Q → Given a sorted array of length n having integers in the range [0, n-2]. Every element is present once but only one element is present twice. find the repeating element.

Ex → [0, 1, 2, 2, 3, 4, 5, 6] $_{8}$

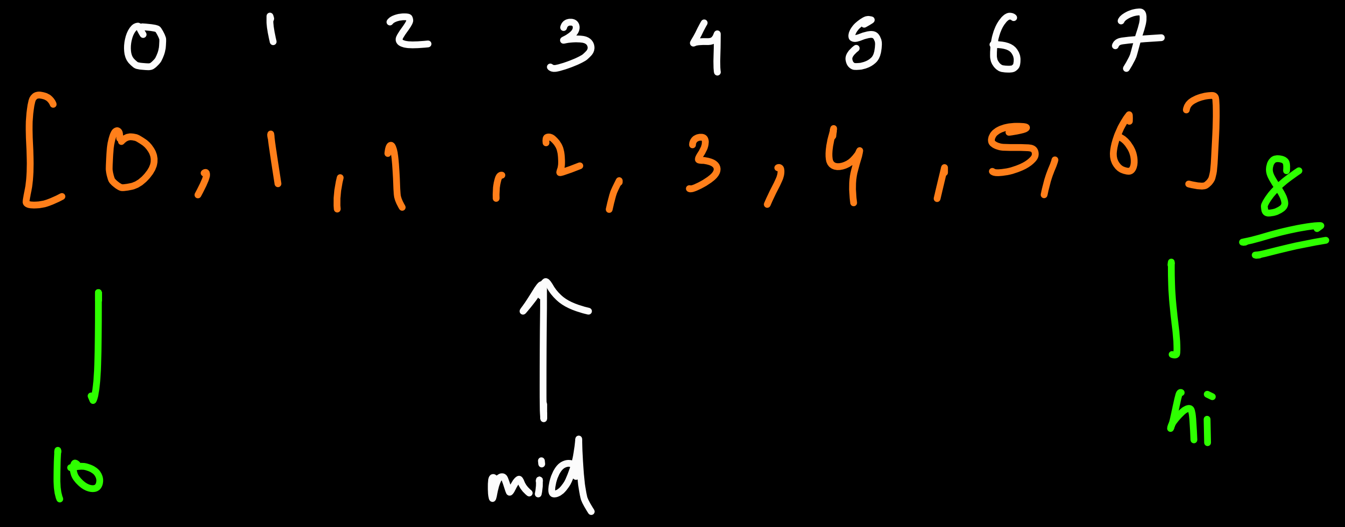
ans → 2

Do something better than O(n)

$[0, 1, 2, 2, 3, 4, 5, 6]_{\underline{\underline{8}}}$

Brute force → linear search → just check if the i^{th} index is equal to the $i-1^{th}$ index $\forall\ i \in [1, n-1]$ $O(n)$

Binary Search



if (arr[mid] == arr[mid-1])

arr[mid] == mid-1

↳ if we hit a mid that satisfies the given condition then we can say either mid is the ans or mid is to the right of ans.

```
while (lo <= hi) {
        mid = lo + (hi-lo)/2;
        if(arr [mid] == mid -1) {

                if ( arr(mid) == arr [mid-1]) {

                        return mid;

                }
                hi = mid-1;
        } else {

                lo = mid+1;

        }
}
```

↳ O(log n)

O(1)

↳ Sol^n 1 Brute Force → $O(n^2)$

[3, 1, 3, 2, 4]

```
for( i=0 ; i<n ; i++)
        for (j=i+1 ; j<n ; j++)
                if(a[i] == a[j])
```

Sol^n 2

We can sort the array

→ if we sort the array then repeating element will be adjacent.

$[1, 2, 3, 3, 4]$

↑

$O(n\log n + n)$

$O(n\log n)$

Space → $O(1)$

Sol^n 3

→ Let's create a frequency **map.**

[3, 1, 3, 2, 4]

{ 3 : 2
  1 : 1
  2 : 1
  4 : 1
}

time O(n)

Space → O(n)

Sol$^{4}_{n}$

the moment we encounter the elemt, let's mark it's presence.

$$[\overset{?}{-2}, -1, -3, -4, \overset{?}{3}]$$

idx → 1 2 3 4 5 actual = |curr|

we were taking extra space because we wanted to keep a track of which element occurs again.

↳ [1, n] → all non zero positive

if the array was not having repeated element then all the one based indexes must have been present in the given array

$$\left[\begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ -5, & -3, & -1, & 6, & -2, & -2, & 4 \\ & & & & & \uparrow & \end{array}\right] \rightarrow \underline{O(n)\ \text{time}}$$

$$\underline{O(1)}\ \text{space}$$

if value at index is -ve, then that index is present already in array.

```
curr element = |arr[i]|

if (arr[curr Element] < 0)
        return curr element;

arr[curr Element] *= (-1);
```

↳ unique element → $[1, n] \longrightarrow \frac{n \times (n+1)}{2}$

arr → arr.length → $n+1$

apart from these uniq element one element is repeaty.

$[1, 2, 5, 4, 3, 2]$ 6

$n+1 = 6$

$\boxed{n = 5}$ → everything from 1 to 5 is in the array

Time → $O(n)$

Space → $O(1)$

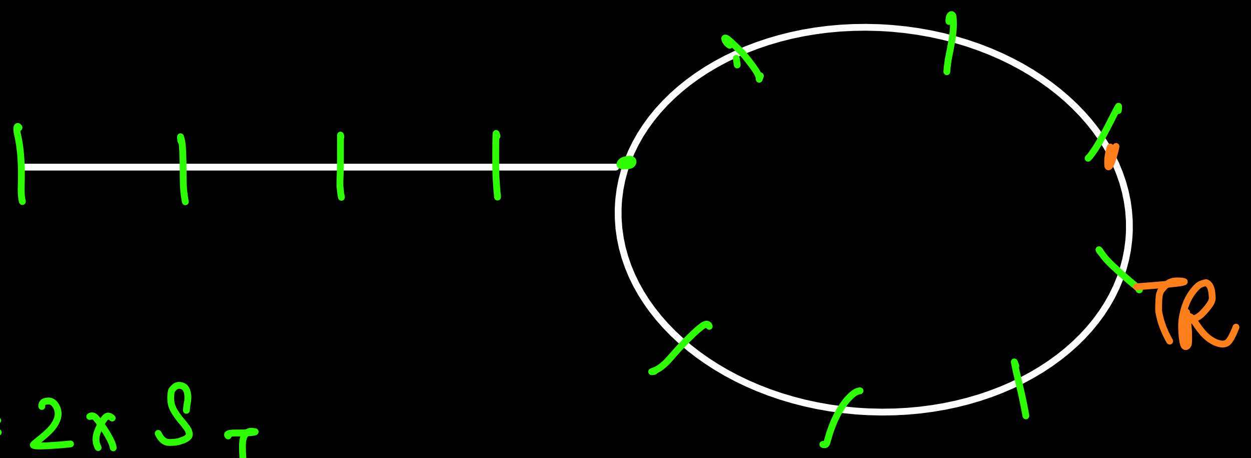
$$\underbrace{1 + 2 + 5 + 4 + 3 + 2}_{\text{sum of the given array}} - \underbrace{(1 + 2 + 3 + 4 + 5)}_{\text{sum of uniq elements}}$$

Hare And Rabbit

R T

[5 , 1, 2, 3, 2, 4]

1 2 3 4 5 6



$S_R = 2 \times S_T$

R

[3 , 1, 3, 2, 4]

T

1 2 3 4 5

TR

[1 3 4 2 2]

R

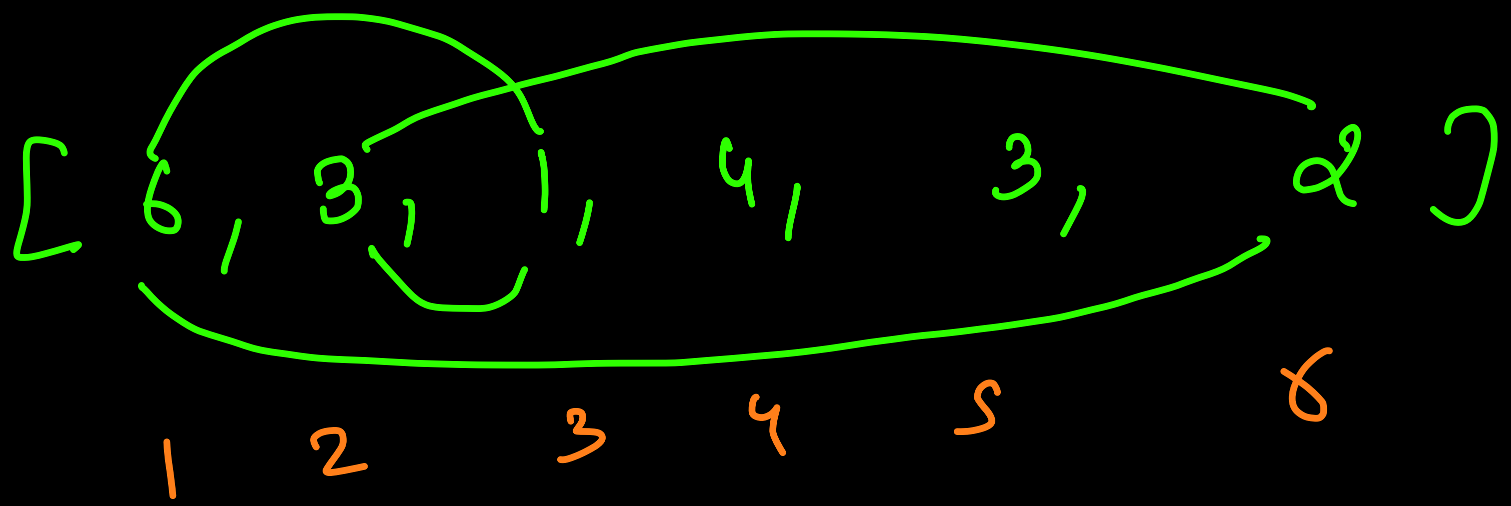
[6, 3, 1, 4, 3, 5]

7

~~1~~ ~~4~~

~~3~~ 4

1 2 3 4 5 6



iteration ⇒ 1

iteration 2 ⇒

iteral 3

iter 4

floyd

floyd

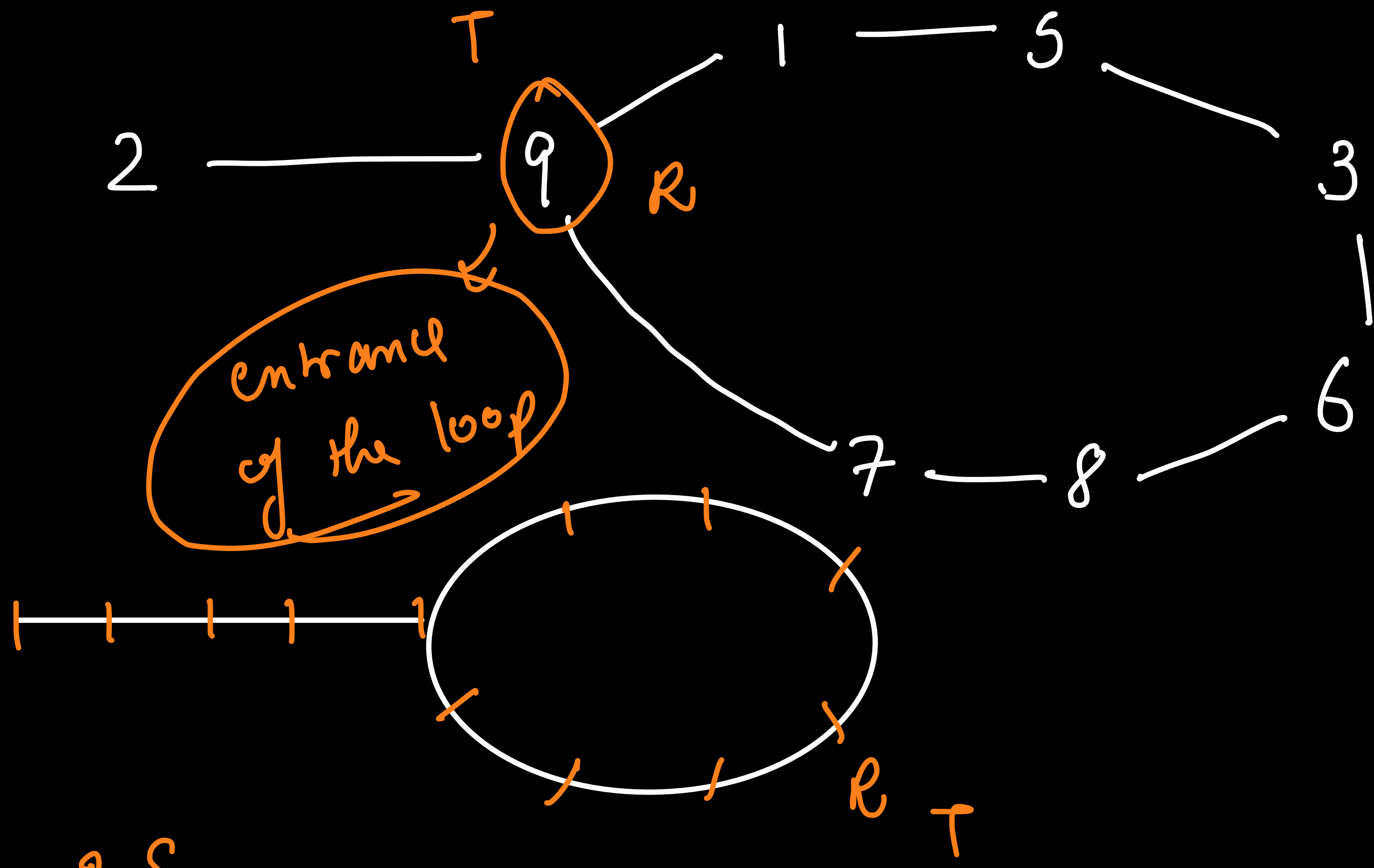
Hare & Rabbit

0	1	2	3	4	5	6	7	8	9
2	5	9	6	9	3	8	9	7	1
↑									

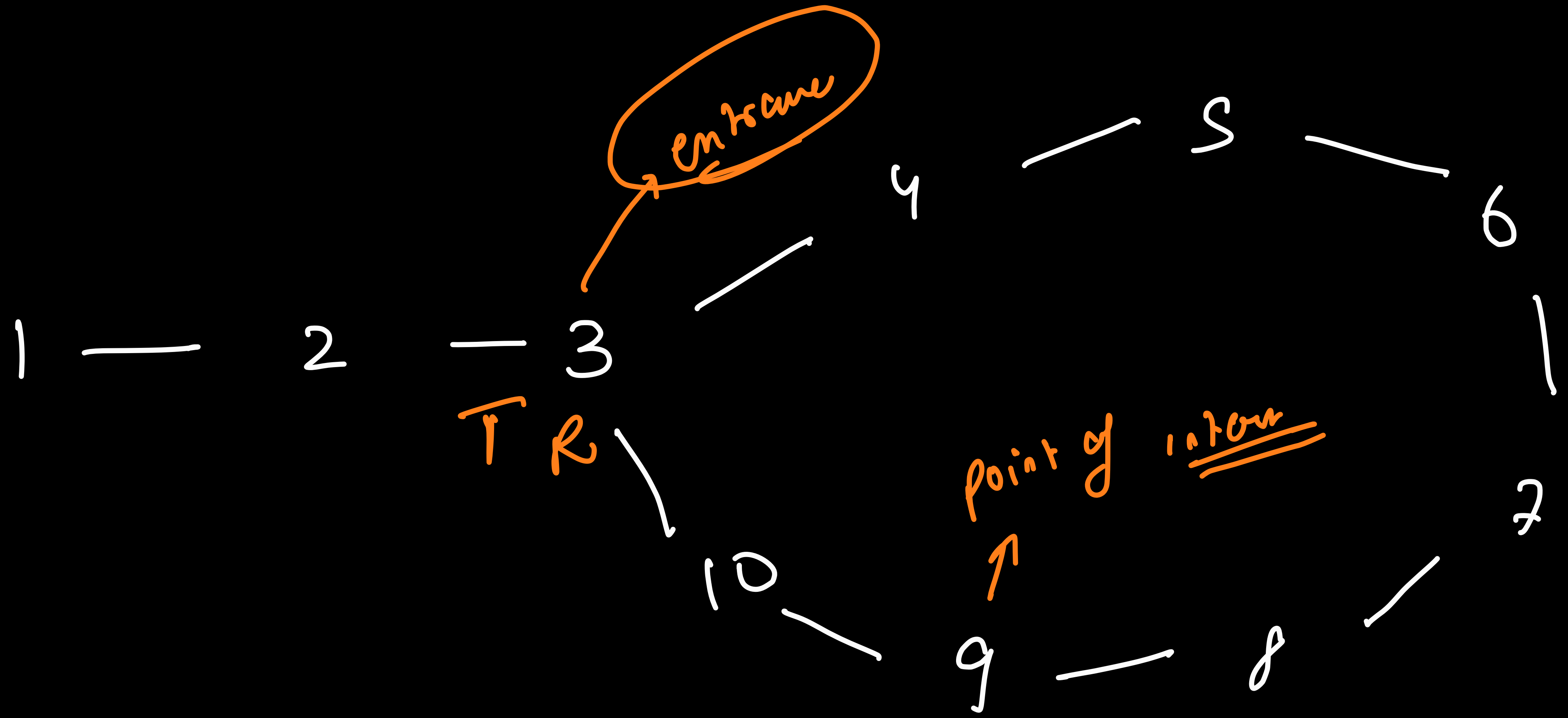
num

↓

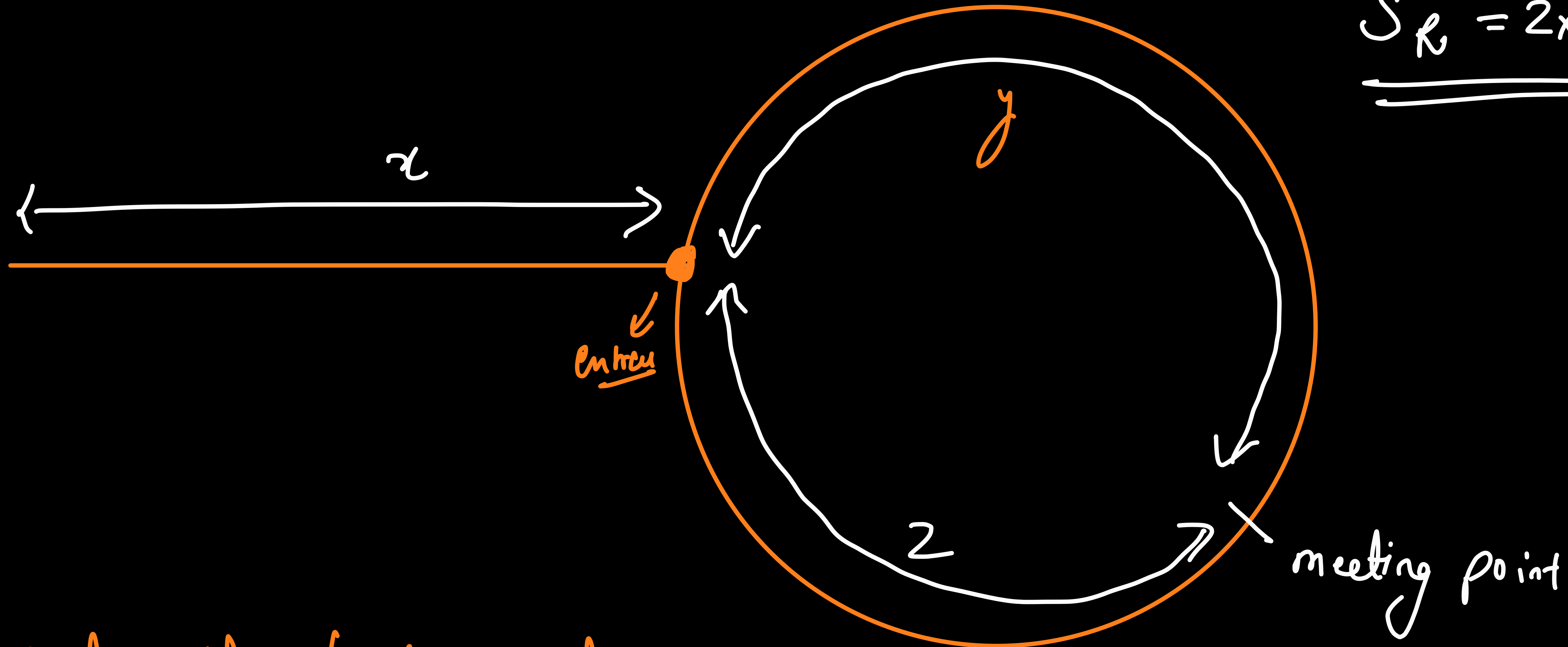
arr[num]



$S_R = 2 S_T$


entrance
1
2
3
4
5
6
7
8
9
10
T
R
point of intersection

$$S_R = 2 \times S_T$$



Consider they both collide in 1 looping

$$d_T = x + y$$

$$d_R = x + y + z + y$$

$$S = \frac{d}{T}$$

$$d_R = 2 \times d_T$$

$$S_R = 2 \times S_T$$

$$x + y + z + y = 2(x+y)$$

$$x + \cancel{2y} + z = 2x + \cancel{2y} \rightarrow \boxed{x = z}$$

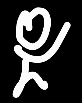
```
-->   find duplicate (arr) {
          t = arr[0];
          r = arr[0];
          do {
               t = arr[t];
               r = arr[arr[r]]
          } while ( t != r );

          t = arr.[0];
          while (t != r)
               t = arr[t];
               r = arr[r];
          }
          return t;
      }
```

$O(n)$

$O(1)$

L, R,
U, D

floyd ✗



R	R	D	U
U	L	L	R
R	D	U	D
L	D	R	R

O(1) space

count=0

n*m

≤ n*m turn

Bruteforce

$1 \longrightarrow n$

$O(n^2)$

```
for( i=1; i<=n ; i++ )
    for (j=0 ; j < a.length ; j++)
        if(a[j] == i){
```

minimum → O(n)

maximum → O(n)

min —— max

⟶

freq Mapping

for (i=min ; i<=max ; i++)
 if (i≤0) continue

O(max-min) TLE

O(n)

for(i=1 ; i≤n+1, i++) → O(n)

O(n)

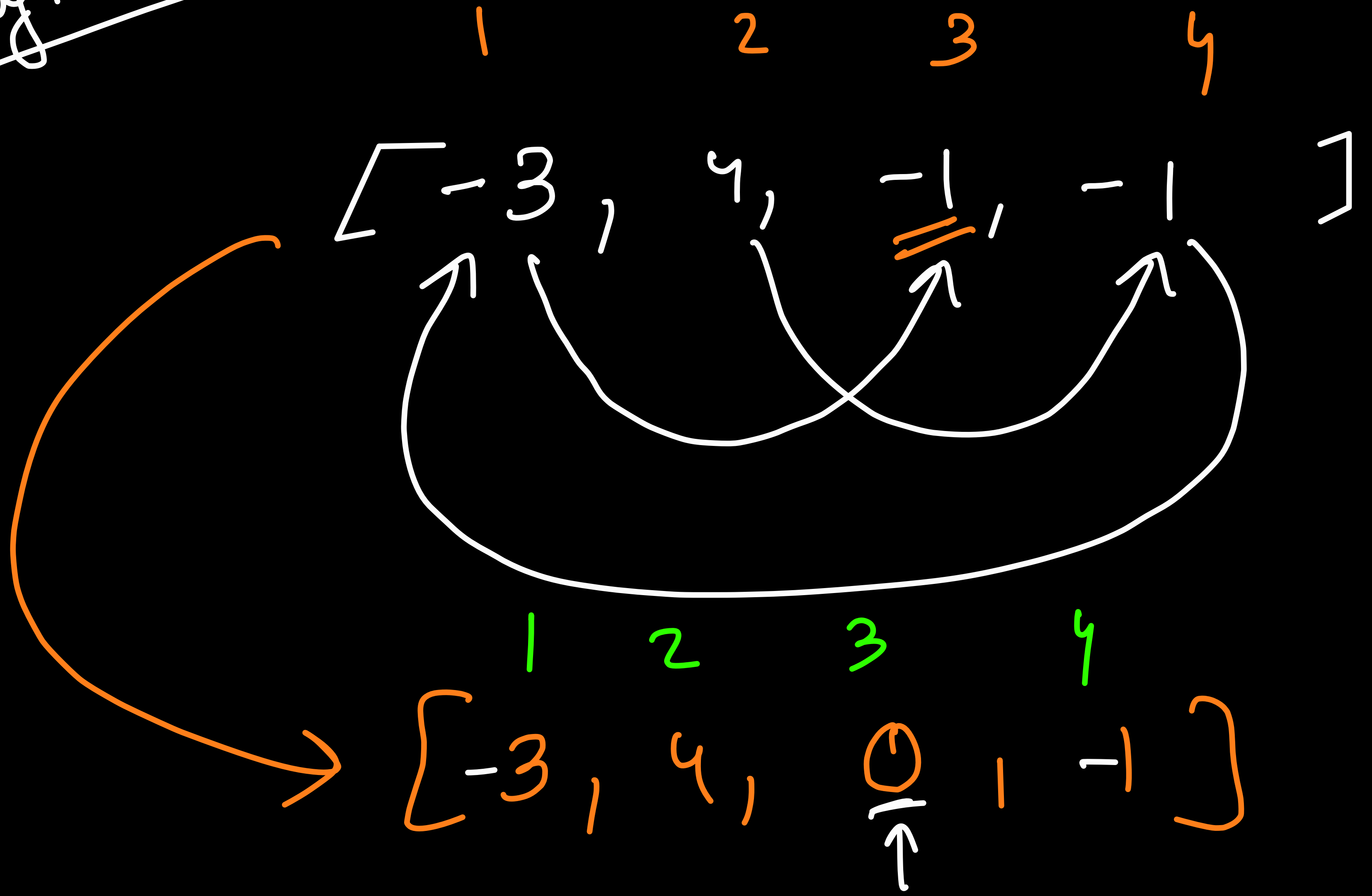
[1, 2, 3] $\textcircled{4}$

$\underline{\underline{[1, n+1]}}$

[2, 1, 3, 4, 5, 6, 7] 7

[1, 8]

modification



$$\begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ [-3, & -4, & \cancel{7}, & -1, & -1, & 5, & 1, & 1, & 2, & 1] \\ & & 0 & & & & & & & \end{array}$$

$\underline{\underline{6}}$

if we made $\underline{\text{this a -ve}}$

$|-1| \rightarrow \textcircled{1}$

$\underline{O(1)}$

$\underline{\underline{O(n)}}$