

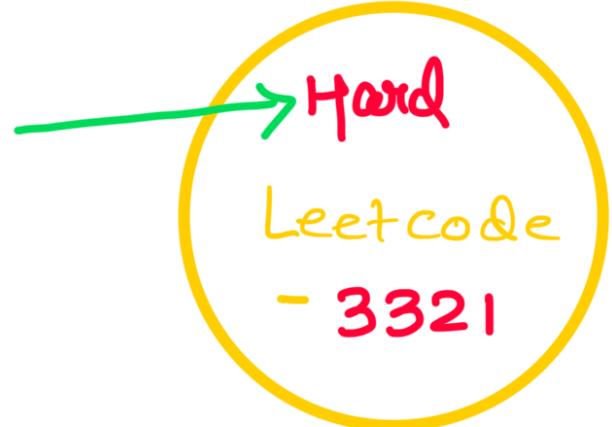
SLIDING

video - 45



WINDOW

MECHANISM...



codestorywithmiK



Cs withMIK



codestorywithMIK



↑
Troy this channel to
see 'Life behind the Scenes'
+
Tech News

Motivation :-

X Something seems difficult because you already give up in the beginning.

If you stay calm, read it, try it, you will realise something can be done.

MIK

3321. Find X-Sum of All K-Long Subarrays II

Hard Topics Companies Hint

You are given an array `nums` of `n` integers and two integers `k` and `x`.

The **x-sum** of an array is calculated by the following procedure:

$1 \leq n \leq 10^5$

- Count the occurrences of all elements in the array.
- Keep only the occurrences of the top x most frequent elements. If two elements have the same number of occurrences, the element with the **bigger** value is considered more frequent.
- Calculate the sum of the resulting array.

10 <= x <= k <= n

Note that if an array has less than x distinct elements, its **x-sum** is the sum of the array.

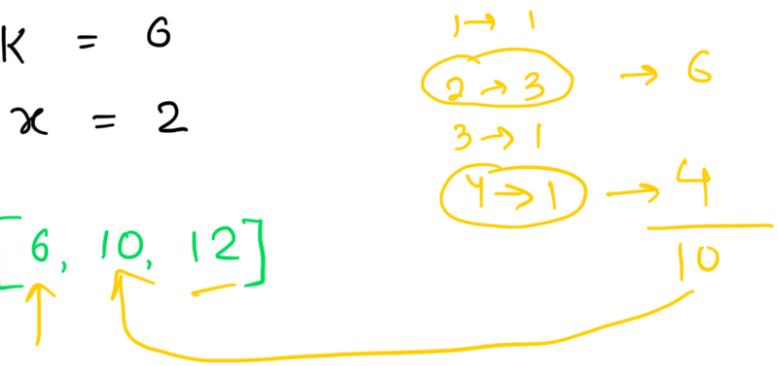
Return an integer array `answer` of length $n - k + 1$ where `answer[i]` is the **x-sum** of the subarray `nums[i..i + k - 1]`.

Example :- $\text{nums} = [1, 1, \underline{2}, \underline{2}, 3, \underline{4}, \underline{2}, 3]$

$$K = 6$$

$$x = 2$$

Output : $[6, 10, 12]$



Thought Process

$\text{nums} = [1, 1, 2, 2, 3, 4, 2, 3]$

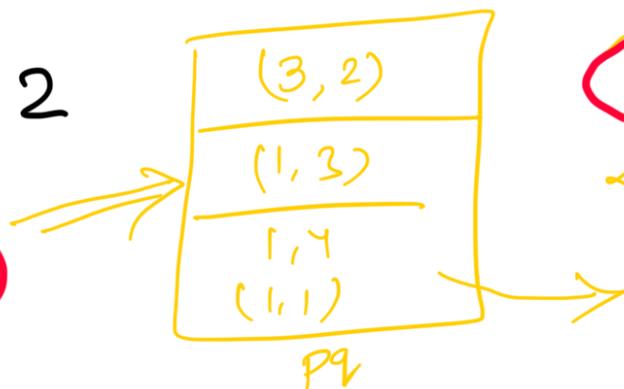
$O(n)$

Sliding w

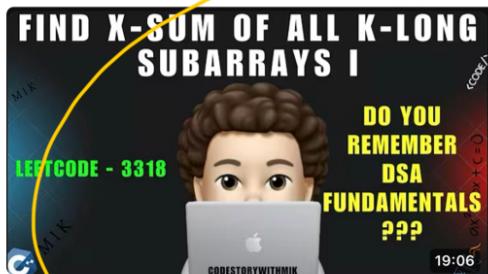
$$K = 6$$

$$\chi = 2$$

$K \log K$



map
func(map, χ)
<freq, elem>



Find X-Sum of All K-Long Subarrays I | Revise DSA Fundamentals | Dry Run | Leetcode 3318 | MIK

codestorywithMIK

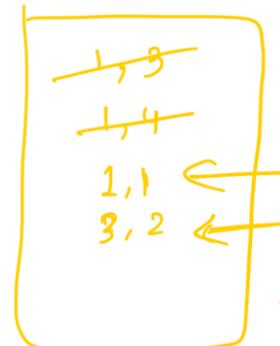
① Sliding window \rightarrow subarray $\rightarrow O(n)$

Improvement in funct \rightarrow top- χ element

$\text{nums} = [\cancel{1}, 1, 2, 2, 3, 4, \cancel{2}, 3]$

$K = 6$

$x = 2$



Data Structure \rightarrow ordered fashion.

$\langle \text{frequency, element} \rangle$

+ Search the element easily.

Ordered Set \rightarrow

Set $\langle \langle \text{freq, element} \rangle \rangle$

Will Set Data Structure Help me ???

nums = [~~1, 1, 2, 2, 3, 4, 2, 3~~]
 $O(n)$

K = 6

x = 2

$\langle 2, 1 \rangle$
 $\langle \text{freq, element} \rangle$

map	
value	frequency
1	≥ 1
2	≥ 3
3	1
4	1

Set { (1, 1), (1, 3), (1, 4), (3, 2) } $O(x)$

set.find (2, 2) $\rightarrow \log(n)$

Sum $\rightarrow O(x)$.

$n * (\frac{x}{10^5 + 10^5})$

Find Sum Optimally

Set $\{(1, 1), (1, 3), (1, 4), (3, 2)\}$

$O(\log(x)) \Leftarrow O(2)$

Sum = constant $\in O(1)$

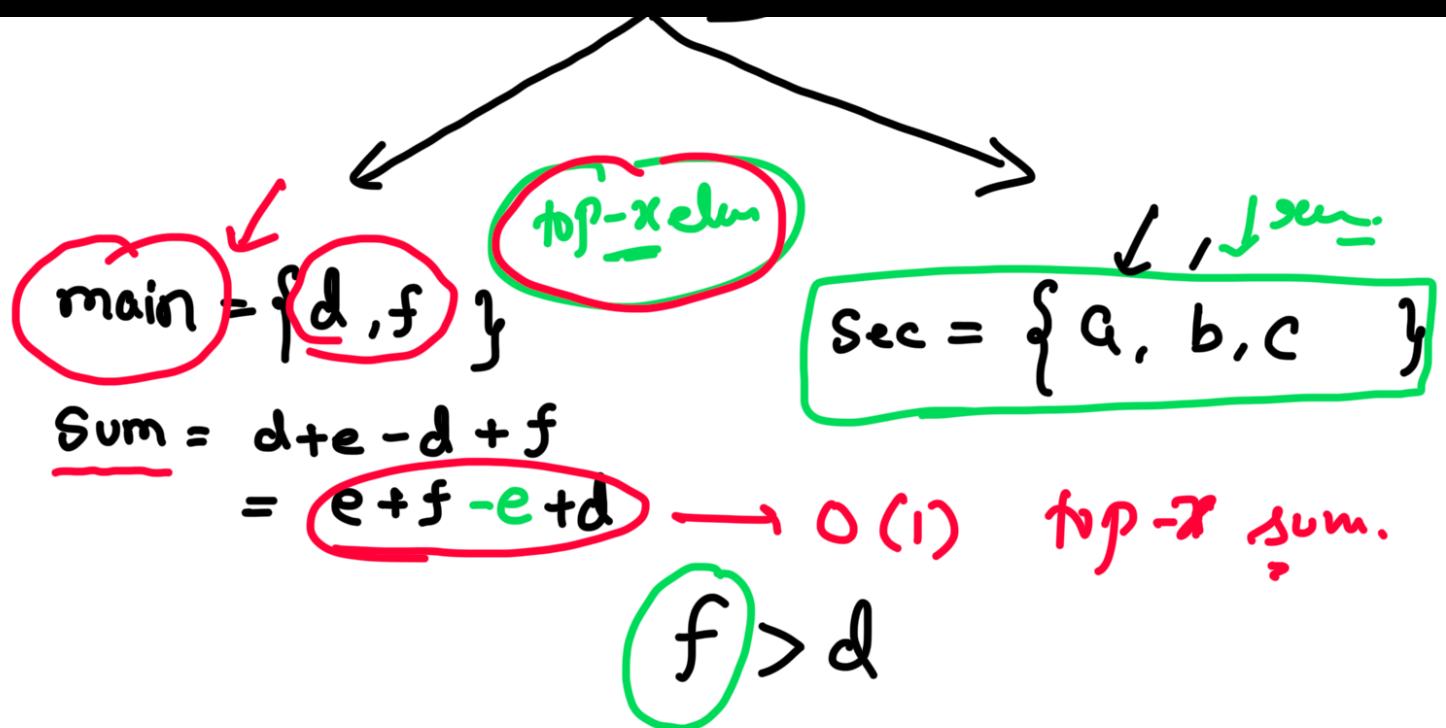
Sum = 0

* Find the median of a stream :-

$n = 2$

~~Set = {a, b, c, d, e}~~

Keeping 2 sets.



Complete Dry Run.

nums = [1, 1, ~~2~~, ⁱ 2, 3, 4, 2, 3] _j

$$K = 6$$

$$x = 2$$

(2, 2)

value	map	frequency
2		β2
3		2

main = $\{(2, 2), (2, 3)\}$
sec = $\{(1, 4)\}$

$$\text{Sum} = 4 + 2 - 2$$

$$+ 4 = 8 - 4 = 4$$

$$+ 3 = 7 - 3 = 4$$

$$+ 6 = 10$$

$$10 - 4 = 6 + 6 = 12$$

$$12 - 6 = 6 + 4 = 10$$

$$10 - 4 = 6 + 4 = 10$$

result = $\{6, 10, 12\}$

$O(1)$.

(*) Sliding window.

(*) main set \rightarrow x maintained size
 \downarrow sum maintained

(*) sec \rightarrow remaining $\langle \text{freq}, \text{elem} \rangle$

DS

medium

DA

Pg . Pg

