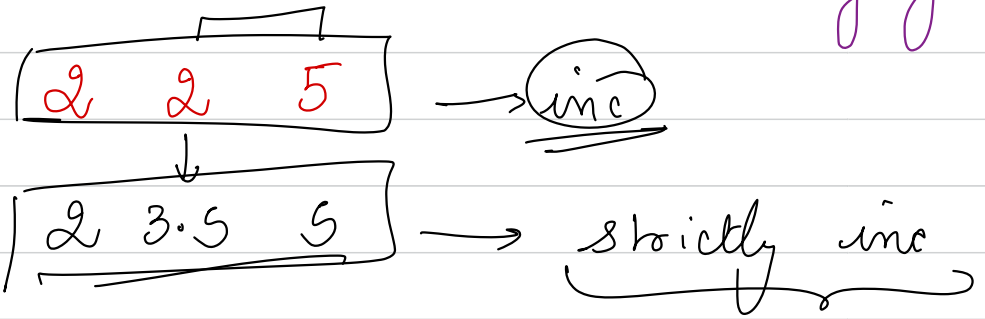
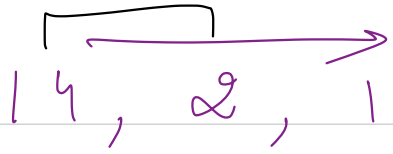
Q →

A_1	A_2	A_3	------	A_N

$(i), (i+1)$ → replace any one of them by avg of two



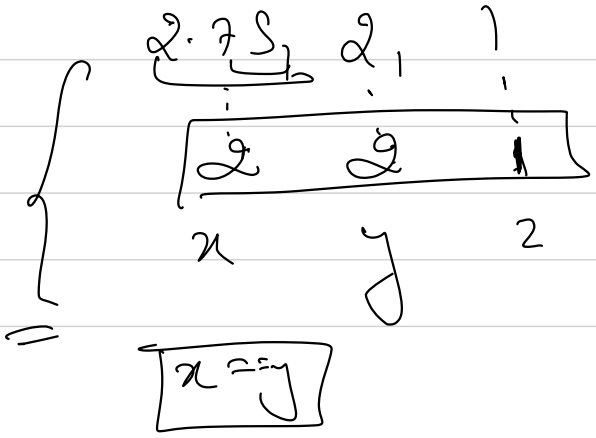
inc ⟶ strictly inc Seq

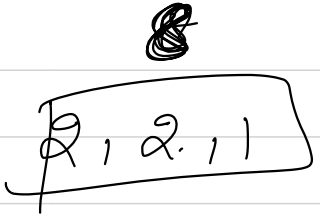


8, 2, 1

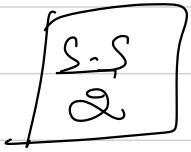
5, 2, 1

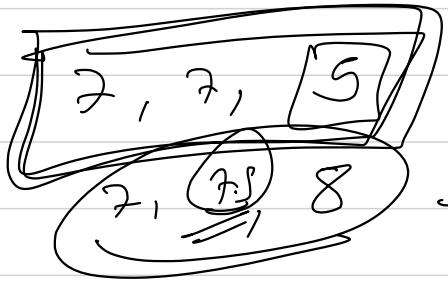
3.5, 2, 1

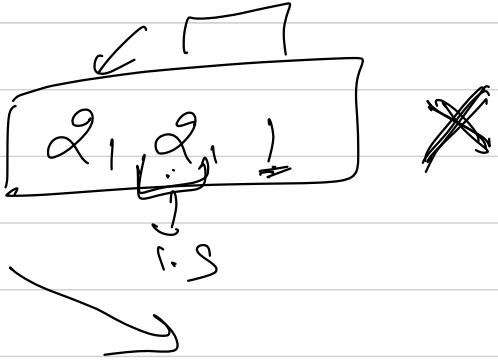


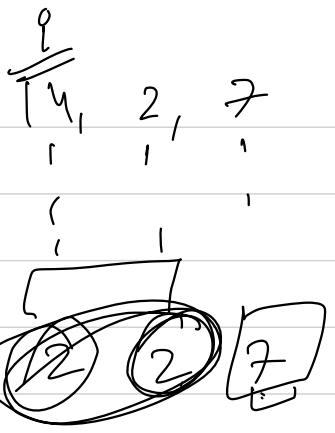


$\frac{2+1}{2} \Rightarrow \frac{3}{2}$ 1.5









$4, 2, 7$

$(a_i == a_{i+1})$

$a_{i+2} \leq a_i$ ✗

$2, 2, 2$

increasing → Yes

2, 2, 2

strictly inc → Yes

8, 8, 12, 19

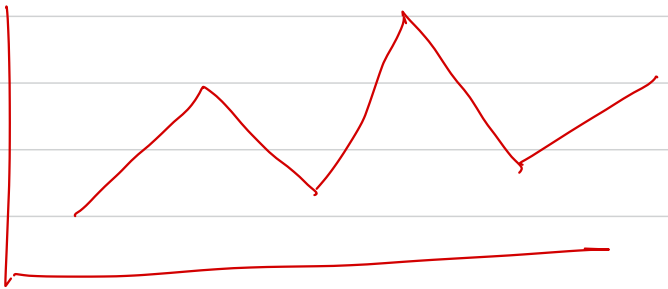
equal → No

8, 10, 12, 19

dec / strictly dec → No

Yes

Random

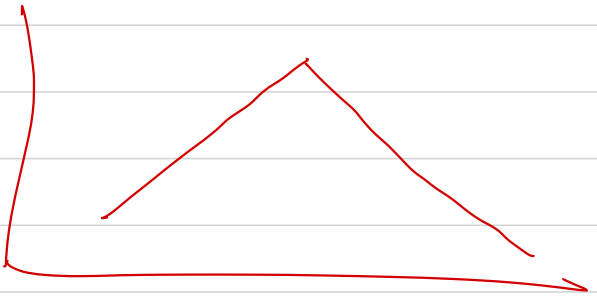


17, 8, 12, 19

12.5, 8, 12, 19

10.25, 8

9.15



4, 6, 8, 3, 1.

4, 4, 8, 3, 1

4, 4, 4 3, 1

4, 4, 4, 3, 1

4, 6, 8

4, 5, 8

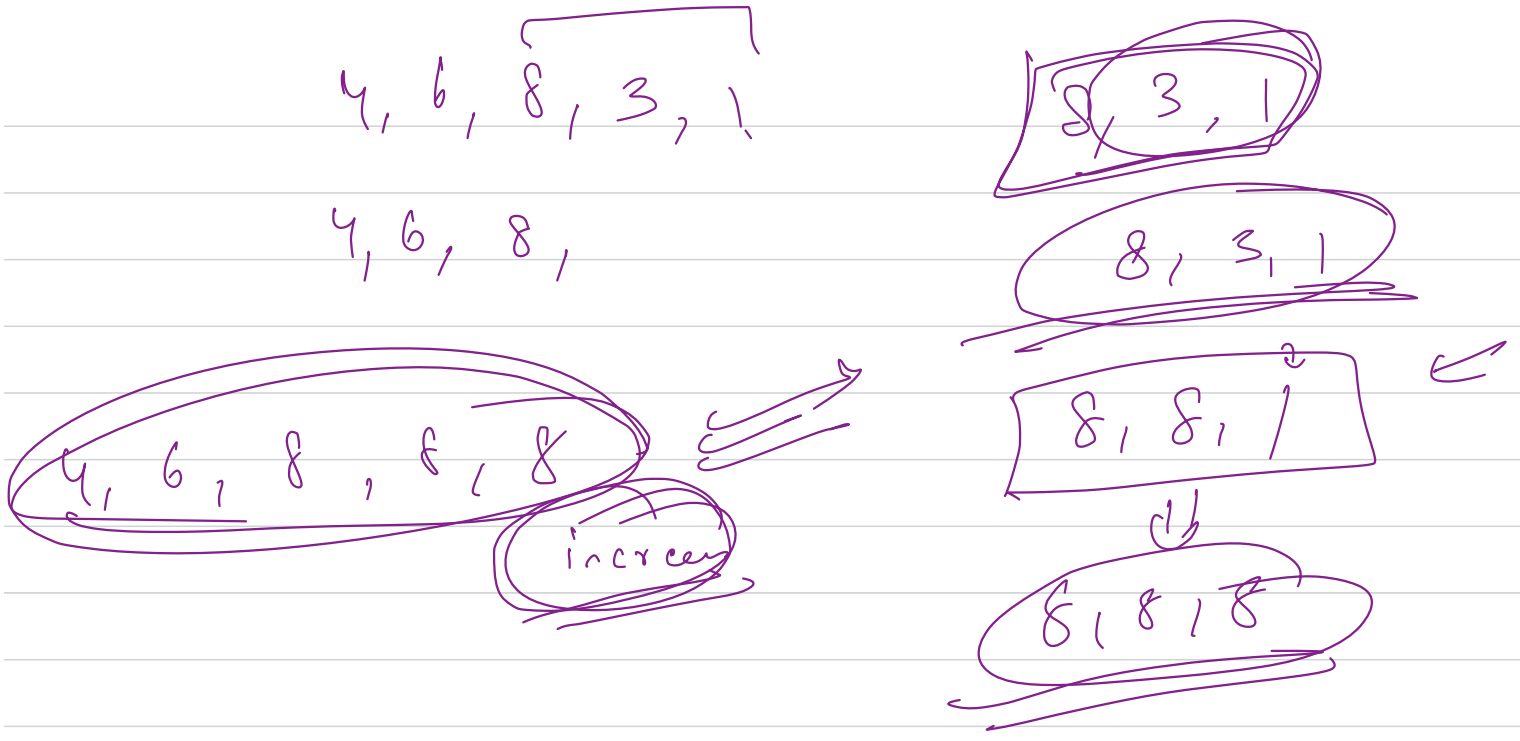
4, 4, 5, 8

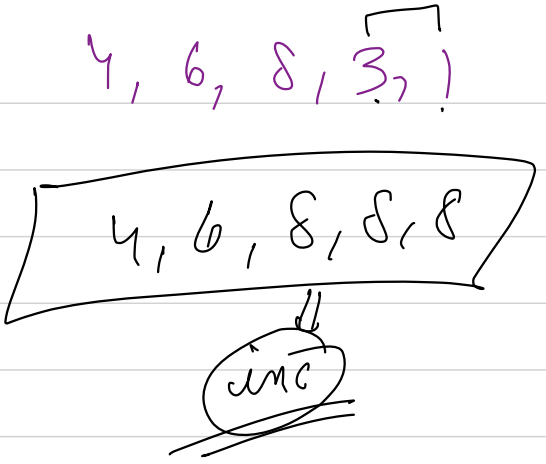
4, 4, 8

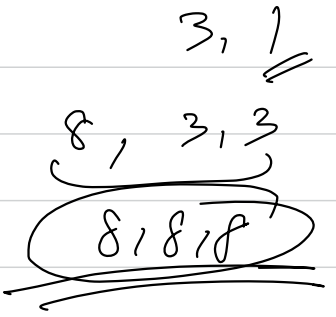
4, 4, 6

4, 4, 5

4, 4, 4, 5


4, 6, 8, 3, 1
4, 6, 8,
4, 6, 8, 8, 8
increa
8, 3, 1
8, 3, 1
8, 8, 1
8, 8, 8


4, 6, 8, 3, 1
4, 6, 8, 8, 8
inc


3, 1
8, 3, 3
8, 8, 8

2 pointers

Q⇒ Given an array of n integers. find the no. of subarrays such that the sum of elements of the subarray is atmost K.

7, 1, 3

K = 5

1, 3

1

3

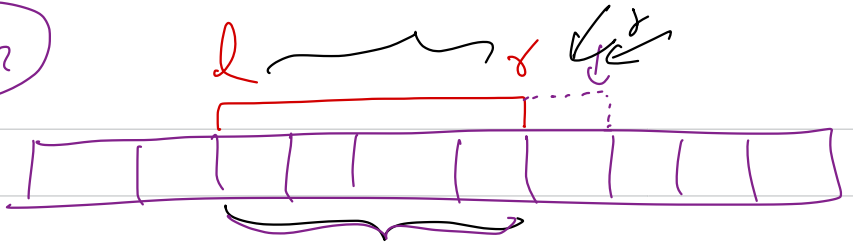
3 ans

$n \leq 10^5$

$K \leq 10^{18}$

$1 \leq a_i \leq 10^9$

$(x+1)(x+2)$?? $\neq$



x ← $\sum_{i=l}^{r} a[i] \leq K$ → satisfies the condition

$[l, r+1]$

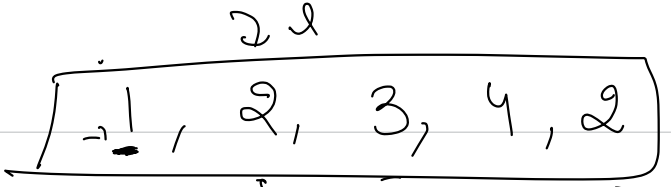
$\sum_{i=l}^{r} a[i] + a[r+1] \leq K$

expand the subarray

$x += (j - l + 1)$

$j = r + 1$

$\sum_{i=l}^{r} a[i] > K$ → Shrink by inc → l points



$K = 11$

Sum = ~~1 3 6 10 12~~ 11

3 ≤ K

res = ~~0 1 3~~ 6 10

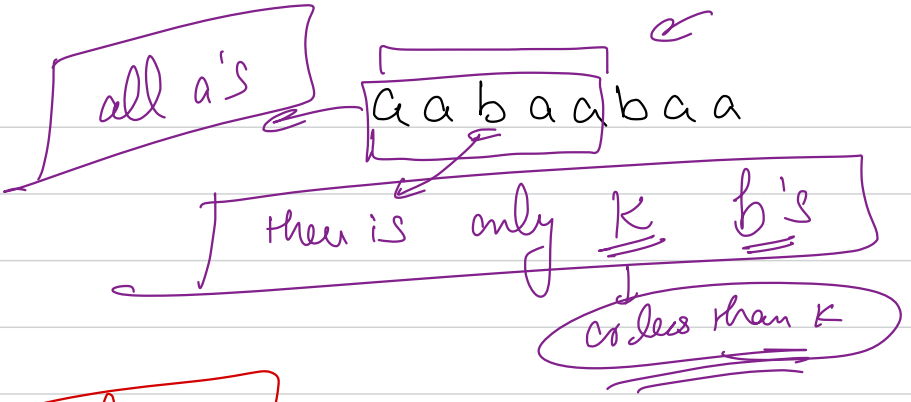
to represent a window we can use 2 pointers

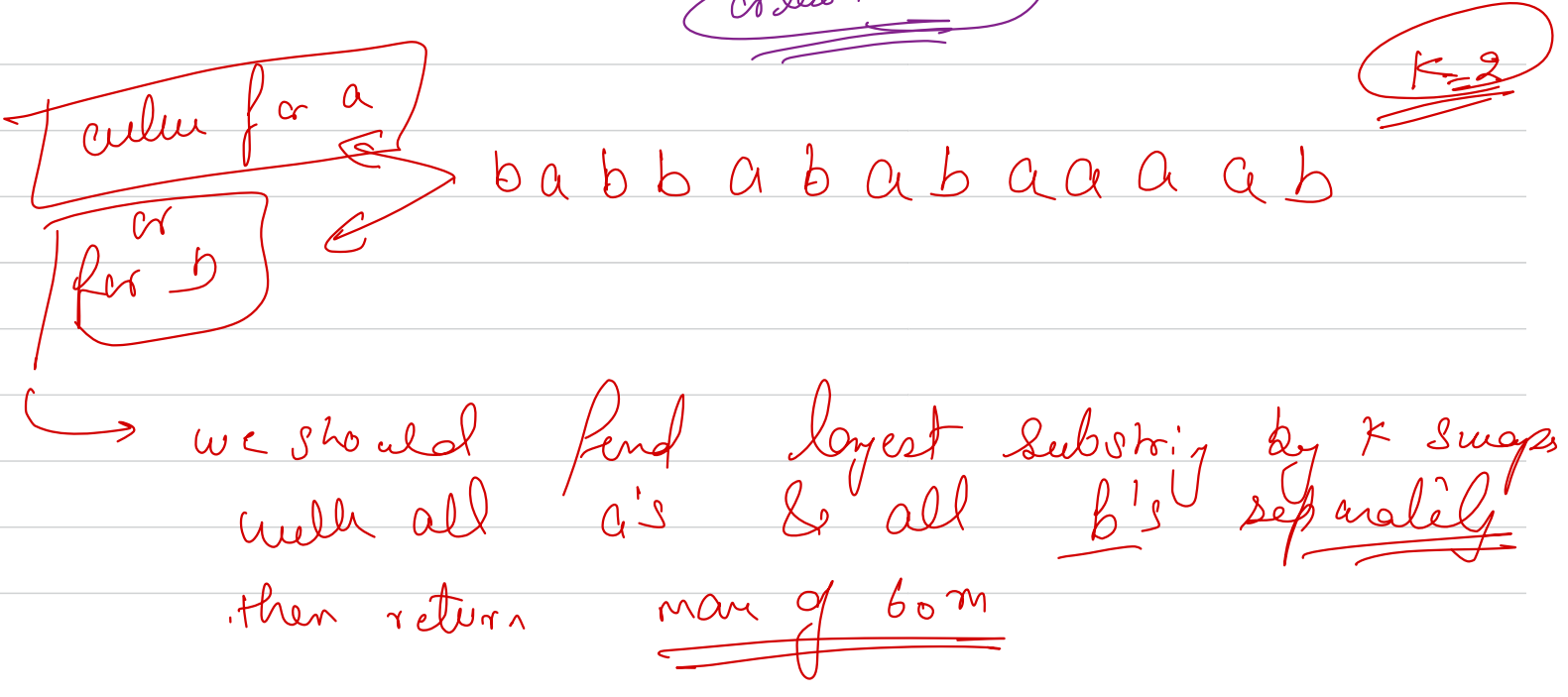
[1]
[2]
[1, 2]
[3]
[2, 3]
[1, 2, 3]

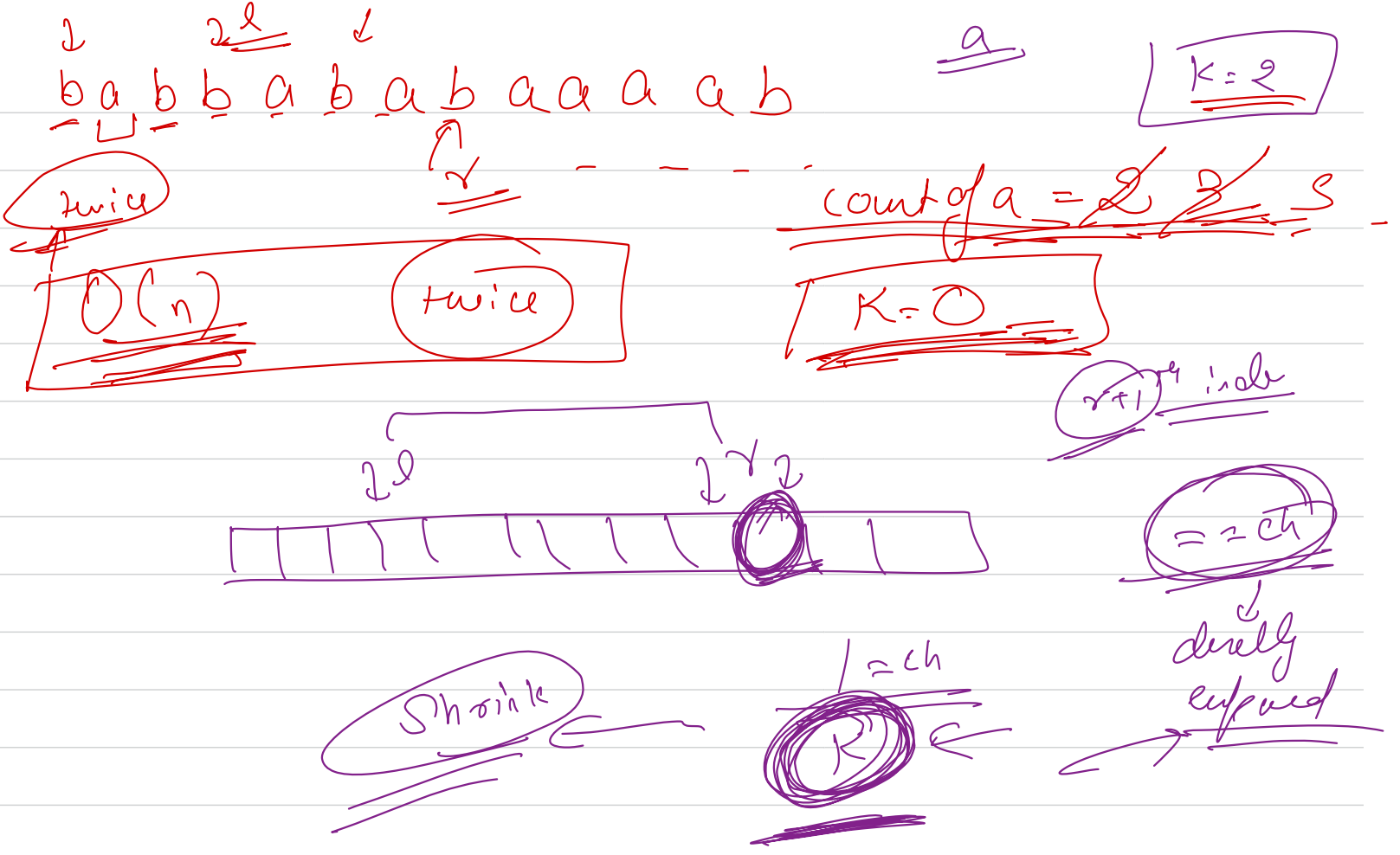
4
3, 4
2, 3 4
1, 2, >

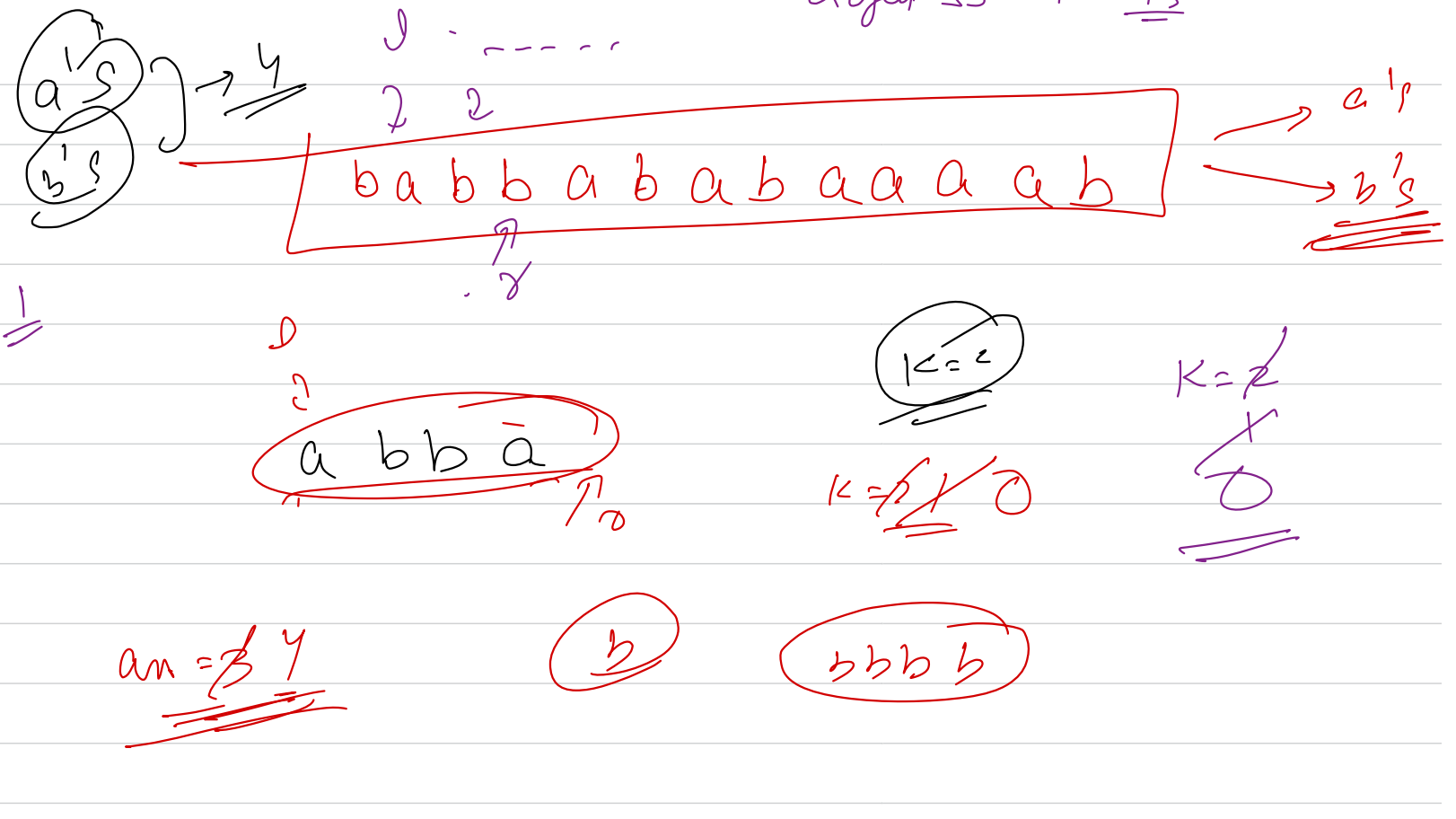
2
4, 2
3, 4, 2
2, 3, 4, 2

res += (r - l + 1)


all a's
a a b a a b a a
then is only K b's
or less than K
K=1
a


value for a
or
for b
b a b b a b a b a a a a b
K=2
we should find longest substring by k swaps
with all a's & all b's separately
then return max of both

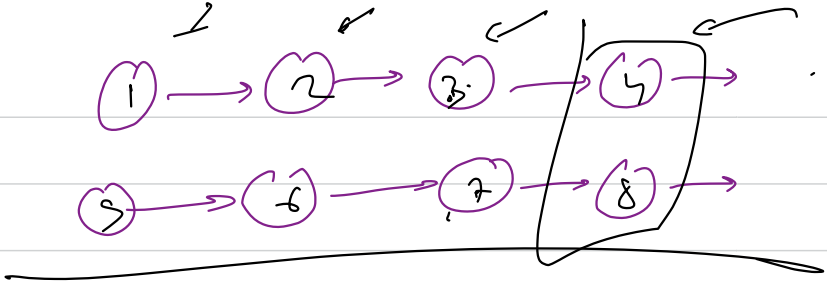

a
K = 2
twice
count of a = 2 3 5
O(n)
twice
K = 0
r+1 index
l
r
=ch
deadly expand
Shrink
!=ch


Largest ss with a's
a's
b's
babbabababaaaab
a's
b's
abba
aabbbbb

Ch → a
   → b

```
l = 0, r = 0
while (r < n) :
    while (r < n and str[r] == ch):
        ans = max(ans, r - l + 1)
        r += 1
    while (r < n and str[r] != ch and k > 0)
        k -= 1
        ans = max(ans, r - l + 1)
        r += 1
    while (l <= r and str[r] != ch and k < 0)
        if (k == 0)
            l = r
            r += 1
        k += (str[l] != ch)        l += 1
```

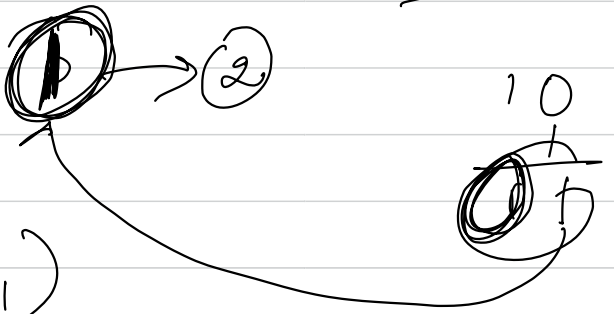
Q →



1 1 1 1
1 2 3 4
5 6 7 8

1

1 2



(1, 1)