**Поворот на 90 градусов**

**int** arrTr[][] = **new int**[m][n];  
**for** (**int** i = 0; i < m; i++) {  
 **for** (**int** j = 0; j < n; j++) {  
 arrTr[i][j] = arr[n - 1 - j][i];  
 }  
}

**Заполнение относительно побочной диагонали**

int arr[][] = new int[n][n];

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++) {

if (i + j == n - 1)

arr[i][j] = 1;

else if (i + j < n - 1)

arr[i][j] = 0;

else

arr[i][j] = 2;

}

}

**Симметричность матрицы**

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

int arr[][] = new int[n][n];

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++) {

arr[i][j] = sc.nextInt();

}

}

boolean fl = false;

for (int i = 0; i < n; i++) {

for (int j = i + 1; j < n; j++) {

if(arr[i][j] != arr[j][i])

fl = true;

}

}

System.out.println(fl == true ? "no" : "yes");

**Змейка**

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

int m = sc.nextInt();

int arr[][] = new int[n][m];

for (int i = 0; i < n \* m; i++) {

arr[i / m][i % m] = (i / m) \* (i % m);

}

for (int i = 0; i < n; i++) {

for (int j = 0; j < m; j++) {

System.out.printf("%4d",arr[i][j]);

}

System.out.print('\n'); } }}