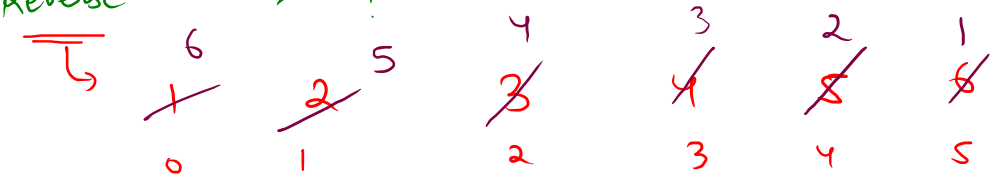
Arrays.

1. Max / Min → ?
2. Reverse → ?

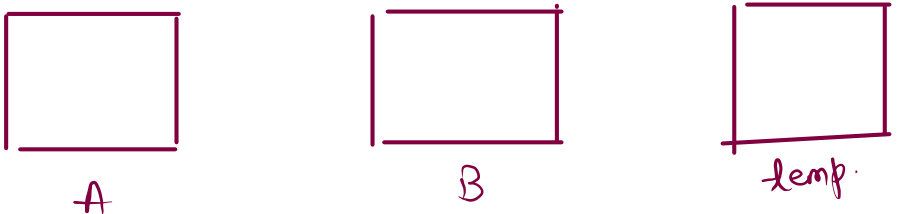



```
while (i < j)
{

}
```

j i

Swap.

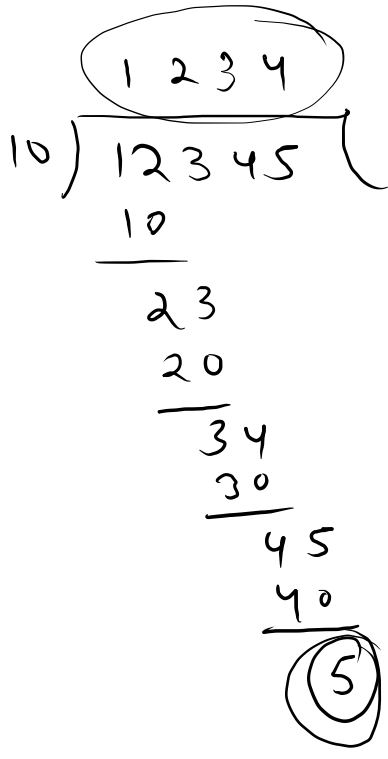


Count Digit

$n = 12345$

* last Digit ← % 10 , /10

5 , 1234



Rotate - Array

k = 3

1 2 3 4 5 6 7

↓

5 6 7 1 2 3 4

1 2 3 4 5 6 7	k = 0	
7 1 2 3 4 5 6	k = 1	→ k = 8
6 7 1 2 3 4 5	k = 2	→ k = 9
5 6 7 1 2 3 4	k = 3	
4 5 6 7 1 2 3	k = 4	
3 4 5 6 7 1 2	k = 5	
2 3 4 5 6 7 1	k = 6	
1 2 3 4 5 6 7	k = 7	

k = k % A.length

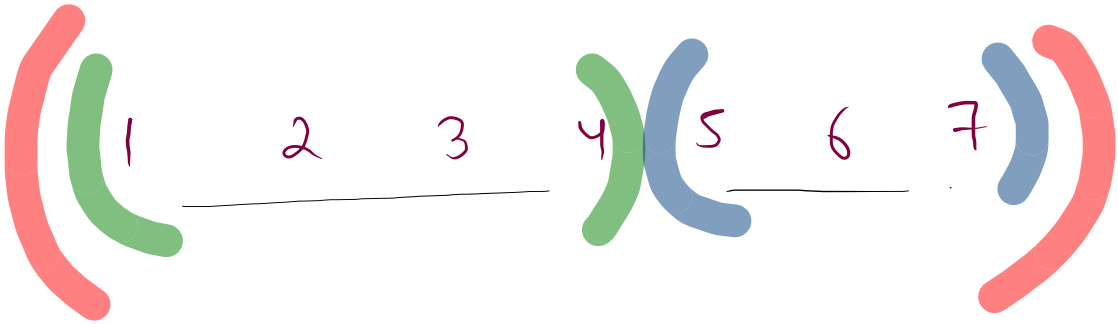
%

p % q

p < q

↳ ans = p

Rotate.



k = 3

reverse.

4 3 2 1 7 6 5

5 6 7 1 2 3 4

Ans. ⟶

Sub-array

10	20	30	40
0	1	2	3

10 (0,0)

10 20 (0,1)

10 20 30 (0,2)

10 20 30 40 (0,3)

20 (1,1)

20 30 (1,2)

20 30 40 (1,3)

30 (2,2)

30 34 (2,3)

40 (3,3)

start	end
0	0, 1, 2, 3
1	1, 2, 3
2	2, 3
3	3

Total Subarray = $n(n+1)/2$