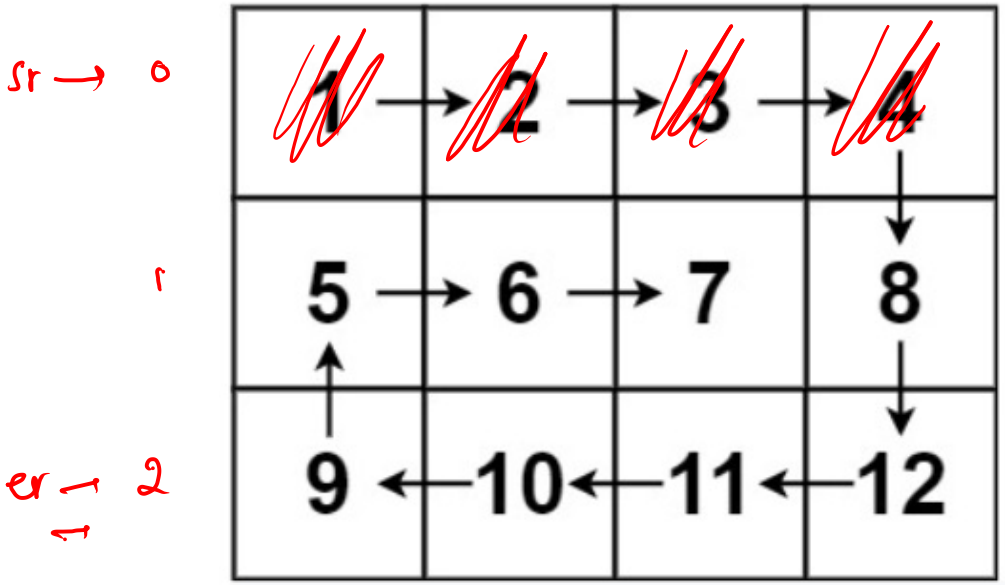
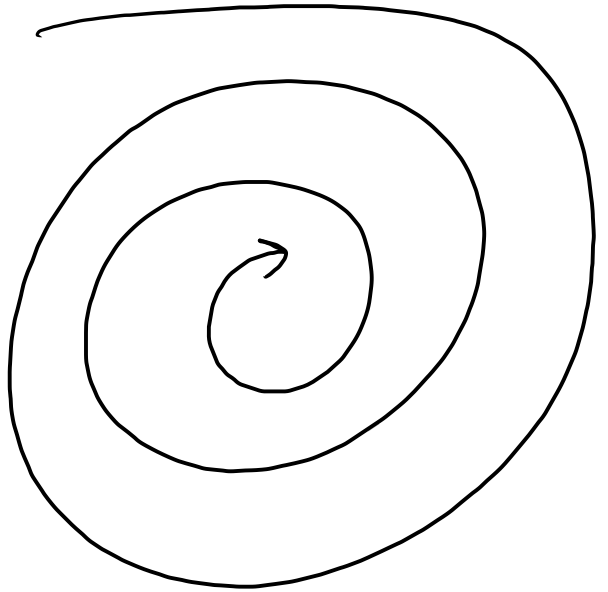
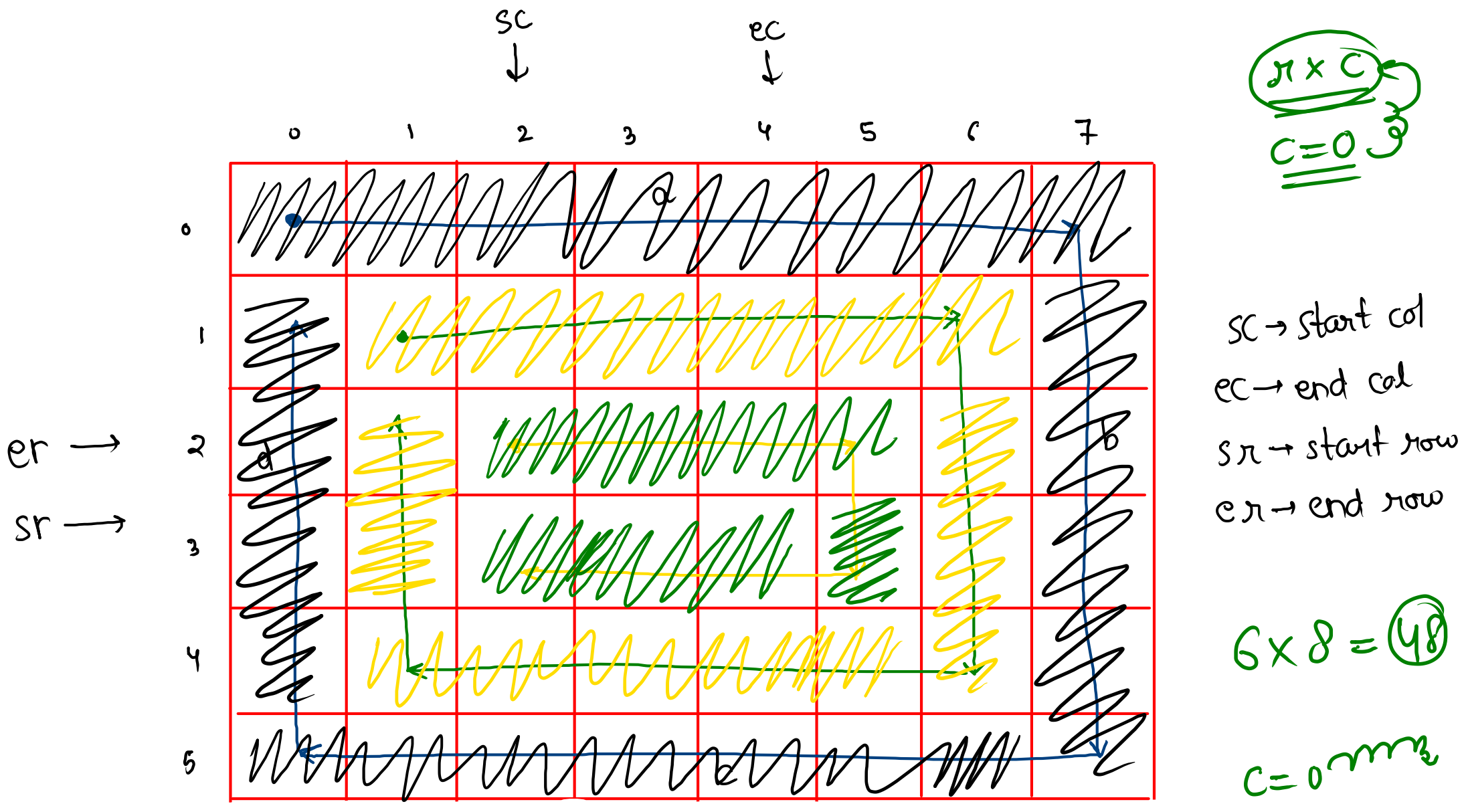
Spiral Matrix 44





```
Input: matrix = [[1,2,3,4],[5,6,7,8],[9,10,11,12]]
Output: [1,2,3,4,8,12,11,10,9,5,6,7]
```

Spiral 1



r × c

c = 0

sc → start col

ec → end col

sr → start row

er → end row

6 × 8 = 48

c = 0 ^m²

a→ traverse from sc to ec in sr

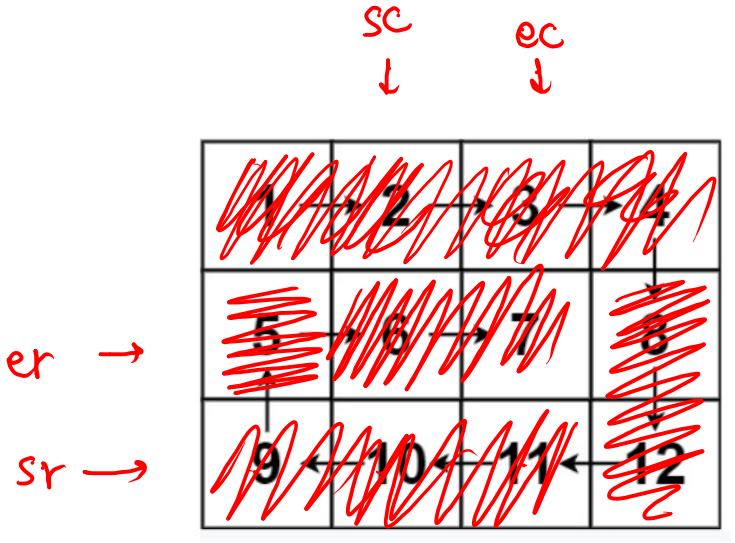
b→ traverse from sr to er in ec

c→ traverse from ec to sc in er

d→ traverse from er to sr in sc

```java
public static void sprial(int[][] arr, int row, int col) {
    int sr = 0;
    int sc = 0;
    int er = row - 1;
    int ec = col - 1;
    int total = row * col;
    int count = 0;
    while (count < total) {
        for (int j = sc; j <= ec && count < total; j++) {
            System.out.print( arr[sr][j] + "   " );
            count++;
        }
        sr++;

        for (int i = sr; i <= er && count < total; i++) {
            System.out.print( arr[i][ec] + "   " );
            count++;
        }
        ec--;

        for (int j = ec; j >= sc && count < total; j--) {
            System.out.print( arr[er][j] + "   " );
            count++;
        }
        er--;

        for (int i = er; i >= sr && count < total; i--) {
            System.out.print( arr[i][sc] + "   " );
            count++;
        }
        sc++;
    }
}
```

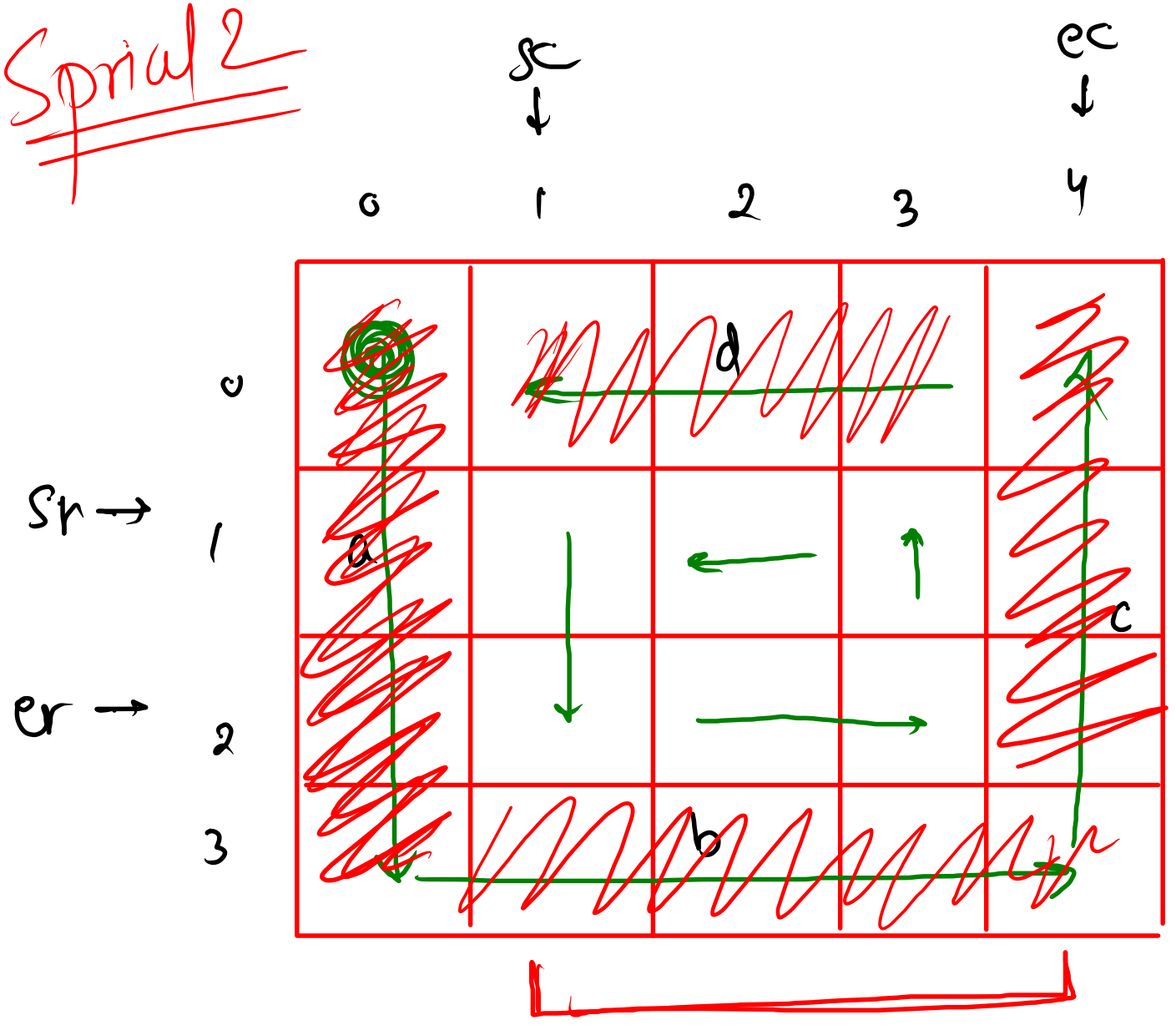
```java
public static void sprial(int[][] arr, int row, int col) {
    int sr = 0;
    int sc = 0;
    int er = row - 1;
    int ec = col - 1;
    int total = row * col;
    int count = 0;
    while (count < total) {
        for (int j = sc; j <= ec && count < total; j++) {
            System.out.print( arr[sr][j] + "   " );
            count++;
        }
        sr++;

        for (int i = sr; i <= er && count < total; i++) {
            System.out.print( arr[i][ec] + "   " );
            count++;
        }
        ec--;

        for (int j = ec; j >= sc && count < total; j--) {
            System.out.print( arr[er][j] + "   " );
            count++;
        }
        er--;

        for (int i = er; i >= sr && count < total; i--) {
            System.out.print( arr[i][sc] + "   " );
            count++;
        }
        sc++;
    }
}
```



total = 12

count = 0 4 6 8 10 12

Spiral 2



a → sr to er in sc, sc++

b → sc to ec in er, er--

c → er to sr in ec, ec--

d → ec to sc in sr, sr++

Compare Two Matrices

- same when of
  - equal size
  - equal values at all indices


```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int row1 = scn.nextInt();
    int col1 = scn.nextInt();
    int[][] arr1 = new int[row1][col1];
    for (int i = 0; i < row1; i++) {
        for (int j = 0; j < col1; j++) {
            arr1[i][j] = scn.nextInt();
        }
    }

    int row2 = scn.nextInt();
    int col2 = scn.nextInt();
    int[][] arr2 = new int[row2][col2];
    for (int i = 0; i < row2; i++) {
        for (int j = 0; j < col2; j++) {
            arr2[i][j] = scn.nextInt();
        }
    }

    compareMatrix(arr1, row1, col1, arr2, row2, col2);
}

public static void compareMatrix(int[][] arr1, int row1, int col1, int[][] arr2, int row2, int col2) {
    if (row1 == row2 && col1 == col2) {

        for (int i = 0; i < row1; i++) {
            for (int j = 0; j < col1; j++) {
                if ( arr1[i][j] != arr2[i][j] ) {
                    System.out.println("Not Same");
                    return;
                }
            }
        }
        System.out.println("Same");

    } else {
        System.out.println("Not Same");
        return;
    }
}
```

Add Two Matrices

$$A = \begin{bmatrix} 1 & 2 & 1 \\ 3 & 1 & 2 \\ 3 & 2 & 3 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 1 & 2 \\ 2 & 0 & 1 \\ 1 & 3 & 2 \end{bmatrix}, \quad ans = \begin{bmatrix} 1 & 3 & 3 \\ 5 & 1 & 3 \\ 4 & 5 & 5 \end{bmatrix}$$

we can only add 2 matrices
- ↳ when they are of same size
- ↳ add values of same index

Code

```java
public static void addMatrix(int[][] arr1, int row1, int col1, int[][] arr2, int row2, int col2) {
    int[][] ans = new int[row1][col1];
    if (row1 == row2 && col1 == col2) {

        for (int i = 0; i < row1; i++) {
            for (int j = 0; j < col1; j++) {
                ans[i][j] = arr1[i][j] + arr2[i][j];
            }
        }

        for (int i = 0; i < row1; i++) {
            for (int j = 0; j < col1; j++) {
                System.out.print(ans[i][j] + " ");
            }
            System.out.println();
        }

    } else {
        System.out.println("-1");
    }
}
```

Multiplication of Two Matrices

$$A = \begin{bmatrix} 1 & 2 & 1 \\ 3 & 1 & 2 \end{bmatrix}_{2\times 3} \quad (a \times b)$$

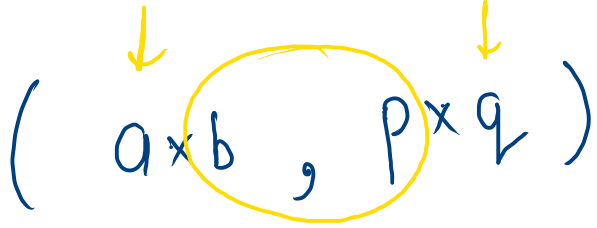
$$B = \begin{bmatrix} 0 & 1 \\ 2 & 0 \\ 1 & 3 \end{bmatrix}_{3\times 2} \quad (p \times q)$$

$$ans = \begin{bmatrix} 1\times0+2\times2+1\times1 & 1\times1+2\times0+1\times3 \\ 3\times0+1\times2+2\times1 & 3\times1+1\times0+2\times3 \end{bmatrix}$$

$$ans = \begin{bmatrix} 5 & 4 \\ 4 & 9 \end{bmatrix}$$

When to multiply

- when $b == p$



- our answer matrix will be of $a \times q$

$$A = \begin{bmatrix} 3 & 1 \\ 2 & 0 \\ 1 & 0 \end{bmatrix}_{3\times 2}, \quad B = \begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix}_{2\times 2}$$

$$ans = \begin{bmatrix} (3\times1)+(1\times2) & (3\times0)+(1\times1) \\ (2\times1)+(0\times2) & (2\times0)+(0\times1) \\ (1\times1)+(0\times2) & (1\times0)+(0\times1) \end{bmatrix}$$

$3 \times \boxed{2 \quad , \quad 2} \times 2$

condition

$$A \times B \neq B \times A$$

$$ans = \begin{bmatrix} 5 & 1 \\ 2 & 0 \\ 1 & 0 \end{bmatrix}$$

Note :- for all rows of matrix A, we will multiply with all cols of matrix B