

Day 1.

LeetCode

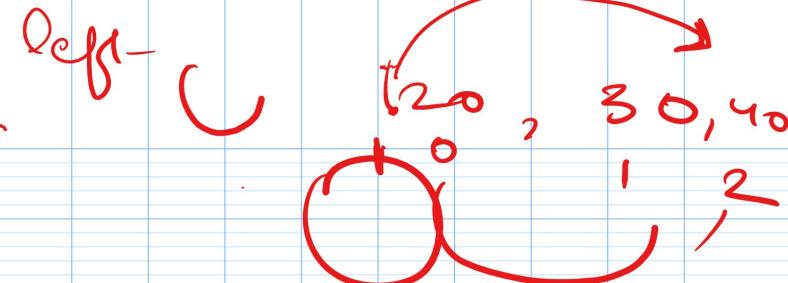
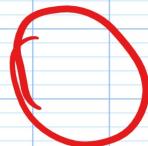
Practice

- RUBLEEN KAUR

Array Rotation.

① left- array rotation

② Right array



arr: [20, 30, 40]

[30, 40, 20] $\xrightarrow{\text{rotate}}$

left . , $k = 1$

count of
rotation.

[20, 30, 40]

right →

temp < ^{Brute force}_{Approach.}
single Element] rotate.

④ Reversal Algo

Juggling Algo

$\text{arr}[] = \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix}$ [N = arr.length.]

K = 2.

Algo.

① start

②

Reverse the entire array.

③

Reverse first 'K' elements.

④

Reverse remaining N-K element.

⑤

End / STOP

①

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

: arr =

$\boxed{1, 2, 3, 4, 5, 6, 7}$

②

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

$\boxed{K = 3}$

7

③

5, 6, 7, 1, 2, 3, 4 } off.

common

$O(n)$

$\frac{O(k)}{O(n-k)}$

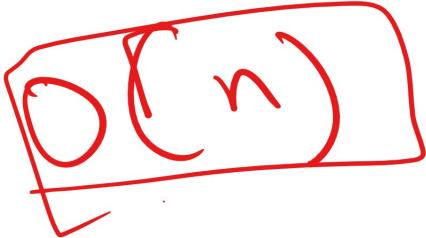
← ① Reverse
← ② $-k$
← ③ $\underline{n-k}$.

constants

= $O(k(1) + n(\delta) + \bar{n}(1))$

~ $O(k + 2n)$ $n \gg \text{large}$

constant

~  ✓ $O(2n)$

~ $O(n)$ ✓

(\neq)

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

$= \neq$

(\neq)
⇒

```

public static void reverse(int[] arr, int start,int end) // 0 end = length
{
    while(start < end)
    {
        int temp= arr[start];
        arr[start] = arr[end];
        arr[_end_] = temp;

        start++;
        end--;
    }
}

```

temp = 2.

