq 2N}$$

53. Maximum Subarray

Easy 17695 842 Add to List Share

Given an integer array `nums`, find the contiguous subarray (containing at least one number) which has the largest sum and return *its sum*.

A **subarray** is a **contiguous** part of an array.

```
Input: nums = [-2,1,-3,4,-1,2,1,-5,4]
```

Kadane's algo

→ max sum subarray in

o(n)

arr: 1 -2 3

1 -2 3

1 -2 -2 3

1 -2 3

```
Input: nums = [-2,1,-3,4,-1,2,1,-5,4]
```

i

-2	1	-3	4	-1	2	1	-5	4
0	1	2	3	4	5	6	7	8

```
if (csum > 0) {
    csum += arr[i];
}
else {
    csum = arr[i];
}
msum = max(msum, csum);
```

msum = ~~-2~~ ~~1~~ ~~4~~ ~~5~~ 6

csum = ~~-2~~ ~~-2~~ ~~4~~ ~~3~~ ~~5~~ ~~6~~ ~~1~~ 5

1191. K-Concatenation Maximum Sum

Given an integer array `arr` and an integer `k`, modify the array by repeating it `k` times.

For example, if `arr = [1, 2]` and `k = 3` then the modified array will be `[1, 2, 1, 2, 1, 2]`.

A = [1, 2] , k = 3

B = [1, 2, 1, 2, 1, 2]

concatenated array

3 2 -1 4 5 -6 2

do it without creating 'B', in O(N)

$A = [-2, -1, 4]$ $K = 5$

$B = \underline{-2, -1, 4}, \underline{\boxed{-2, -1, 4}, \boxed{-2, -1, 4}, \boxed{-2, -1, 4}}, \underline{-2, -1, 4}$

$(-2, -1, 4)$ sum > 0 $\rightarrow$ $(K-2)^*$ sum $+$ 2' arr kadanes

$(80, -100, 2)$ sum ≤ 0 $\rightarrow$ 2' arr kadanes