

Modify The Matrix

→ row = 3

→ col = 3

arr

	0	1	2	3
0	1	0	0	1
1	0	0	1	0
2	0	0	0	0

val is 1 at (0,0), (1,2), (0,3)

ans

	0	1	2	3
0	1	1	1	1
1	1	1	1	1
2	1	0	1	1

$$\text{operations} = N * M * (N + M)$$

$$T.C = O(M * N * (N + M))$$

$$= O(MN^2 + M^2N)$$

$$S.C = O(N * M)$$

ans

	↓ 0	1	↓ 2	↓ 3
col	1	0	1	1

	row
→ 0	1
→ 1	1
→ 2	0

	0	1	2	3
arr 0	1	0	0	1
1	0	0	1	0
2	0	0	0	0

row = 3
col = 4

row col
1 → (0, 0)
(0, 3)
(1, 2)

$$S.C = O(N + M)$$

$$N = 2000$$

$$M = 2000$$

$$N * M = 40,0000$$

$$N + M = 4000$$

```
public static void modifyMatrix(int[][] arr, int m, int n) {
```

```
    int[] row = new int[m];
```

```
    int[] col = new int[n];
```

```
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            if (arr[i][j] == 1) {
                row[i] = 1;
                col[j] = 1;
            }
        }
    }
```

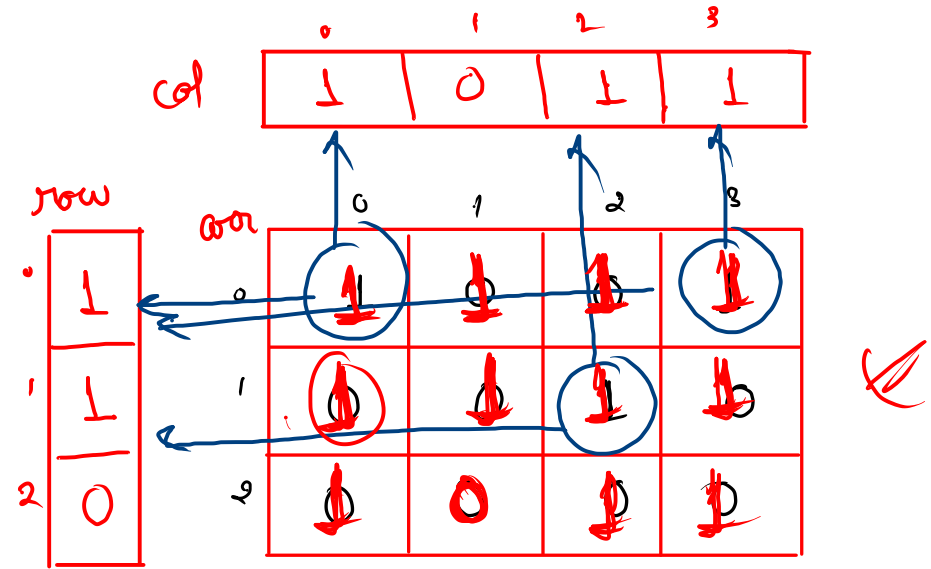
N*M

$T.C = O(N*M)$
 $S.C = O(N+M)$

```
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            if (row[i] == 1 || col[j] == 1) {
                arr[i][j] = 1;
            } else {
                arr[i][j] = 0;
            }
        }
    }
```

N*M

```
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            System.out.print(arr[i][j] + " ");
        }
        System.out.println();
    }
```



$m=3$
 $n=4$

i	j	True	True
0	0	True	1,2 True
0	1	True	1,3 True
0	2	True	2,0 True
0	3	True	2,1 False
1	0	True	2,2 True
1	1	True	2,3 True

Number of square matrices with all 1s

size=1 size=2 size=3

row = 3

col = 4

$$sq = 10 + 4 + 1$$

$$= 15$$

	0	1	2	3
0	0	1	1	1
1	1	1	1	1
2	6	1	1	1

∅) meaning of each index?

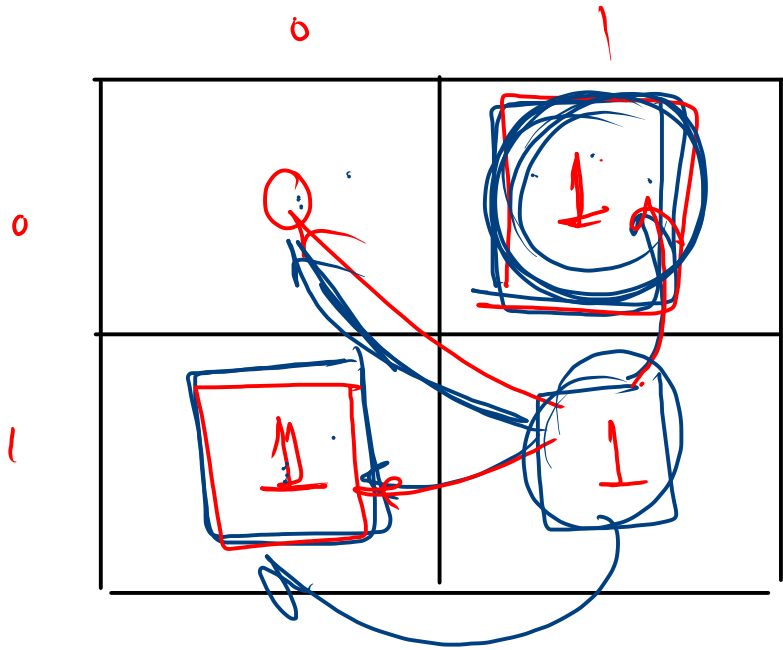
meaning of each idx \rightarrow containing no. of square matrices we
can make ending here

	0	1	2	3
0	0	1	1	1
1	1	1	2	2
2	0	1	2	3

no. of $\Rightarrow 0 + 1 + 1 + 1 +$
sq $1 + 1 + 2 + 2 +$
matrices $1 + 2 + 3$

$\Rightarrow \underline{\underline{15}}$

how to calculate ans for each idx



$$\text{ans}[i][j] = \min(\text{ans}[i-1][j], \text{ans}[i][j-1], \text{ans}[i-1][j-1]) + 1$$

check ans from
left idx, up_idx and
diagonal idx

```
// main logic
public static int countOfSquareMatrices(int[][] arr, int m, int n) {
    // each index store
    // no. of square matrices with all 1 , ending here
    for (int i = 1; i < m; i++) {
        for (int j = 1; j < n; j++) {
            → if ( arr[i][j] == 1 ) {
                arr[i][j] = Math.min(arr[i - 1][j - 1], Math.min( arr[i - 1][j], arr[i][j - 1] )) + 1;
            }
        }
    }

    int sum = 0;
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            sum += arr[i][j];
        }
    }
    return sum;
}
}
```

over

	0	1	2	3
0	0	1	1	1
1	1	1	2	2
2	0	1	2	3

$$T.C = O(n \times m)$$

$$S.C = O(1)$$