Subarray Sum equal to K

int[] arr = {10, 2, -2, -20, 10} , K = 10

3 - subarrays

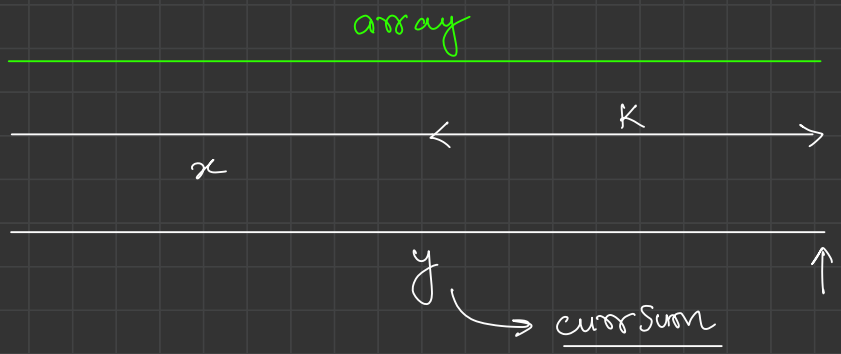
Brute force

↳ get all subarrays sum and check sum == k, inc. ans.

TC: $O(N^2)$
SC: $O(1)$

int[] arr = { 0, 0, 10, 2, -2, -20, 10 }, K = 10

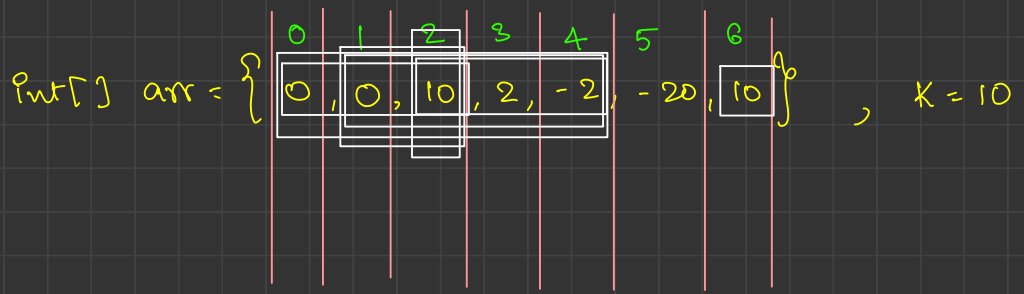
(0 1 2 3 4 5 6)



$x + K = y$

$x = y - K$

find



0 0 0 10 12 10 -10 0

$x = 12 - 10 = 2$

$x = 10 - 10 = 0$

$x = 0$

$x = -10 - 10 = -20$

$x = 0 - 10 = -10$

numSubarray = ~~0~~ ~~3~~ ~~4~~ 7

freq

0 ⟶ ~~4~~

10 ⟶ ~~2~~

12 ⟶ 1

-10 ⟶ 1 ✓

Group Anagrams

arr[] = { "cat" (1), "dog" (2), "tac" (1), "god" (2), "act" (1) }

o/p "cat, tac, act, dog, god"

S_1 and S_2

↳ when no. of occ of each char same.

(1) Sort(s1) and sort(s2)

↳ equals → anagrams

TC: NlogN
SC: O(1)

② Create freqancy map

String S1 → freqmap of char

String S2 → freqmap of char

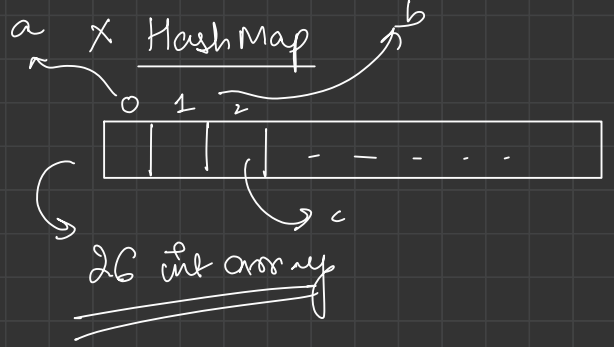
TC : O(N)
SC : O(26) ≈ O(1)

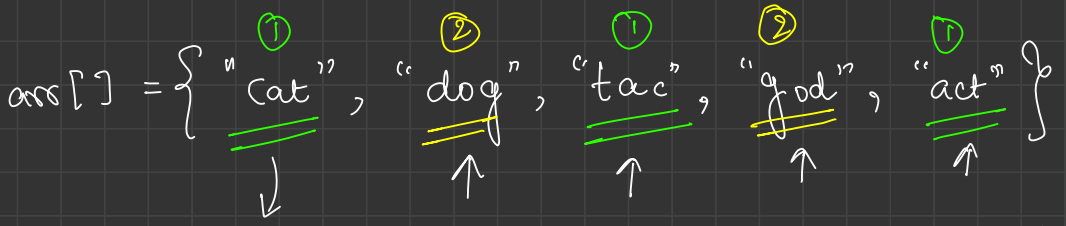
is freq of each char same

↳ Anagramic

26 Alphabets

space O(26) → O(1)





freqMap

a → 1
c → 1
t → 1

code → "a1c1t1"

"a1c1t1"

code

"a|c|t|"

d|g|o|

groups

cat, tac, act

dog, god

"d|g|o|"

Minimum window substring

str1 d b a e c b b a b d c a a f b d d c a b g b a

str2 a b b c d c

d → 4
b → 7
a → 6
e → 1
f → 1
g → 1
c → 3

a → 1
b → 2
c → 2
d → 1

Brute force
↳ generate all subarray, then check is str2 part of str1, get min length

TC: $O(N^3)$

exc

Str1 d b a e c b b a b d c a a f b d d c a b g b a

inc

Str2 a b b c d c

mct = ~~0 1 2 3 4 5~~ 6

Freq Map

a → 1
b → 2
c → 2
d → 1

dmct
6 imp char

Freq Map

d → 1
b → ~~1 2~~ 3
a → ~~1~~ 2
e → 1
c → ~~1~~ 2