

## Transpose of matrix

$$\begin{bmatrix} 10 & 20 & 30 \\ 80 & 90 & 100 \\ 40 & 50 & 60 \end{bmatrix}$$

Exchanging the values between rows and columns

1st Row  $\rightarrow 10, 20, 30$

1st column  $\rightarrow 10, 80, 40$

After Exchange

$$\begin{bmatrix} 10 & 80 & 40 \\ 20 & & \\ 30 & & \end{bmatrix}$$

2nd Row  $\rightarrow 80, 90, 100$

2nd Column  $\rightarrow 20, 90, 50$

After Exchange

$$\begin{bmatrix} 10 & 80 & 40 \\ 20 & 90 & 50 \\ 30 & 100 & \end{bmatrix}$$

3rd Row  $\rightarrow 40, 50, 60$

3rd col  $\rightarrow 30, 100, 60$

0	10 00	180 01	240 02
1	20 10	190 11	250 12

→ Transposed Matrix

1	$\begin{matrix} 20 \\ 10 \end{matrix}$	$\begin{matrix} 11 \\ 90 \end{matrix}$	$\begin{matrix} 12 \\ 50 \end{matrix}$	→ Transposed Matrix
2	$\begin{matrix} 30 \\ 20 \end{matrix}$	$\begin{matrix} 100 \\ 21 \end{matrix}$	$\begin{matrix} 60 \\ 22 \end{matrix}$	

0      1      2

Original matrix →

0	$\begin{matrix} 10 \\ 00 \end{matrix}$	$\begin{matrix} 20 \\ 01 \end{matrix}$	$\begin{matrix} 30 \\ 02 \end{matrix}$
1	$\begin{matrix} 80 \\ 10 \end{matrix}$	$\begin{matrix} 90 \\ 11 \end{matrix}$	$\begin{matrix} 100 \\ 12 \end{matrix}$
2	$\begin{matrix} 40 \\ 20 \end{matrix}$	$\begin{matrix} 50 \\ 21 \end{matrix}$	$\begin{matrix} 60 \\ 22 \end{matrix}$

Code:

$$\text{arr}[i][j] = \text{arr}[j][i]$$

$$\text{arr}[0][i] = \text{arr}[i][0]$$

$$\text{arr}[2][i] = \text{arr}[i][2]$$

```

for (int i=0; i<row; i++) {
    for (int j=i; j<col; j++) {
        int temp = arr[i][j];
        arr[i][j] = arr[j][i];
        arr[j][i] = temp;
    }
}

```

0	1	2	
0	$\begin{matrix} 80 \\ 01 \end{matrix}$	$\begin{matrix} 40 \\ 02 \end{matrix}$	
1	$\begin{matrix} 20 \\ 10 \end{matrix}$	$\begin{matrix} 90 \\ 11 \end{matrix}$	
2			

i=0	i=1	i=2
j=0	j=0	j=0
j=1	j=1	j=1
j=2	j=2	j=2

1	50	40	12
2	30	50	<u>60</u>
2	20	21	22

j  
j=2

j  
j=1  
j=2

j  
j=1

```

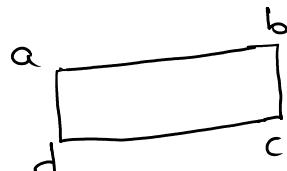
5 public static void main(String[] args) {
6     /* Enter your code here. Read input from STDIN. Print output to STDOUT */
7     Scanner sc = new Scanner(System.in);
8     int n = sc.nextInt();
9     int arr[][] = new int[n][n];
10    for(int i=0;i<n;i++){
11        for(int j=0;j<n;j++){
12            arr[i][j] = sc.nextInt();
13        }
14    }
15    for(int i=0;i<n;i++){
16        for(int j=i;j<n;j++){
17            int temp = arr[i][j];
18            arr[i][j] = arr[j][i];
19            arr[j][i] = temp;
20        }
21    }
22
23    for(int i=0;i<n;i++){
24        for(int j=0;j<n;j++){
25            System.out.print(arr[i][j]+" ");
26        }
27        System.out.println();
28    }
29 }
30 }
31 }
```

# Rotate matrix by 90°

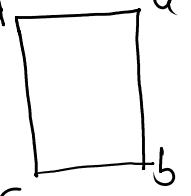
10	20	30
60	90	40
15	18	19



90° rotate



90° rotate



a	0	1	2	b
0	10 00	20 01	30 02	
1	60 10	90 11	40 12	
2	15 20	18 21	19 22	c
d				

90° rotate

d	0	1	2	a
0	15 00	60 01	10 02	
1	18 10	90 11	20 12	
2	19 20	40 21	30 22	b
c				

Transpose + Reverse = Rotate by 90°

① Transpose.

	0	1	2
0	10 00	20 01	30 02
1	60 10	90 11	40 12
2	15 20	18 21	19 22



10	60	15
20	90	18
30	40	19

Original Matrix

② Reverse (Mirror Image) → Within a row, Value at start exchange with end and so on

	0	1	2
0	10	60	15

0	1	2
1	60	15

	0	1	2
0	10 <sub>00</sub>	60 <sub>01</sub>	15 <sub>02</sub>
1	20 <sub>10</sub>	90 <sub>11</sub>	18 <sub>12</sub>
2	30 <sub>20</sub>	40 <sub>21</sub>	19 <sub>22</sub>

→ Rotated by 90°

Col = n

$$00 \rightarrow 02$$

$$01 \rightarrow 01$$

$$10 \rightarrow 12$$

$$11 \rightarrow 11$$

$$20 \rightarrow 22$$

$$21 \rightarrow 21$$

$$[0][0] \rightarrow [0][n-1]$$

$$[0][1] \rightarrow [0][n-2]$$

$$[0][1]$$

$$\rightarrow [1][0] \rightarrow [1][2]^{n-1}$$

$$[1][1] \rightarrow [1][1]^{n-2}$$

$$\times [1][2] \rightarrow [1][0]^{n-3}$$

$$[2][0] \rightarrow [2][2]^{n-1}$$

$$[2][1] \rightarrow [2][1]^{n-2}$$

$$\times [2][2] \rightarrow [2][0]^{n-3}$$

1st Approach:

```
for(int i=0; i<row; i++) {
    for(int j=0; j<col/2; j++) {
        int temp = arr[i][j];
        arr[i][j] = arr[i][col-1-j];
        arr[i][col-1-j] = temp;
    }
}
```

$\stackrel{n=\text{col.}}{=} \begin{matrix} j & col-1-j \\ 0, & 1, n-1-0 \\ 1, & n-2, n-1-1 \\ 2, & n-3, n-1-2 \\ \dots & \dots \end{matrix}$

	0	1	2	3
0	10	20	30	40
1	40	30	20	10

	0	1	2	3
0	40	20	30	10
1	40	30	20	10

40	30	20	10
----	----	----	----

2nd Approach.

```

for(int i=0; i< row; i++) {
    int left=0, right = col-1;
    while(left < right) {
        int temp = arr[i][left];
        arr[i][left] = arr[i][right];
        arr[i][right] = temp;
        left++;
        right--;
    }
}

```

$\begin{matrix} 0 & 1 & 2 \\ 10 & 60 & 15 \end{matrix}$

$\begin{matrix} 15 & 60 & 10 \\ \text{left} & \downarrow & \text{right} \\ 10 & 60 & 15 & 90 & 40 \end{matrix} \rightarrow 40 \ 90 \ 15 \ 60 \ 10$

$\begin{matrix} 40 & 90 & 15 & 60 & 10 \\ \text{left} = 0, \text{right} = n-1 \\ \text{while}(\text{left} < \text{right}) { \\ \text{swap} (\text{arr}[i][\text{left}], \text{arr}[i][\text{right}]) } \end{matrix}$

`left++ ;  
right-- ;`

`}`