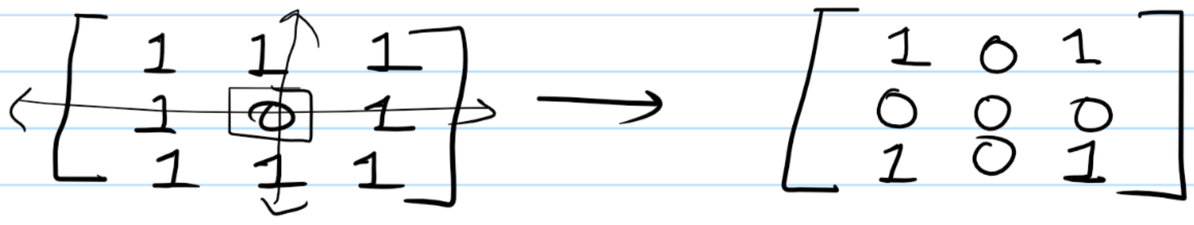
2D arrays class 3

Q1 Set matrix zeroes



For any element which is "0" make corr. row & column 0.

Approach 1:

Algo: 1) Make a pass over original array & look for 0 elements

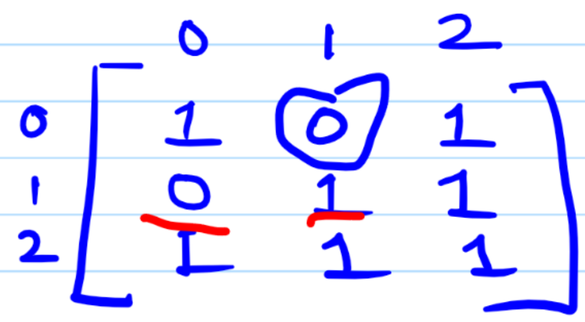
2) If entry at arr[i][j] = 0, then record i in row hashset & j in column hashset

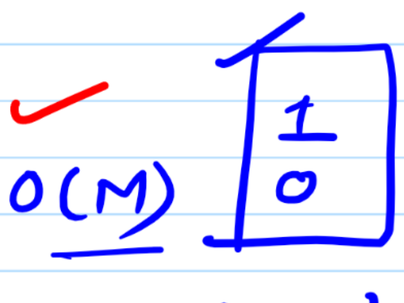
3) Iterate over the original matrix

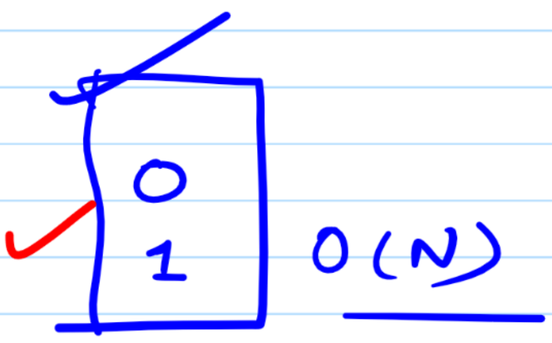
if arr[r][c]

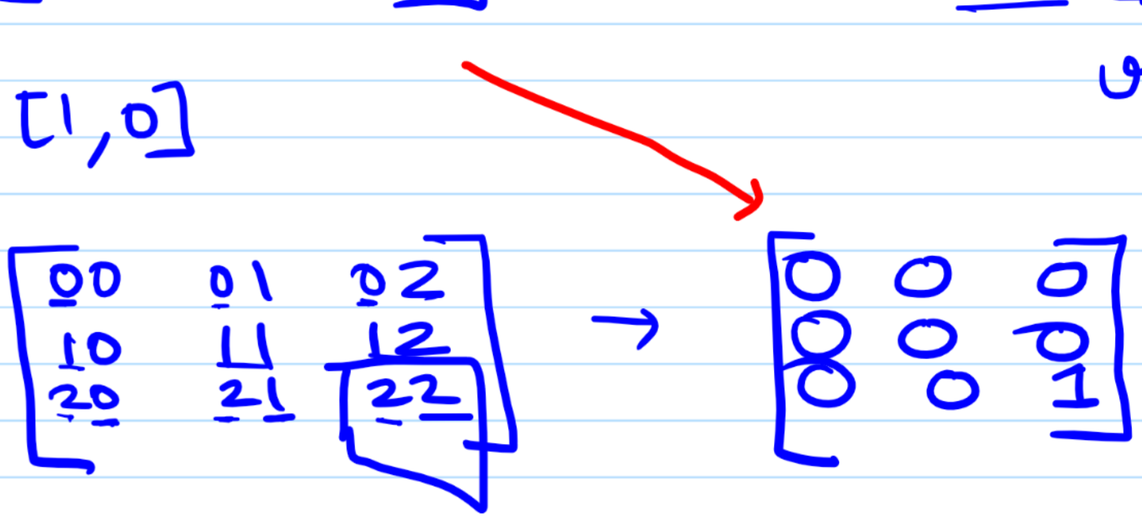
if r is in rowset or c is in columnset

set arr[r][c] = 0









TC: O(MxN)

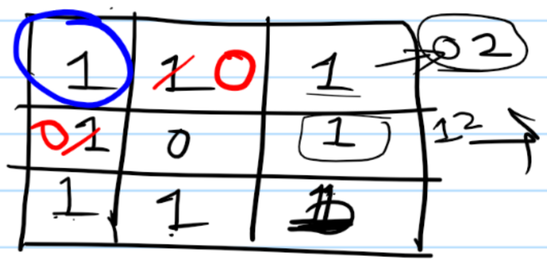
SC: O(M+N)

2 x (M+N) = O(M+N)

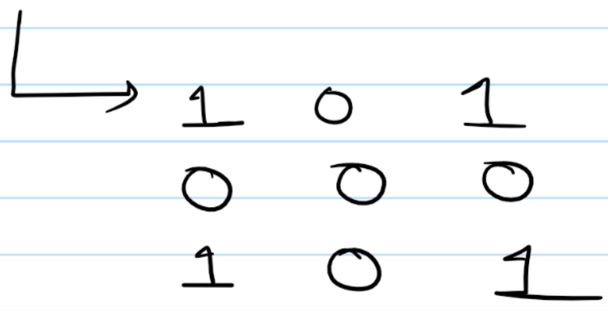
Approach 2:

Hint: use first element of every row & column as a FLAG.

10

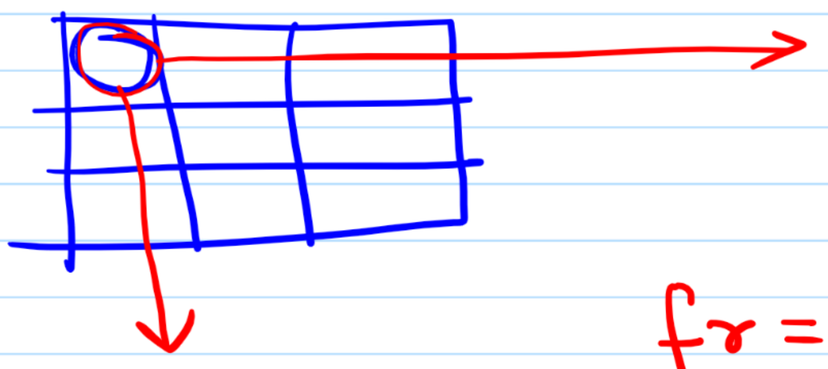


1 0 1
0 0 0
1 0 1



1 0 1
0 0 0
1 0 1

Catch



0 1 0 1
1 1 1
1 1 1

→ 0
0

$fr = 0 \quad fc = 0$

Algorithm :

1) We iterate over the matrix & mark first cell of row i & first cell of column j; if cell[i][j] = 0

Catch

2) for The first cell of row & column cell[0][0], we'll use 2 variables fr and fc to know if first row | first column has to be set 0 or not -

3) Now we iterate over matrix again, matrix[r][c]
if matrix[r][0] == 0 or matrix[0][c] == 0
we set matrix[r][c] = 0

4) Based on value of fr & fc, we'll set first row / first column as 0.

TC: $O(m \times n)$

SC: $O(1)$

Q2: Reshape the matrix

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$ r c 1, 4

$m * n =$ r x c

$2 \times 2 = 1 \times 4$

[1 2 3 4] ✓

[1 2 4 3] ✗

1st Approach: Queue ✓

First in First out

Stack

LIFO (Last-in First out).

r x c

1	2	3	4

Queue | 1 | 2 | 3 | 4 |

TC: $O(M*N)$

SC: $O(M*N)$

2nd approach:

1 2 3

4 5 6

r=0 c=0

3*2 matrix

	c=0	c=1	c=2
r=0	1	2	
r=1	3	4	
r=2	5	6	

Algorithm:

Start pushing elements in the resultant matrix directly. We start from a particular row & keep on incrementing the column number till we reach end of required columns. At that moment, we increment row & reset column no. to 0.

TC: $O(m*n)$

SC: $O(1)$

Approach:

3x4 matrix

c ← total no. of columns

0	1	2	3
00	01	02	03
4	5	6	7
10	11	12	13
8	9	10	11
20	21	22	23

1*4+2

count / c = row

count % c = column

i*c+j

✓

$\frac{6}{4} = 1$ $\quad 6\%4 = 2$

[12]

$\frac{11}{4} = 2$ $\quad 11\%4 = 3$

[23]

i, j $\qquad$ i*c+j

93: Flipping an image

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \xrightarrow{\text{Flip}} \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \xrightarrow[(1\to0,\ 0\to1)]{\text{Reverse}} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

1 1 0 → 1 0 0

i^{th} bit = $n-i-1$ bit (flip it).

≠ (No operation)

0 1 2

1 0 1 → 101 → 0 1 0

1 0 1 → 0 1 0

i^{th} $\qquad$ $n-i-1$

→ 1 1 0 1 → 1 0 1 1 → 0 1 0 0

n=4

1 1 0 1 → 0 1 0 0

0 1 2 3

i, n-i-1

(0, 3)

0 to n/2

2 $\qquad$ 4-2-1

[2 1]

1 2

(2 1)

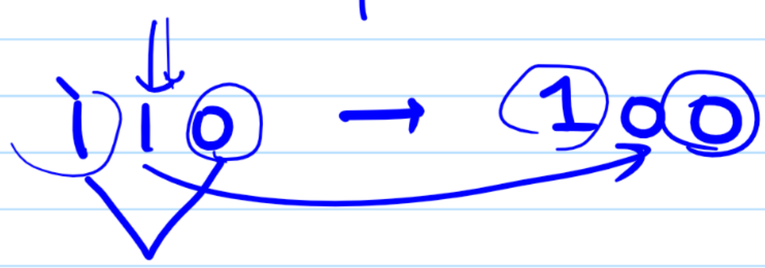
[1 2]

(1, 2)

TC: O(MxN)

SC: O(1)

[1 1 0] | 0 11 | 100



101

$x \hat{} 1 = \bar{x}$

$0 \hat{} 1 = 1$

$1 \hat{} 1 = 0$

⊕ XOR operation