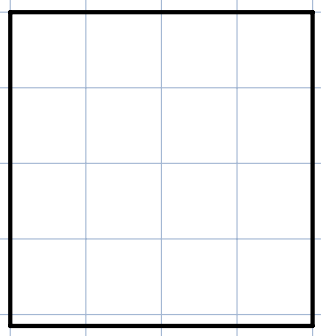


1. Searching in row wise column wise sorted matrix

2. Contribution Technique
   Sum of all submatrices

3. First missing no.

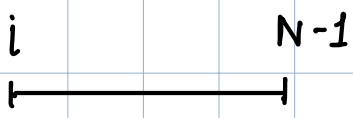
Classes
- Assignments
- Contest with 3 reattempts

After DSA 4 → DSA Mock interview

Q1

Maximum Subarray Sum - Kadane's Algo

Q2



prefix sums

Q3

$O(Q + N)$

Q

Searching in row wise column wise sorted matrix

Observe that rows and columns are both sorted.

$$A = \begin{bmatrix} -5 & -2 & 1 & 13 \\ -4 & 0 & 3 & 14 \\ -3 & 2 & 6 & 18 \end{bmatrix}$$

k = 13 true

k = 2 true

K = 15 false

Brute force :

traverse each element of matrix & find

```
for( i=0 ;   i < N ;   i++) {
    for (j = 0   j < M;   j++) {
        if ( A[i][j] = = K)
            return true
    }
}
return false
```

TC: O(N*M)

SC: O(1)

Observe that rows and columns are both sorted.

inc

$$A = \begin{bmatrix} -5 & -2 & 1 & 13 \\ -4 & 0 & 3 & 14 \\ -3 & 2 & 6 & 18 \end{bmatrix}$$

$$\begin{matrix} -5 & -2 & 1 & 13 \\ -4 & 0 & 3 & 14 \\ -3 & 2 & 6 & 18 \end{matrix}$$

K=13

sm

grea

$$\begin{matrix} -5 & -2 & 1 & 13 \\ -4 & 0 & 3 & 14 \\ -3 & 2 & 6 & 18 \end{matrix}$$

K=2

l

d

discarding the column

j--

discarding the row down

i++

S l

$$\begin{matrix} -5 & -2 & 1 & 13 \\ 4 & 0 & 3 & 14 \\ -3 & 2 & 6 & 18 \end{matrix}$$

d

g

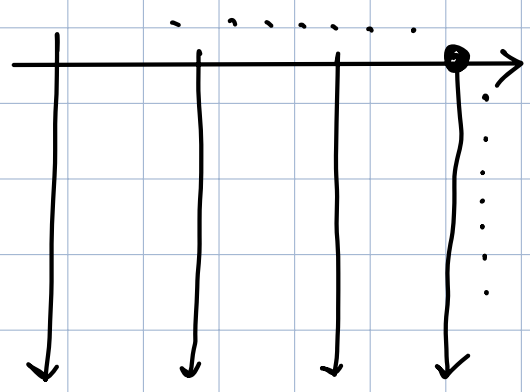
K=0

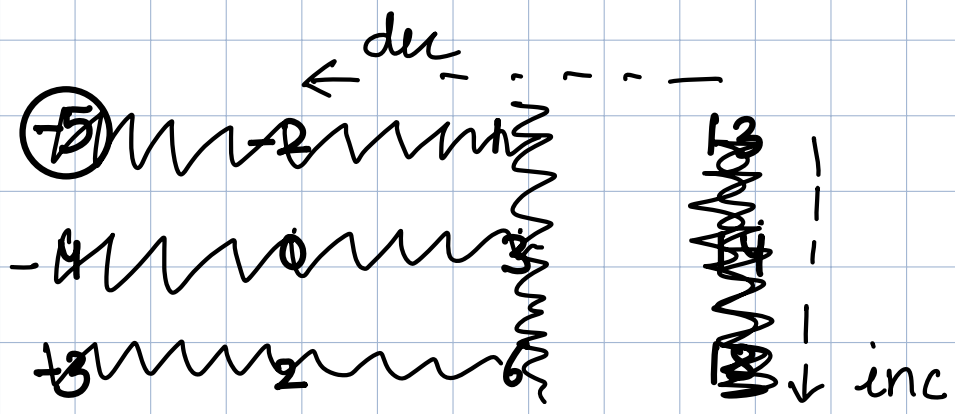
TC: $O(N+M)$

SC = $O(1)$

```
i=0 , j= M-1
while (i<N && j>=0) {
    if (A[i][j] == K) {
        return true;
    }
    else if (A[i][j] < K ) {
        i++
    } else if ( A[i][j] > K) {
        j--
    }

}
return false
```





$K = 5$

← out of matrix

return false

$i{+}{+}$ $\qquad$ $j{-}{-}$

$i < N$ $\quad$ && $\quad$ $j >= 0$

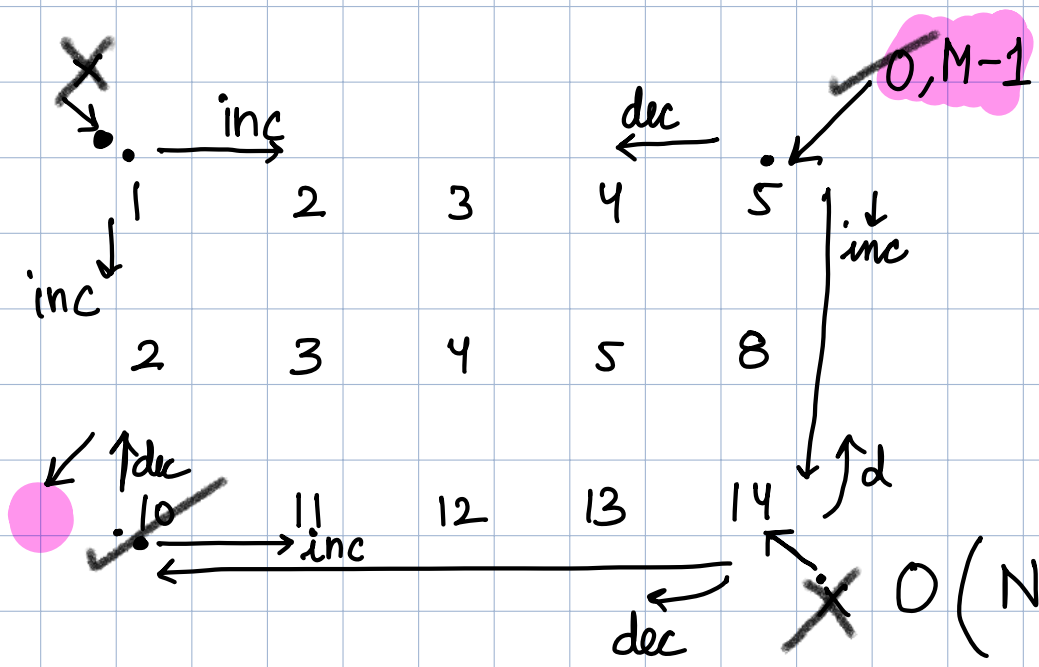
Till it is within bounds

$i{+}{+}$ $\qquad$ $i < N$

$j{-}{-}$ $\qquad$ $j >= 0$

discard rows / columns $\Big]$ $N + M$

2-way search



$K = 9$

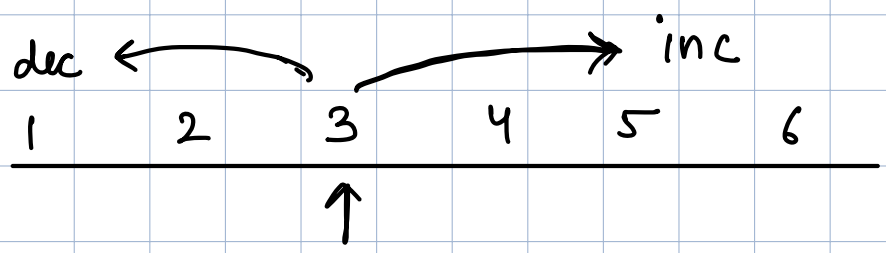
$O(N+M)$

Ques →

a) 0, 0

b) N-1, 0 ✓

c) N-1, M-1



Submatrix → continuous part of matrix

subarray → continuous part of array

$$mat = \begin{bmatrix} 1 & 2 & 3 & 4 \\ \boxed{5} & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \quad 4 \times 4$$

$$\begin{matrix} 2 & 3 \\ 6 & 7 \end{matrix} \qquad \begin{matrix} 5 & 6 \\ 9 & 10 \\ 13 & 14 \end{matrix}$$

5

1x1	1x2	1* 3	1* 4
2x1	2x2	2x 3	2x4
3x1	3x2	3x3	3x4
4x1	4x2	4x3	4x4

4x4

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix}$$

TL

BR

TL $A[1][0]$ to BR $A[3][1]$

$$A = \begin{bmatrix} 4 & 9 & 6 \\ 5 & -1 & 2 \end{bmatrix}_{2*3}$$ matrix

All possible sub matrices

1 * 1

4^{4} 9^{9} 6^{6} 5^{5} -1^{-1} 2^{2}

1 * 2

$\begin{bmatrix} 4 & 9 \end{bmatrix}^{13}$ $\begin{bmatrix} 9 & 6 \end{bmatrix}^{15}$ $\begin{bmatrix} 5 & -1 \end{bmatrix}^{4}$ $-1 \quad 2^{1}$

1 * 3

$\begin{bmatrix} 4 & 9 & 6 \end{bmatrix}^{19}$ $5 \quad -1 \quad 2^{6}$

$4 + 9 + 6 + 5 + -1 + 2$ 24 25

2 * 1

$\begin{matrix} 4 \\ 5 \end{matrix}^{9}$ $\begin{matrix} 9 \\ -1 \end{matrix}^{8}$ $\begin{matrix} 6 \\ 2 \end{matrix}^{8}$

$2 * 2$

$$\begin{matrix} 4 & 9 \\ 5 & -1 \end{matrix} \quad 17 \qquad \begin{matrix} 9 & 6 \\ -1 & 2 \end{matrix} \quad 16$$

$2 * 3$

$$\begin{matrix} 4 & 9 & 6 \\ 5 & -1 & 2 \end{matrix} \quad 25$$

$$\begin{aligned} &+13 \quad 38 \\ &+15 \quad 53 \\ &+4 \quad 57 \\ &+1 \quad 58 \\ &+19 \quad 77 \\ &+6 \quad 83 \\ &+9 \quad 92 \\ &+8 \quad 100 \\ &+8 \quad 108 \\ &+17 \quad 125 \quad +16 \quad 141 \\ &+25 \quad 166 \end{aligned}$$

Sum of submatrices sum $\quad 166$

Contribution technique

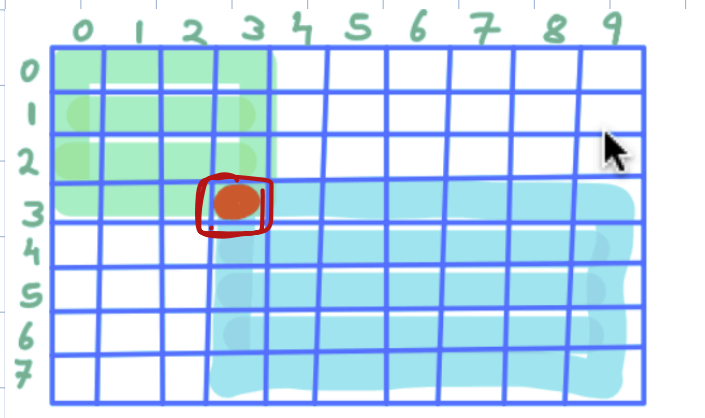
$$\begin{aligned} &4 * 6 = 24 \\ +\ &9 * 8 = 72 \\ +\ &6 * 6 = 36 \\ +\ &5 * 6 = 30 \\ +\ &-1 * 8 = -8 \end{aligned}$$

$+ \; 2 * 6 = 12$

166

4 (6)	9 (8)	6 (6)
5 (6)	-1 (8)	2 (6)

$\Rightarrow$ A[i][j] — how many submatrices A[i][j] is a part of



$$\boxed{\text{ans} = \sum_{\text{for all elements}} A[i][j] * \underset{\text{count}}{X}}$$

$A[i][j]$ — how many submatrices $A[i][j]$ is a part of

TL $i = 0, 1, 2$
$j = 0, 1, 2$

0,0 1,0 2,0
0,1 1,1 2,1 = 9
0,2 1,2 2,2

TL
row = 0 to i
col = 0 to j

	0	1	2	3	4
0	✓	✓	✓		
1	✓	✓	✓		
2	✓	✓	✓ (2,2)	✓	✓
3			✓	✓	✓
4			✓	✓	✓

9 → 2, 2

5 * 5

9 9

9 * 9 = 81

TL = 9 possiblities → 2, 2

TL i =

0 — i 0 - 2

0 — j 0 - 2

TL

3 * 3

0 1 2 0 1 2 = 9

$0 - i = (i+1)$ ← TL

$0 - j = (j+1)$

BR

BR row i to $N-1$

col j to $M-1$

$N - 1 - i + 1 = N - i$

*

$M - 1 - j + 1 = M - j$

$N - i$

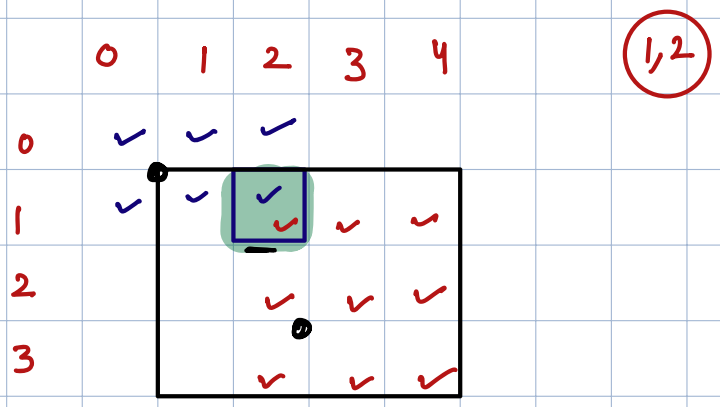
*

$M - j$

N=5
M=5
i=2 j=2

⇒ $5 - 2 \; * \; 5 - 2 = 3 * 3 = 9$

$$\underbrace{\boxed{(i+1) * (j+1)}}_{TL} * \underbrace{\boxed{(N-i) * (M-j)}}_{BR}$$



TL * BR

6 * 9 = 54

TL.

BR

$N = 4, \; M = 5.$

$i = 1; \; j = 2.$

TL

$i+1 \quad 2$

$* \quad * \; = 6$

$j+1 \quad 3$

$= 54$

BR

$N-i \quad 3$

$* \quad * \; = 9$

$M-j \quad 3$

```
total = 0
for( i=0;  i<N;      i++) {
    for(j=0     j<M     j++) {
        TL = (i+1) * (j+1)
        BR = (N-i) * (M-j)
        count=     TL * BR
        total = total + A[i][j] * count
    }
}
return total
```

$TC = O(N * M)$

$SC = O(1)$

$O(N^3 * M^3)$

Given an unsorted array of integers, Find first missing Natural Number.

Examples

1

arr[5] = {3, -2, 1, 2, 7} Ans: 4

ar[7] = {-9, 2, 6, 4, -8, 1, 3} Ans: 5

5?

{-2, 4, -1, -6, 3, 7, 8, 4, -3} Ans: 1

1?

{1, 0, -5, -6, 4, 2} Ans: 3

{1, 2, 5, 6, 4, 3} Ans: 7

7?

1

{-4, 8, 3, -1, 0} Ans: 1

{4, 2, 1, 3} 5? Ans: 5

Smallest missing natural no = 1

n biggest = n+1

1 to n+1

all are present

[1 4 2 3]

n = 4 . ans = 5

n+1

[1 5 6 7 3 2 4]

n = 7 ....> n+1 : 8

1 ✓

2 ✓

3 ✓

4 ?

1 2 13 5 3 ans = 4

first missing natural no

1 to n+1

1 ✓

2 ✓

3 ✓

n+1

[1 2 2 2 3]

1 ✓

2 ✓

3 ✓

4 ans

App 1 —

arr.length

```
for ( i = 1 ;   i <= (n+1) ;   i++) {
    flag = 0
    for ( j = 0   j < arr. length   j++){
        if (arr[j] = = i){ flag = 1
                           break; }
    }
    if (flag = = 0) { return i }
}
```

$TC = O(N^2)$

$SC = O(1)$

$[-4, -3, -18, -5]$ ans = 1

i = 1

$\{5 \quad 3 \quad 1 \quad -1 \quad -2 \quad -4 \quad 7 \quad 2\}$

i = 1 f = 0 .... f = 1

i = 2 f = 0 .... f = 1

i = 3 f = 0 .... f = 1

i = 4 f = 0 $\boxed{\text{return } 4}$

```
for ( i = 1 ;   i <= (n+1) ;   i++) {
    flag = 0
    for ( j = 0   j < arr. length   j++){
        if (arr[j] = = i){ flag = 1
                           break; }
    }
    if (flag = = 0) { return i }
}
```

2) Sorting → TC = $O(N \log N)$

3) Using another array →

	0	1	2	3	4	5
arr =	-5	4 (no)	3	1	2	6
	I	no-1				

⇒ 5 ans

1 to N+1 , N = 6 → 7

boolean array ⇒

0	1	2	3	4	5	6
~~F~~ T	~~F~~ T	~~F~~ T	T	F	~~F~~ T	F

1	2	3	4	5
✓	✓	✓	✓	Not present

1 to N+1

<1 ignore

>N+1 ignore

⇒

0	1	2	3	4
5	1	-5	2	13
=	=	I	-	I

N = 5

N+1 = 6

0	1	2	3	4	5
~~F~~ T	~~F~~ T	F	F	~~F~~ T	F

→ 3 ans

```
boolean[] flag = new boolean[N+1];

for ( i = 0          i < arr.length;   i++) {
        if ( arr[i] < 1  || arr[i] > N )
                continue

        else
                flag[arr[i] - 1] = true
}

for ( j = 0          j < flag.length   j++) {
        if ( flag[j] == false)
                return j + 1
}
```

TC = $O(N)$

$N^2 \longrightarrow N \log N \longrightarrow N$

$=$

1	4.	3	2

4

5 ans

$N+1 = 5$

~~F~~T	~~F~~T	~~F~~T	~~F~~T	F
0	1	2	3	4

index = 4

ans = 5

Doubts $\longrightarrow$