

Google
Amazon

PSS

20th Nov

Sunday

11~1 pm

1) find sum of all subarrays of length K .

$K=5$

$arr[10] =$ 0 1 2 3 4 5 6 7 8 9
 -3 4 2 5 3 -2 8 2 -1 4

S	e	sum
0	4	11
1	5	12
2	6	16
3	7	16
4	8	10
5	9	11

Idea: for every subarray of length K , iterate and get the sum.

Tc: $O((N-K+1) \times K)$

$S = 0, e = K - 1$

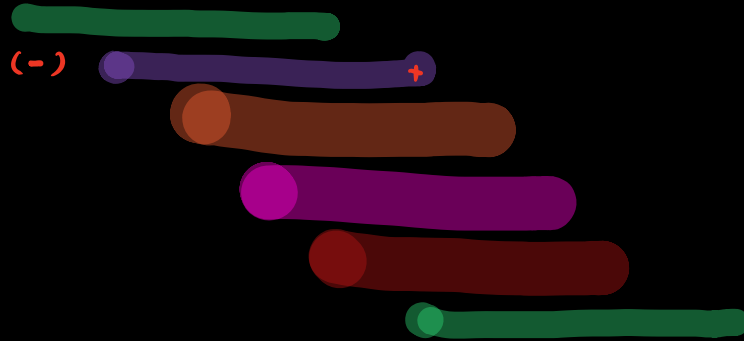
while ($e < N$)

$\left\{ \begin{array}{l} \text{sum} = 0 \\ \text{for } (i = S; i \leq e; i++) \\ \quad \text{sum} = \text{sum} + \text{arr}[i] \\ \text{Print}(\text{sum}) \\ S++ \\ e++ \end{array} \right.$

SC: $O(N)$

arr[10] =

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



sum = 0

for(i=0 ; i <= k-1 ; i++)

{ sum = sum + arr[i]

print(sum)

s = 1

e = k

while(e < n)

sum = sum - arr[s-1]

sum = sum + arr[e]

print(sum)

s++

e++

TC: $O(N - k + k)$

TC: $O(N)$

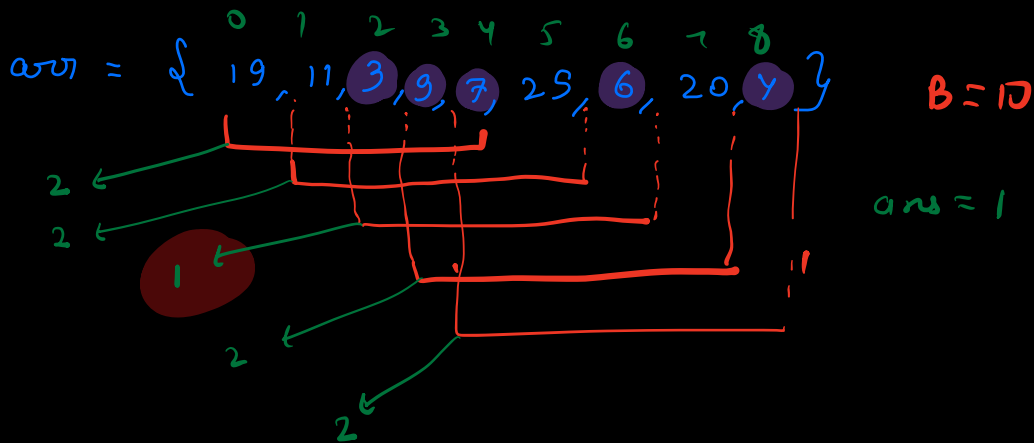
SC: $O(1)$

Q. Given an array $[n]$ & a number B .

Find and return min no. of swaps required
to bring all numbers $\leq B$ together.

Ex: arr = [1, 12, 10, 3, 14, 10, 5] $B=8$

ans = 2

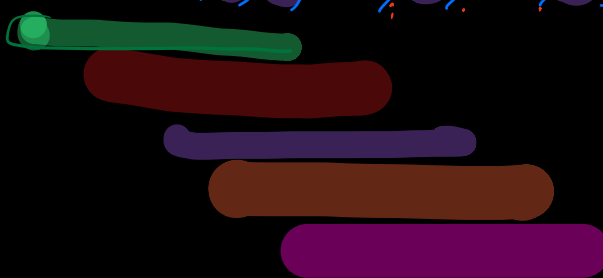


arr = { 19, 11, 3, 9, 7, 25, 6, 20, 4 }

tot ele $C=B$

↓

5 - element of
 $\leq B$



① window size

```

ws = 0
for (i = 0; i < N; i++)
    if (arr[i] ≤ B) ws++

```

no of elements ≤ B in curr window
 $\rightarrow cnt = 0$

```

for (i = 0; i ≤ ws - 1; i++)
    if (arr[i] ≤ B) cnt++

```

ans = ws - cnt

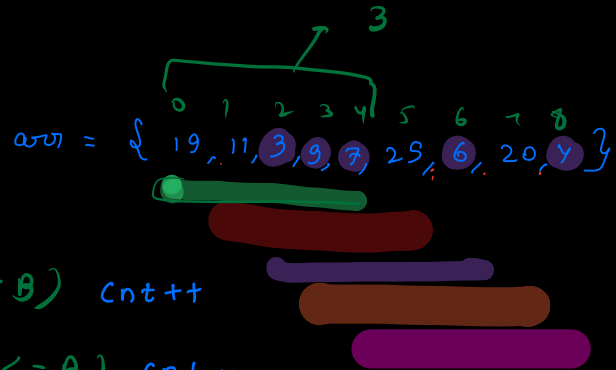
s = 1
e = ws

while (e < N)

```

{
    if (arr[e] ≤ B) cnt++
    if (arr[s-1] ≤ B) cnt--
    tmpans = ws - cnt
    ans = min(ans, tmpans)
    s++, e++
}
return ans

```



ws N-1

TC : $O(N - ws + ws + N) = O(N)$

SC: $O(1)$

DeShaw

Q. Given a $mat[N][N]$. Print in spiral order (clockwise)

5x5

	0	1	2	3	4
0	1	2	3	4	5
1	6	7	8	9	10
2	11	12	13	14	15
3	16	17	18	19	20
4	21	22	23	24	25

1, 2, 3, 4, 5, 10, 15, 20, 25, 24, 23, 22, 21, 16, 11, 6

7, 8, 9, 14, 19, 18, 17, 12, 13

	0	1	2	3	4
0	1	2	3	4	5
1	6	7	8	9	10
2	11	12	13	14	15
3	16	17	18	19	20
4	21	22	23	24	25

$N = 5$

$mat[5][5]$

$(0,0) \rightarrow (0,1) \rightarrow (0,2) \rightarrow (0,3) \rightarrow (0,4)$
 \uparrow
 $1,0$
 \uparrow
 $2,0$
 \uparrow
 $3,0$
 \uparrow
 $4,0$
 $\leftarrow 4,1 \leftarrow 4,2 \leftarrow 4,3 \leftarrow 4,4$
 \downarrow
 $1,4$
 \downarrow
 $2,4$
 \downarrow
 $3,4$
 \downarrow

$N-1$ times, Print number and go right.

$N-1$ times, Print number and go bottom

$N-1$ times, Print number and go left

N-1 times, Print number and go up.

$i=0, j=0, N=N$

while ($N > 1$)

for ($z=101, z \leq 100+N-1; z++$)

{
 print[win[i][j])
 j++
}

N-1 Right
side

for ($k=1; k \leq N-1; k++$)

{
 print[win[i][j])
 i++
}

N-1 Down
side

for ($k=1; k \leq N-1; k++$)

{
 print[win[i][j])
 j--
}

N-1 Left
side

for ($k=1; k \leq N-1; k++$)

{
 print[win[i][j])
 i--
}

N-1 Up
side

i++, j++, $N = N-2$

if (N == 1)

{ Print C over C[i][j] }

0,0

N=5

N=5

	0	1	2	3	4
0	1	2	3	4	5
1	6	7	8	9	10
2	11	12	13	14	15
3	16	17	18	19	20
4	21	22	23	24	25

i=1 j=1

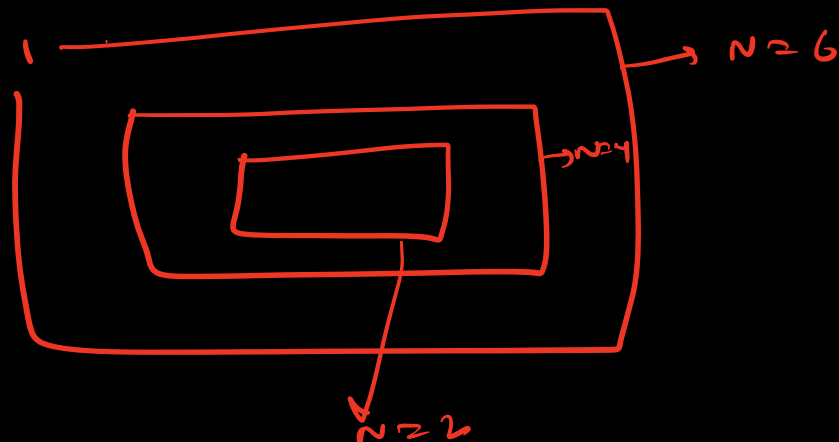
N=3

N=1

T.C: $O(n^2)$ j++ j++

N = N-2

S.C: $O(1)$



$$B=10$$

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
2	12	5	6	7	11	12	13	14	1	0									

Sunday 20th Nov

11am - 1pm