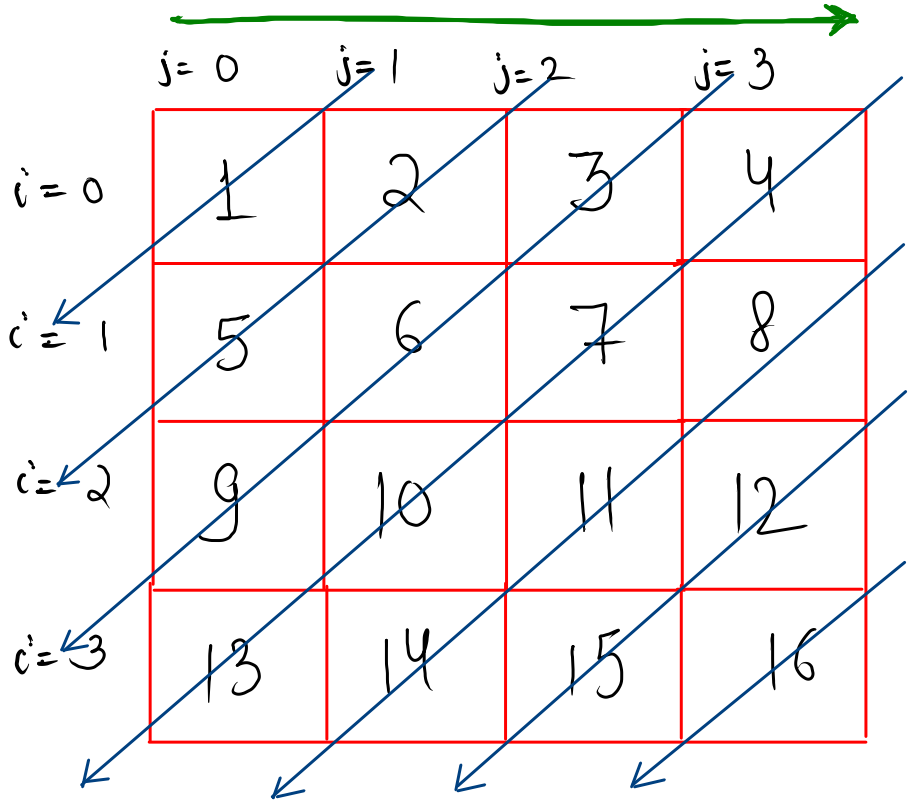
Print the matrix left-diagonal wise



starting

(0,0)

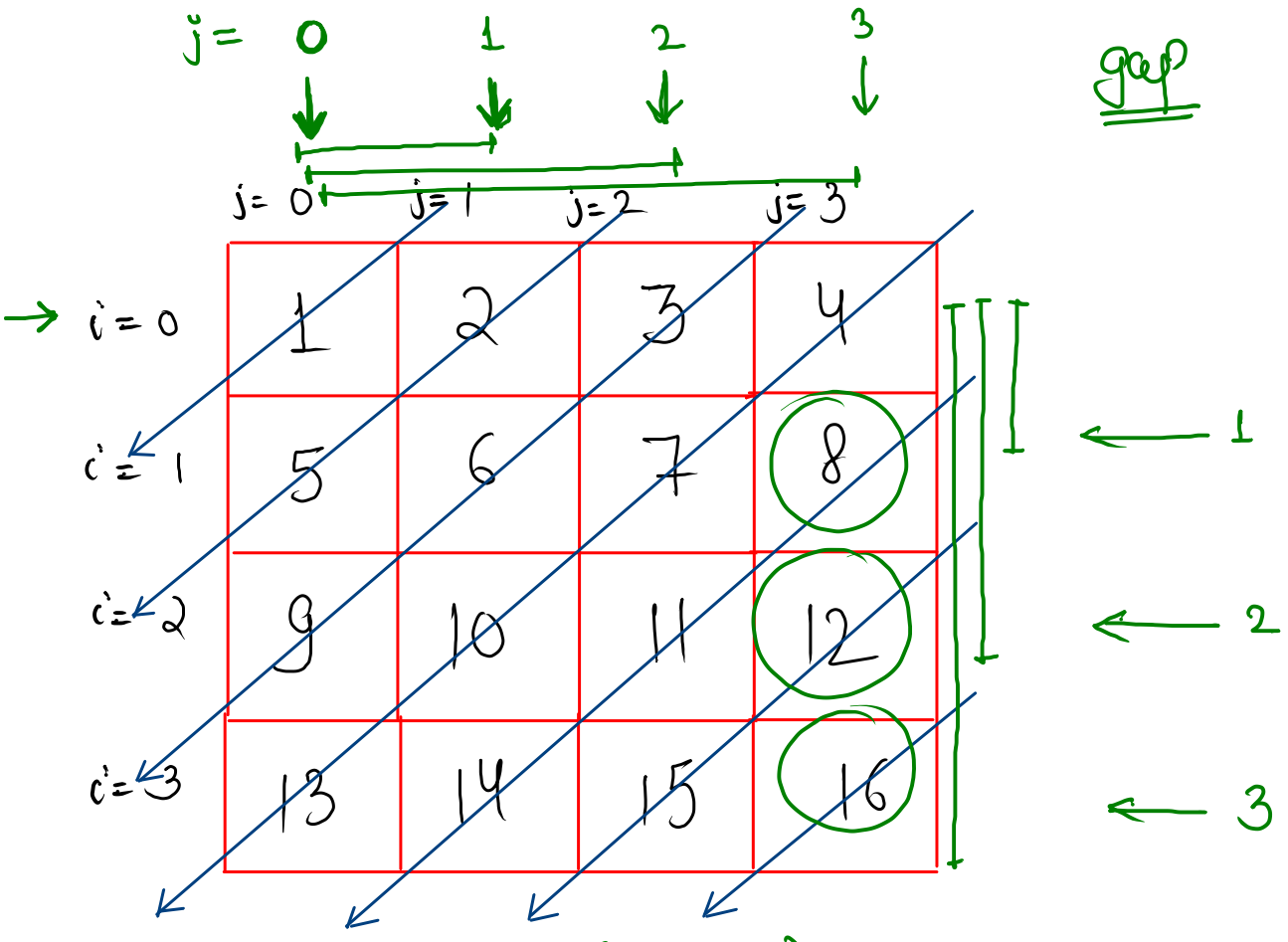
(0,1)

(0,2)

(0,3)

(i, j)

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


j = 0 1 2 3
gap
n = 4
j = 0 j = 1 j = 2 j = 3
i = 0
i = 1
i = 2
i = 3
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16
1
2
3

square (n*n)

Code

```
for (int gap = 0; gap < n; gap++){
    for (int i = 0, j = gap ; j >= 0 ; i++, j--) {
        Syso( arr[i][j] + " ");
    }
}
```

```
for (int gap = 1 ; gap < n ; gap++) {
    for (int i = gap, j = n-1; i < n ; i++, j--){
        Syso( arr[i][j] + " " );
    }
}
```

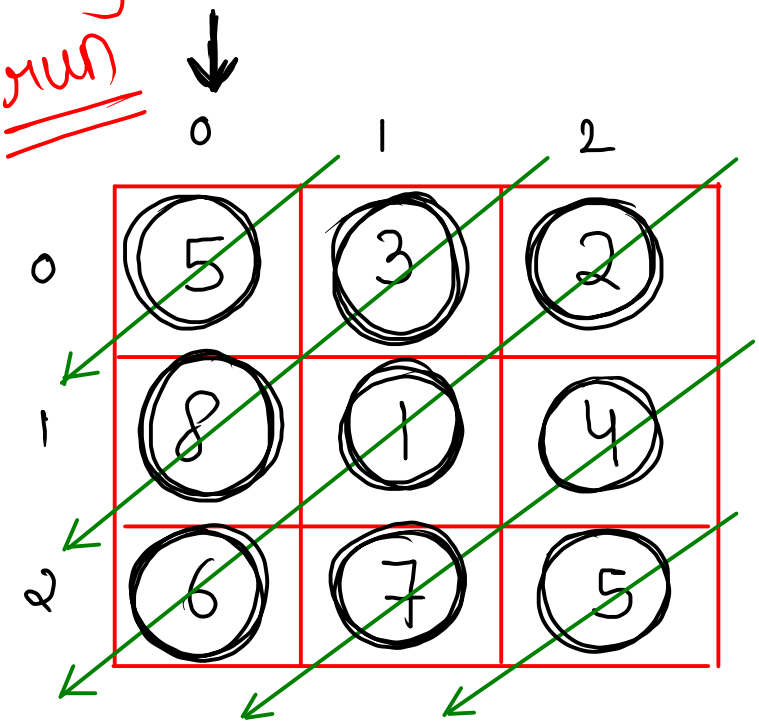
code

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

    diagonal(arr, n);
}
public static void diagonal(int[][] arr, int n) {
    for (int gap = 0; gap < n; gap++) {
        for (int i = 0, j = gap; j >= 0; i++, j--) {
            System.out.print(arr[i][j] + " ");
        }
    }

    for (int gap = 1; gap < n; gap++) {
        for (int i = gap, j = n - 1; i < n; i++, j--) {
            System.out.print(arr[i][j] + " ");
        }
    }
}
```

dry run




```
for (int gap = 0; gap < n; gap++){
    for (int i = 0, j = gap ; j >= 0 ; i++, j--) {
        xxx   Syso( arr[i][j] + " ");
    }
}
```

gap = 0, (0,0) (1,-1) (2,-1)

gap = 1, (0,1) (1,0) (2,-1)

gap = 2, (0,2) (1,1) (2,0) (3,-1)

```
for (int gap = 1 ; gap < n ; gap++) {
    x   for (int i = gap, j = n-1 ; i < n ; i++, j--){
        xx   Syso( arr[i][j] + " " );
    }
}
```

gap = 1, (1,2) (2,1) (3,0)

gap = 2, (2,2) (3,1)

ans :- 5 3 8 2 1 6 4 7 5

Transpose of Matrix of N*N

when all rows become cols & cols become rows

	j=0	j=1	j=2	j=3
i=0	1	2	3	4
i=1	5	6	7	8
i=2	9	10	11	12
i=3	13	14	15	16

transpose

	j=0	j=1	j=2	j=3
i=0	1	5	9	13
i=1	2	6	10	14
i=2	3	7	11	15
i=3	4	8	12	16

$(0,1) \rightarrow (1,0)$

$(0,2) \rightarrow (2,0)$

$(1,2) \rightarrow (2,1)$

Code

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

    transpose(arr, n);
}
public static void transpose(int[][] arr, int n) {
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            if ( i > j ) {
                int temp = arr[i][j];
                arr[i][j] = arr[j][i];
                arr[j][i] = temp;
            }
        }
    }

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

Rotate The Matrix by 90 Degree

$n = 4$

	j=0	j=1	j=2	j=3
i=0	1	2	3	4
i=1	5	6	7	8
i=2	9	10	11	12
i=3	13	14	15	16

transpose

1D

row

i j

	j=0	j=1	j=2	j=3
i=0	1	5	9	13
i=1	2	6	10	14
i=2	3	7	11	15
i=3	4	8	12	16

swap cols from start and end

(row, i) , (row, j)

	0	1	2	3
0	13	9	5	1
1	14	10	6	2
2	15	11	7	3
3	16	12	8	4

Code

1

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[][] arr = new int[n][n];
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            arr[i][j] = scn.nextInt();
        }
    }
    rotate90(arr, n);

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

2

```java
public static void rotate90(int[][] arr, int n) {
    transpose(arr, n);
    swappingCols(arr, n);
}
```

3

```java
public static void swappingCols(int[][] arr, int n) {
    for (int row = 0; row < n; row++) {
        int i = 0;
        int j = n - 1;
        while (i < j) {
            int temp = arr[row][i];
            arr[row][i] = arr[row][j];
            arr[row][j] = temp;

            i++;
            j--;
        }
    }
}
```

4

```java
public static void transpose(int[][] arr, int n) {
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            if ( i > j ) {
                int temp = arr[i][j];
                arr[i][j] = arr[j][i];
                arr[j][i] = temp;
            }
        }
    }
}
```

Rotate The Matrix by 180 Degree

```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[][] arr = new int[n][n];
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            arr[i][j] = scn.nextInt();
        }
    }
    rotate90(arr, n);
    rotate90(arr, n);

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