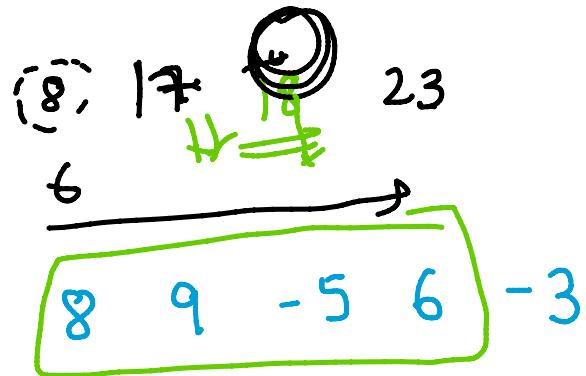
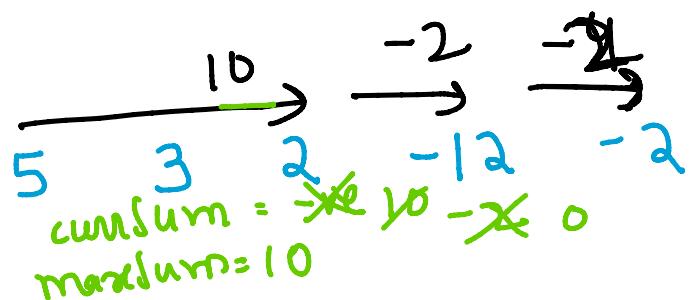
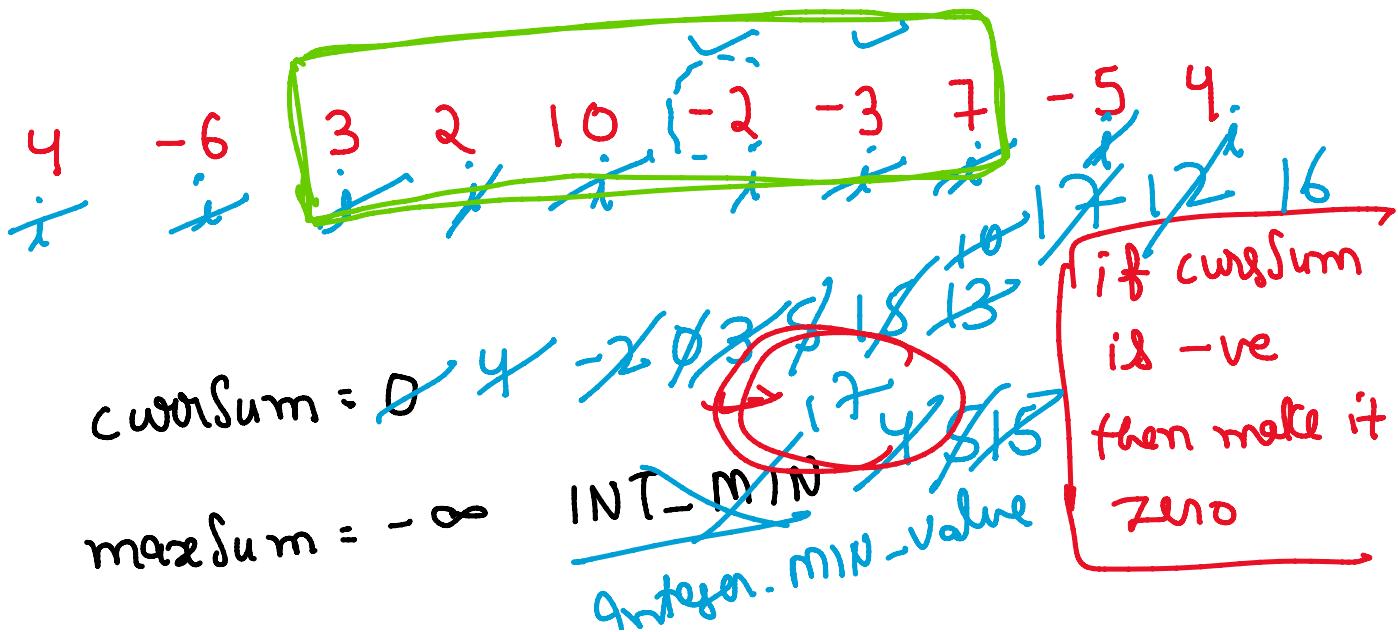
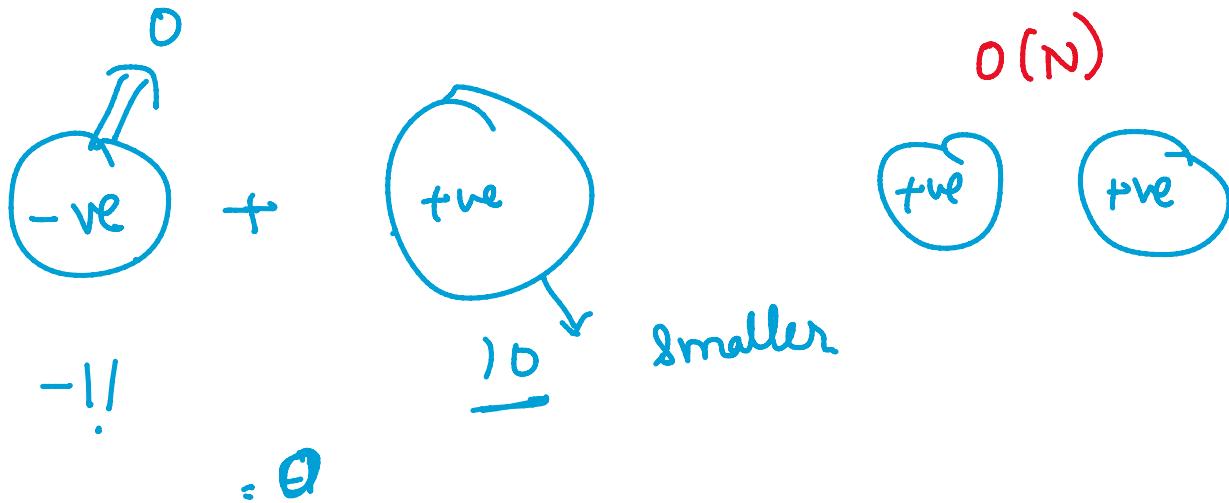


→ Maximum subarray sum



① Two loops. Generate all subarrays } and find the maximum sum. } (N^2)

②



maxSum -

integer.MIN - v~

-5 -3 -10

(-3)

-2
↓
0

3
3

$$\text{res} = 1$$
$$\text{res} = 3$$

-2 3 5

$$-2 + 3 + 5 = 6$$

(8)

ans = -2

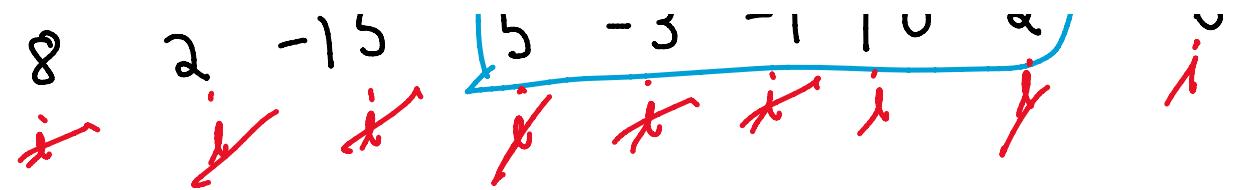
-5 -3 -10 -2 -4
x i x x x

currSum = 0 -5 0 -3 0 -10 0 -2 0 -4

maxSum =

(0) -5 -3 -2

8 2 -15 [5 -3 -1 10 2] -6
13



~~currSum = 1846 - 8082 + 11137~~

~~maxSum = -∞~~ ~~846~~ ~~113~~

$\rightarrow O(N)$

$\rightarrow O(1)$

Maximum product of 3 numbers

04 September 2022 13:33

$\rightarrow 1 \circled{5} 0 \circled{3} \circled{4}$ strictly
3

$\Rightarrow 60$

$\rightarrow -1 \circled{-5} \circled{0} \circled{-3} \circled{-4}$

-20
0

$$n-3 \quad n-2(10, -8, -6) \quad 480 \\ j \quad (\times) \quad 240$$

$\rightarrow -3 \circled{10} \circled{2} \circled{-8} \circled{4} \circled{-6}$

$-8 \quad -6 \quad -3 \quad 2 \quad 4 \quad 10$

sort

last 3 values ||
 $(2 \times 4 \times 10)$
 $\underline{\underline{240}}$

first two and last +
 $(-8 \times -6 \times 10)$

480

i 0 to $n-3$
j $i+1$ to $n-2$
k $j+1$ to $n-1$

\rightarrow ① 3 loops. $\underline{\underline{N^3}}$

② sort

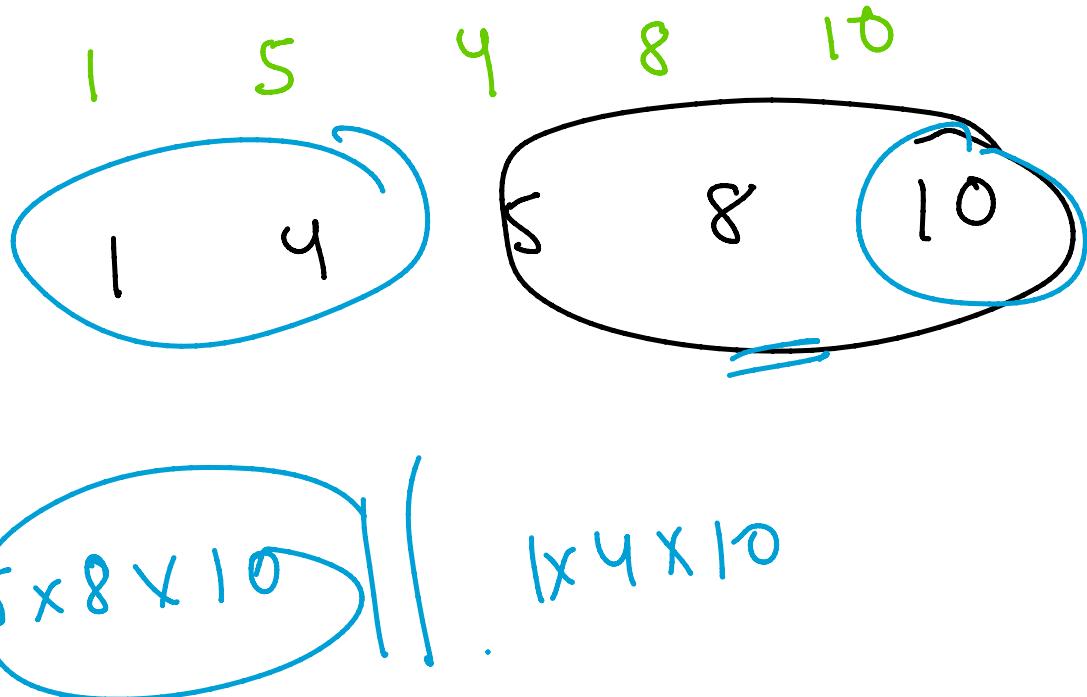
$-8 \quad -6 \quad -5 \quad -4 \quad -2 \quad -1$

UX

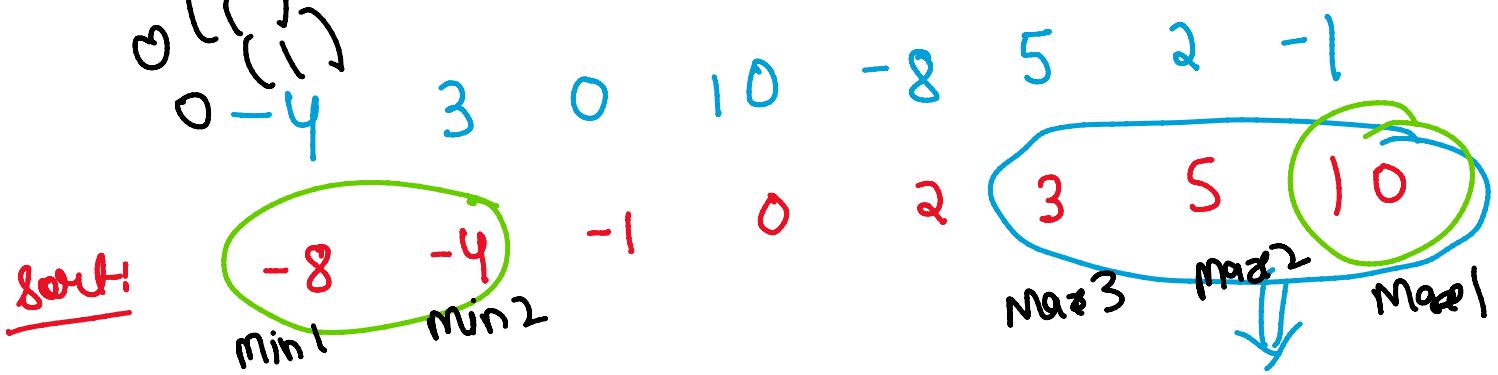
-48
-8

48

====



- 480
- ① 3 loops $\underline{(N^3)}$
 - ② sorting $\underline{(N \log N)}$
 - ③ 5 variables $\underline{(N)}$
- 5 variables
single traversal
 \rightarrow 2 min
 \rightarrow 3 max



320

$$\checkmark \min_1 = \text{INT_MAX} \quad \cancel{-5 - 7}$$

$$\checkmark \min_2 = \text{INT_MAX} \quad \cancel{+ -5}$$

-5, 1

$$\checkmark \max_1 = \text{INT_MIN} \quad \cancel{10 - 8} \quad \cancel{X_2}$$

$$\checkmark \max_2 = \text{INT_MIN} \quad \cancel{4 - 2} \quad \cancel{X_1}$$

$$\checkmark \max_3 = \text{INT_MIN} \quad \cancel{-5 \neq X_2}$$

greater than 2

less than

$$-5 \quad 1 \quad 2 \quad -7 \quad 0 \quad 10 \quad -3 \quad 4$$

$$i \quad i \quad i \quad i \quad i \quad i \quad i \quad i$$

$(10 + 4 + 2) \parallel (-7 \times -5 \times 10)$

356

2nd longest

$$80$$

$$5 \quad 3$$

$$i \quad i \quad i \quad i \quad i \quad i \quad i \quad i$$

$$-2 \quad -10 \quad 8 \quad 7 \quad 1$$

$$\max_1 = \text{INT_MIN}$$

$$\max_2 = \text{INT_MIN}$$

5 8
3 7

20

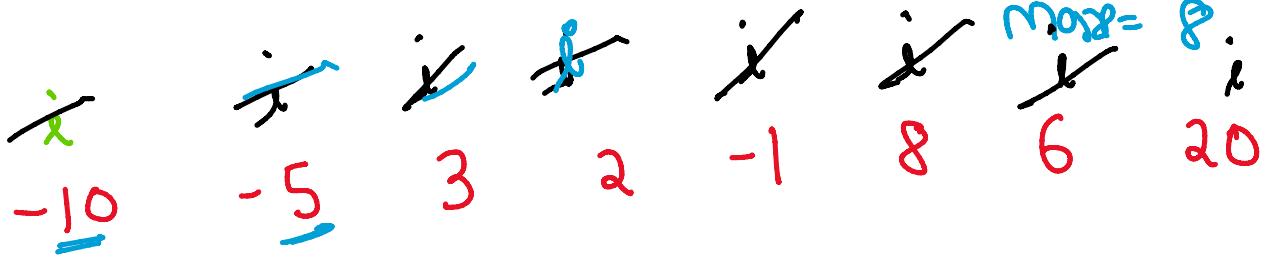
5, 10 / 20

$$\max_1 = \cancel{10} \quad 20$$

$$\max_2 = \cancel{5} \quad 10$$

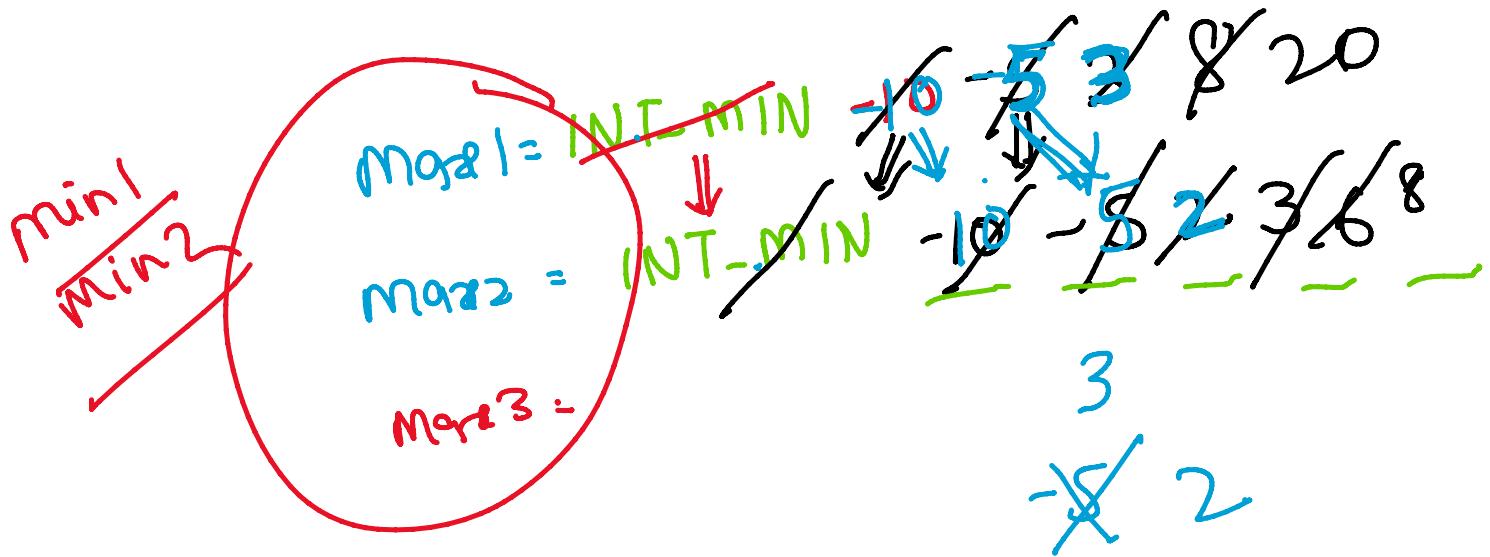
$$\max_1 = 20$$

$$\text{max1} = \cancel{5} \quad \cancel{10}$$



$$\text{max1} = 20$$

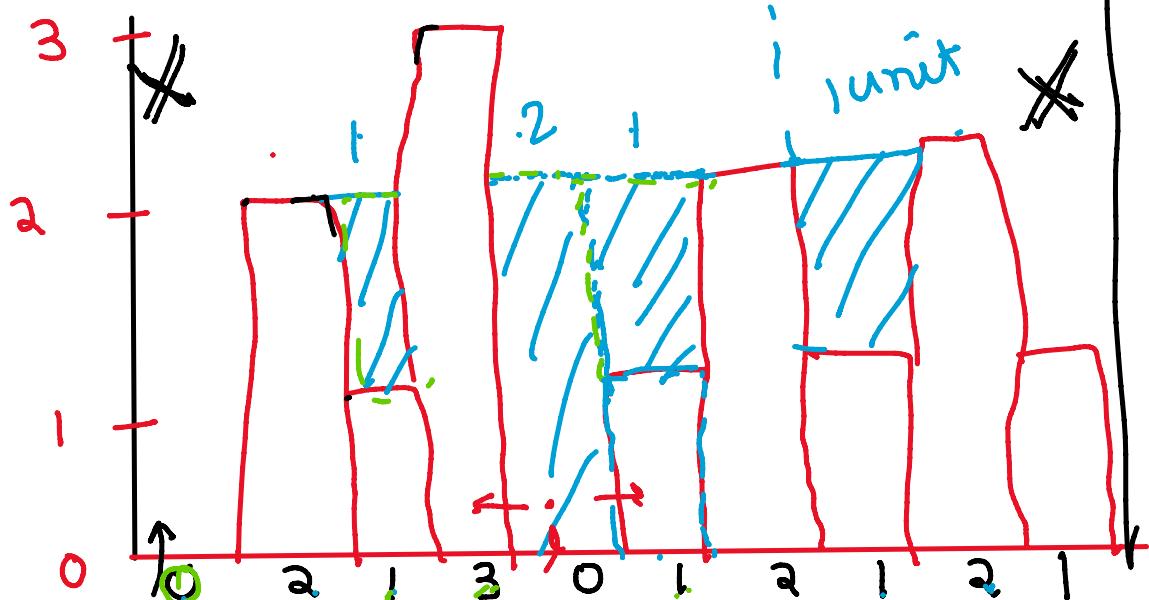
$$\text{max2} = 8$$



Trapping rain water

04 September 2022 14:20

0, 2, 1, 3, 0, 1, 2, 1, 2, 1, 5

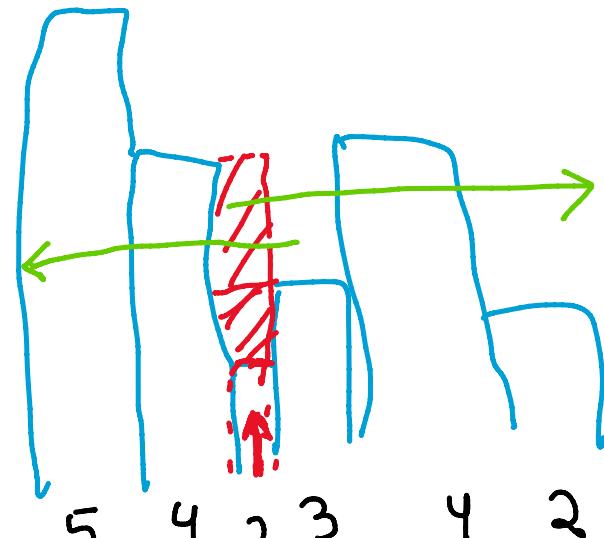


leftMax: 0 2 2 3 3 3
RightMax: 3 3 3 2 2 2 2 2 1
water: 0 0 1 0 2 1 0 1 0 0

$\sum 5$

$\frac{2}{3}$

- ① ↘
- ② ↗
- ③ H.W.



$$\text{Water} = \min(\text{left Max}, \text{right Max}) - \text{current Height}$$

$$\min(5, 4) - 2 = 2$$

$$\min(5, 9) - 2$$
$$(4) - 2 = \boxed{2}$$

$$\text{leftmax} - O(N)$$
$$\text{rightmax} - O(N)$$
$$N + N = \cancel{O(N)}$$

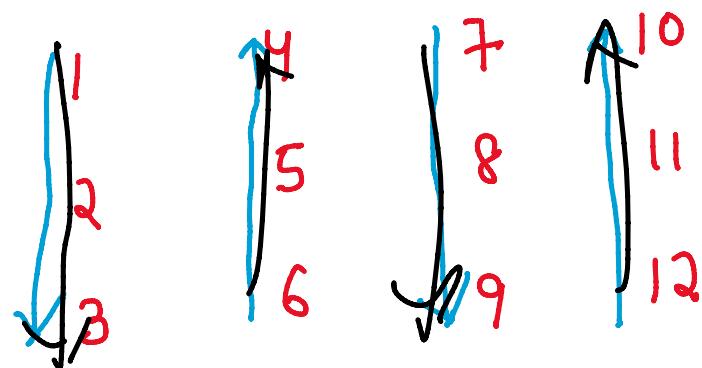

2D arrays

int matrix[3][3];

row = 3

col = 4

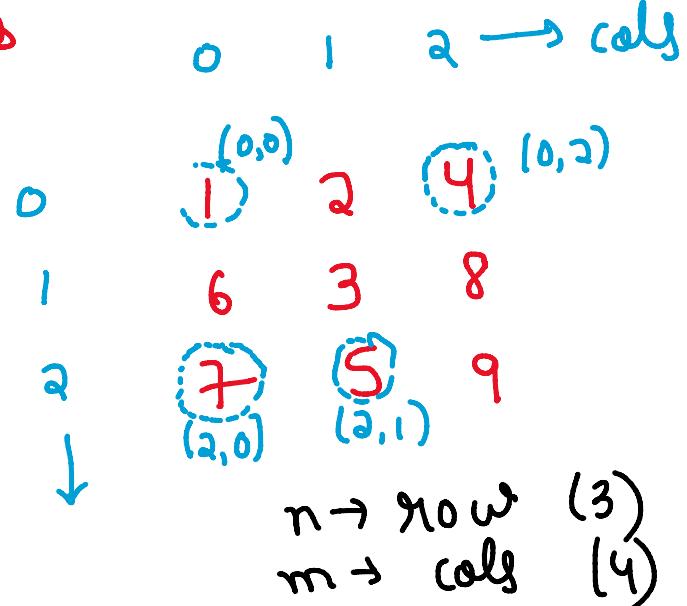
0 1 2 3



1, 2, 3, 6, 5, 4, 7, 8, 9, 12, 11, 10

```

for( col = 0 to col < m)
{
    if( col % 2 == 0)
        for( row = 0 to row < n)
            {
                print();
            }
    else
        for( row = n - 1 to row >= 0)
            {
                print();
            }
}
  
```



Col is Even \rightarrow Top down
Col is odd \rightarrow Bottom up

4 cols.
0, 1, 2, 3
even ($N \times M$)
(N^2)
//odd

→ rotate matrix 90° clockwise direction.

input

output

square matrix

~~inplace~~ some matrix
without using any extra space.

$$\textcircled{1} \quad \underline{N^2}, \quad \underline{N^2}$$
$$\textcircled{2} \quad N^2, \quad 1$$

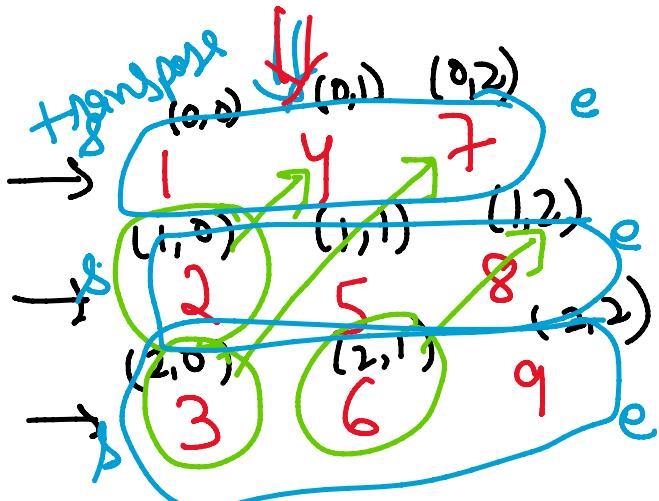
jd

A diagram showing a swap operation in an array. A green oval encloses elements 1, 2, and 3. Above the oval, the word "swaps" is written in green. A blue bracket labeled "input" in blue spans across the oval, indicating the range of elements being swapped. Below the oval, elements 4, 5, and 6 are shown in blue.

7 9
8 5

for (j=0 to 2)
 swap ()

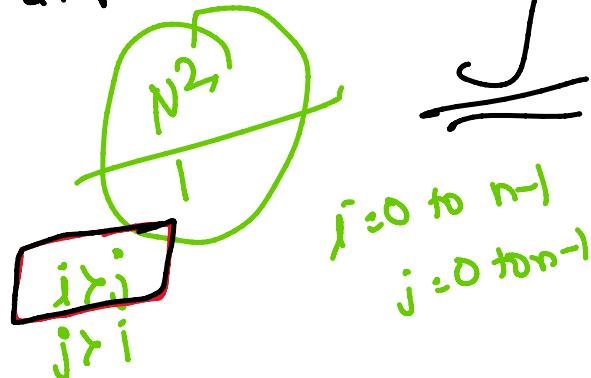
4 5 6 →
7 8 9



8 5 2
9 6 3

1. transpose

2. reverse each row



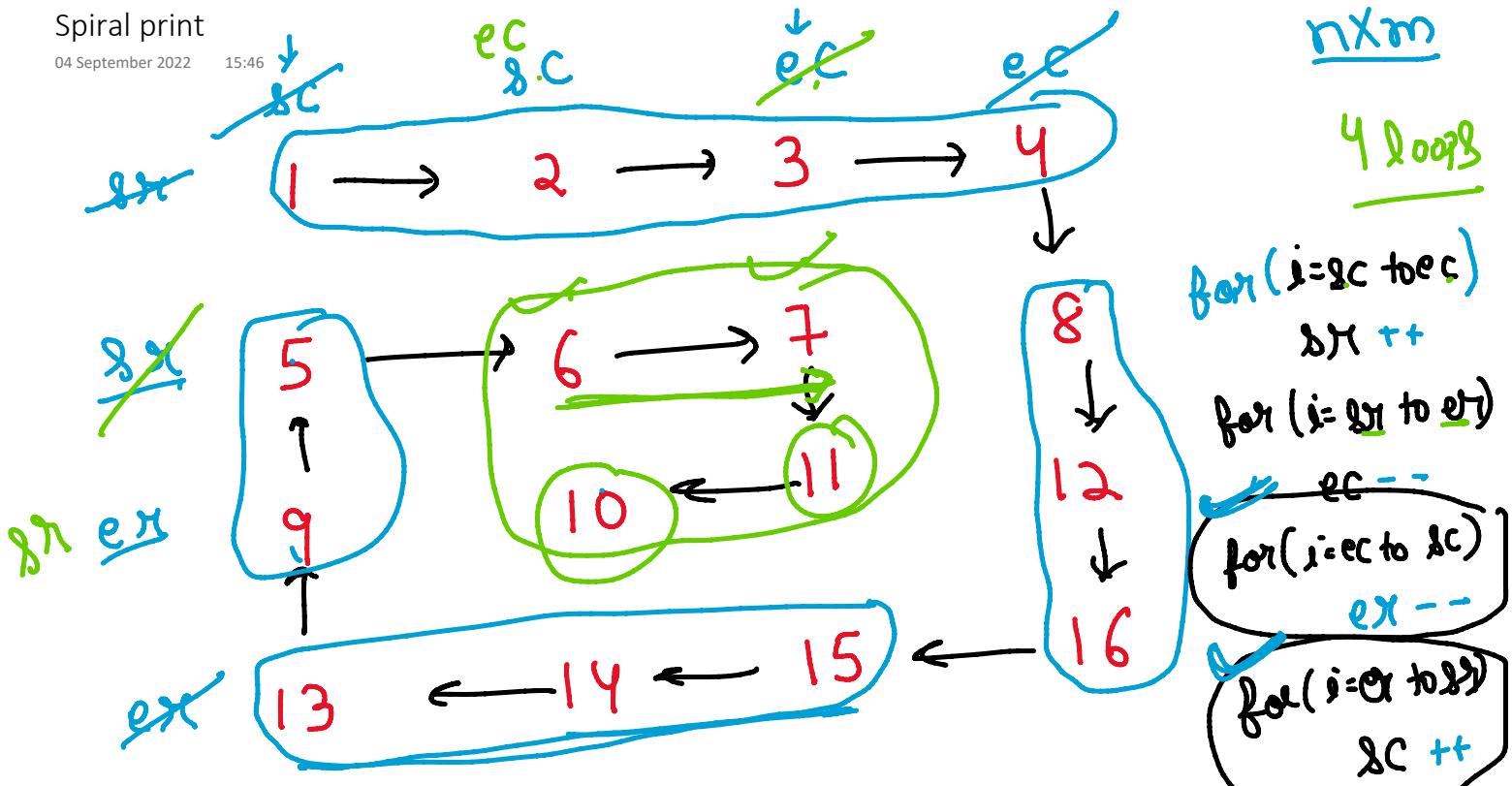
6 5 4
8 2 3
e 8 7

while ($8 < e$)
 $\delta \leftarrow \delta + 8$
 $e \leftarrow e - 1$

Spiral print

04 September 2022

15:46



1, 2, 3, 4, 8, 12, 16, 15, 14, 13, 9, 5, 6, 7, 11, 10

while ($sc < ex \text{ & } sc \leq ec$)

{
 }
 }
 }

ex sc

4 →

5x3

row 1 for ✓

col 1 for
row last for
 $sc \leq ex$

col last for
 $sc \leq sc$