



## Implement stack using a linked list



st

✓ push(10)

✓ push(20)

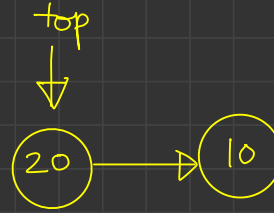
✓ push(30)

✓ push(40)

✓ pop() → 40

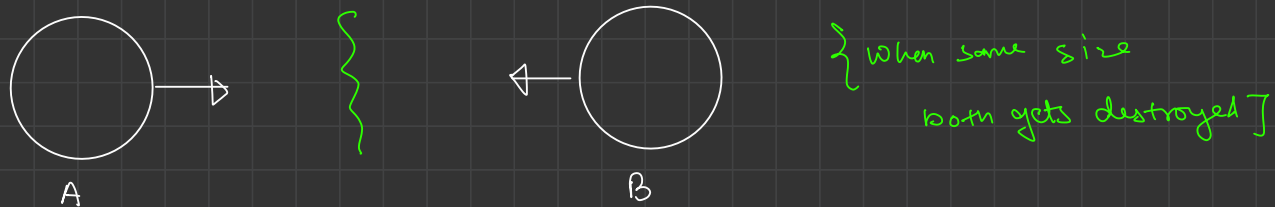
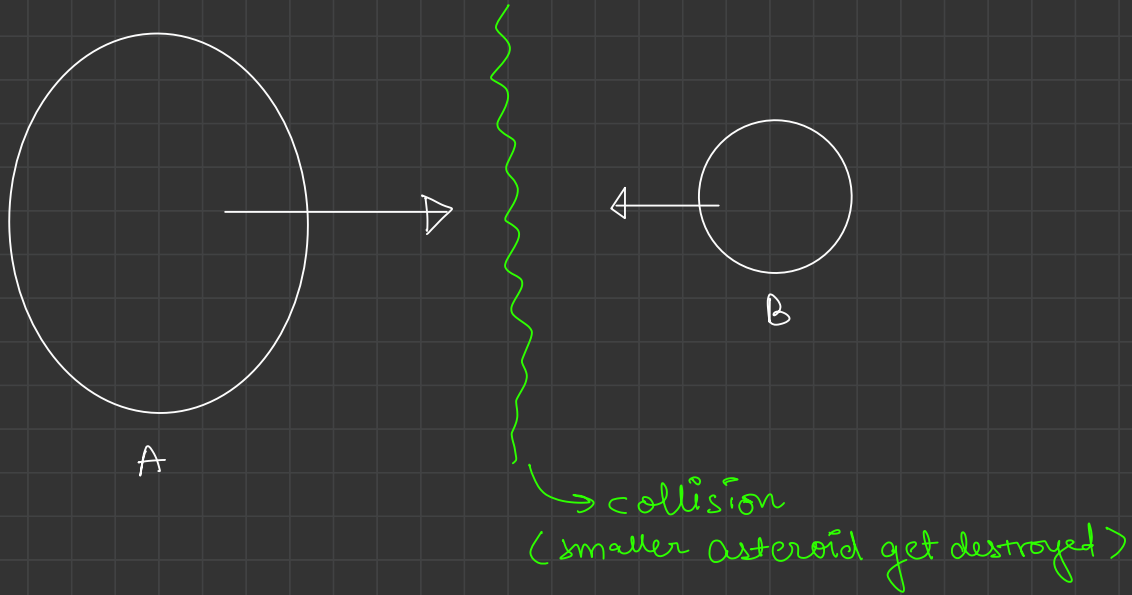
pop() → 30

push(50)



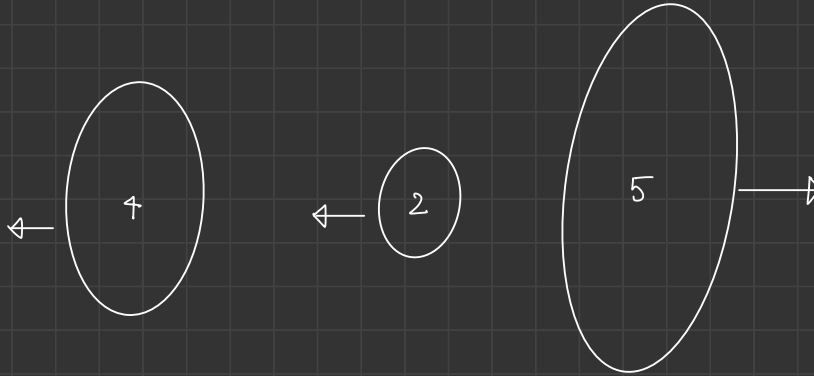
{ push() → addFirst() in linked list  
TC: O(1)  
pop() → remove first() in ll  
TC: O(1)

# Asteroid Collision



asteroids[] = { 1, 2, 3, -4, -2, 5, -3 }

↑    ↑    ↑    ↑    ↑    ↑    ↑



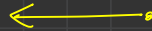
---

moving → freely in space

ans = { -4, -2, 5 }

## Conditions of collision

①



X

②



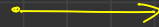
X

③



✓

④



X

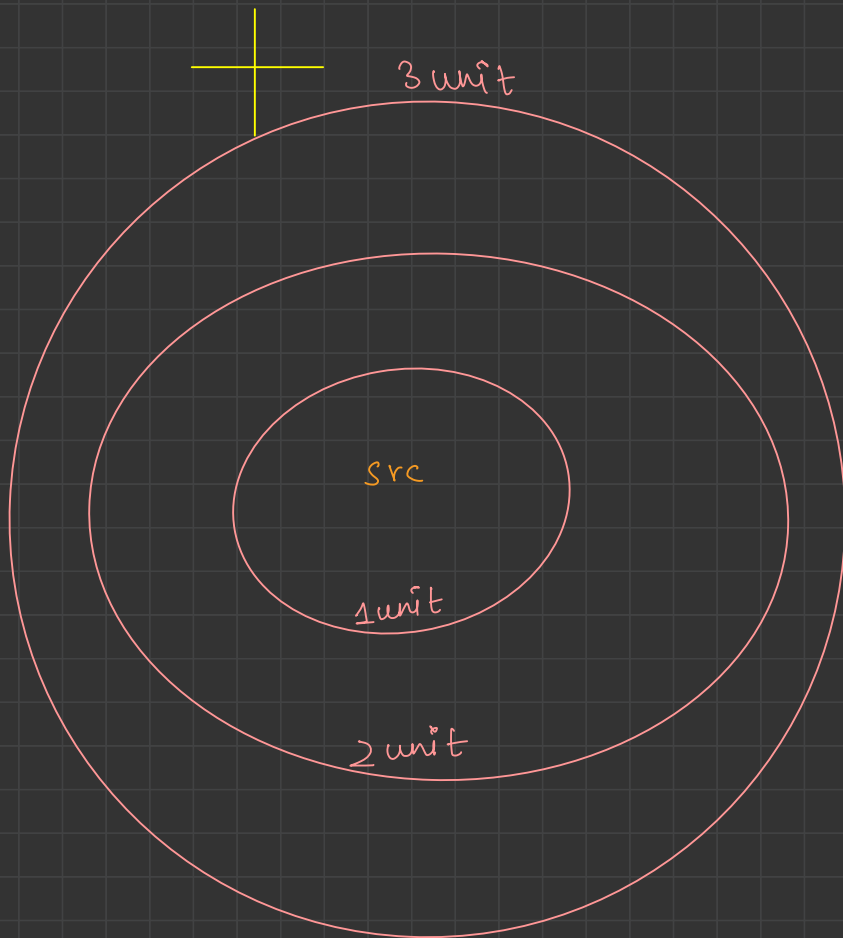
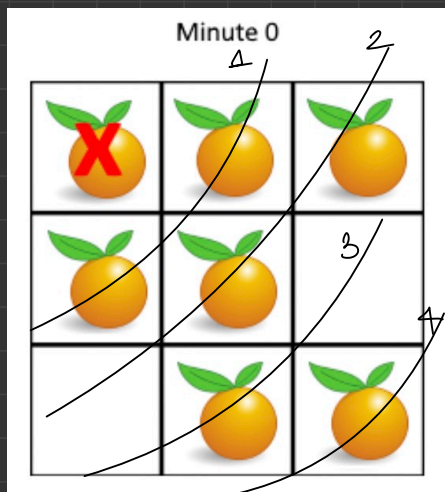
asteroids[] = { 1, 2, 3, -4, -2, 5, -3 }   
 explored      ↑   ↑   ↑   ↑   ↑   ↑   ↑      unexplored

5  
-2  
-4








{ -4, -2, 5 }  
←

people freely moving in space

# Rotten Oranges



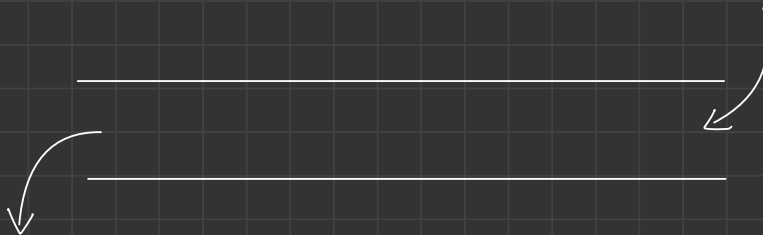
Minute 0

	0	1	2
0			
1			
2			

$$\underline{\text{time} = \text{level} - 1}$$

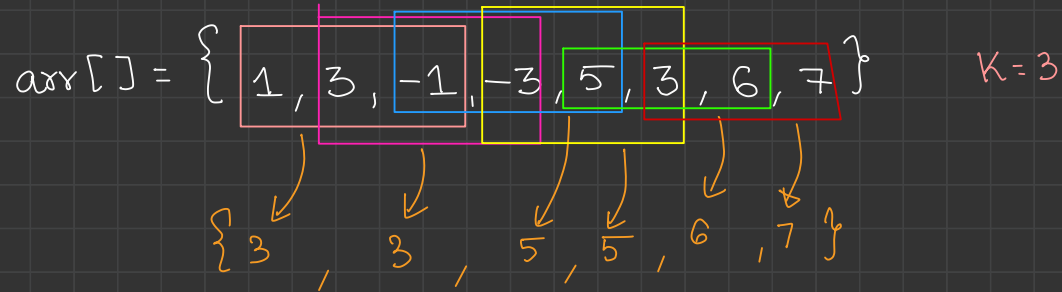
$$\text{level} = \cancel{0} \cancel{1} \cancel{2} \cancel{3} \cancel{4} \overset{\curvearrowright}{5}$$

$$\text{size} = 0$$





## Sliding Window Maximum



Brute force

TC:  $O(N \times K)$

SC:  $O(1)$

$arr[] = \{ \overset{0}{1}, \overset{1}{3}, \overset{2}{-1}, \overset{3}{-3}, \overset{4}{5}, \overset{5}{3}, \overset{6}{6}, \overset{7}{7} \}$

3, 3, 5, 5, 6, 7

$ngexil[] = \{ 1, 4, 4, 4, 6, 6, 7, 8 \}$

$\underline{\underline{N-K+1}}$   
 $\downarrow$   
Windows

arr[] = {<sup>0</sup>1, <sup>1</sup>3, <sup>2</sup>-1, <sup>3</sup>-3, <sup>4</sup>5, <sup>5</sup>3, <sup>6</sup>6, <sup>7</sup>7}

add Last

remove Last

3, 3, 5, 5, 6, 7

