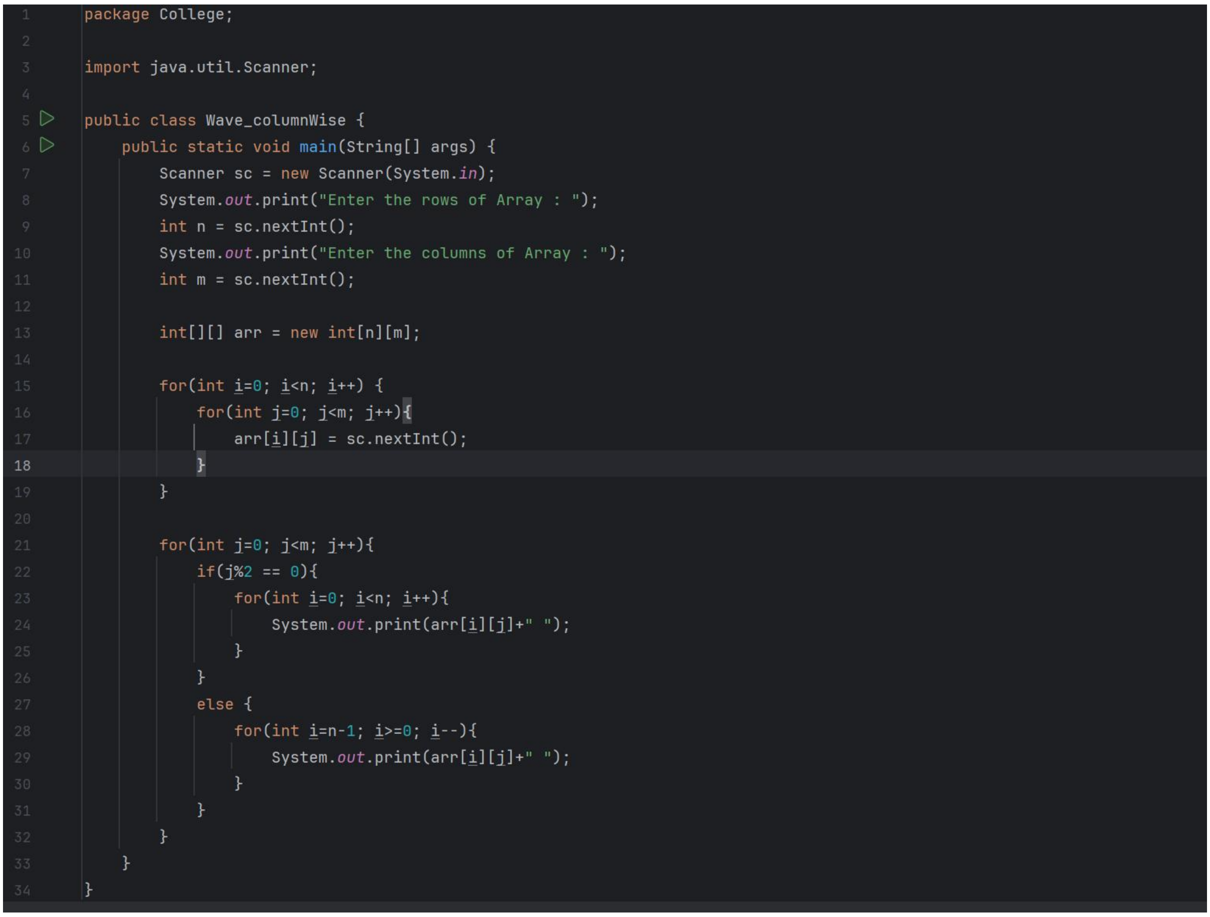
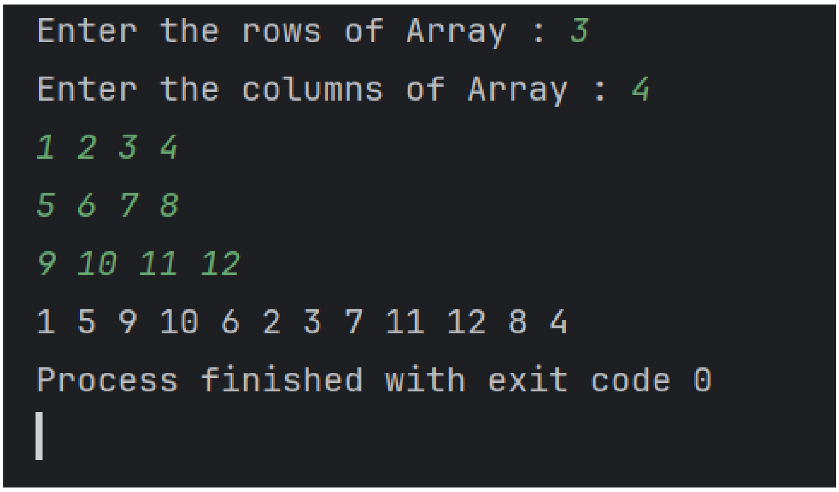
Problem Statement – Wave Form Traversal

SOLUTION :



```java
package College;

import java.util.Scanner;

public class Wave_columnWise {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the rows of Array : ");
        int n = sc.nextInt();
        System.out.print("Enter the columns of Array : ");
        int m = sc.nextInt();

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

        for(int i=0; i<n; i++) {
            for(int j=0; j<m; j++){
                arr[i][j] = sc.nextInt();
            }
        }

        for(int j=0; j<m; j++){
            if(j%2 == 0){
                for(int i=0; i<n; i++){
                    System.out.print(arr[i][j]+" ");
                }
            }
            else {
                for(int i=n-1; i>=0; i--){
                    System.out.print(arr[i][j]+" ");
                }
            }
        }
    }
}
```

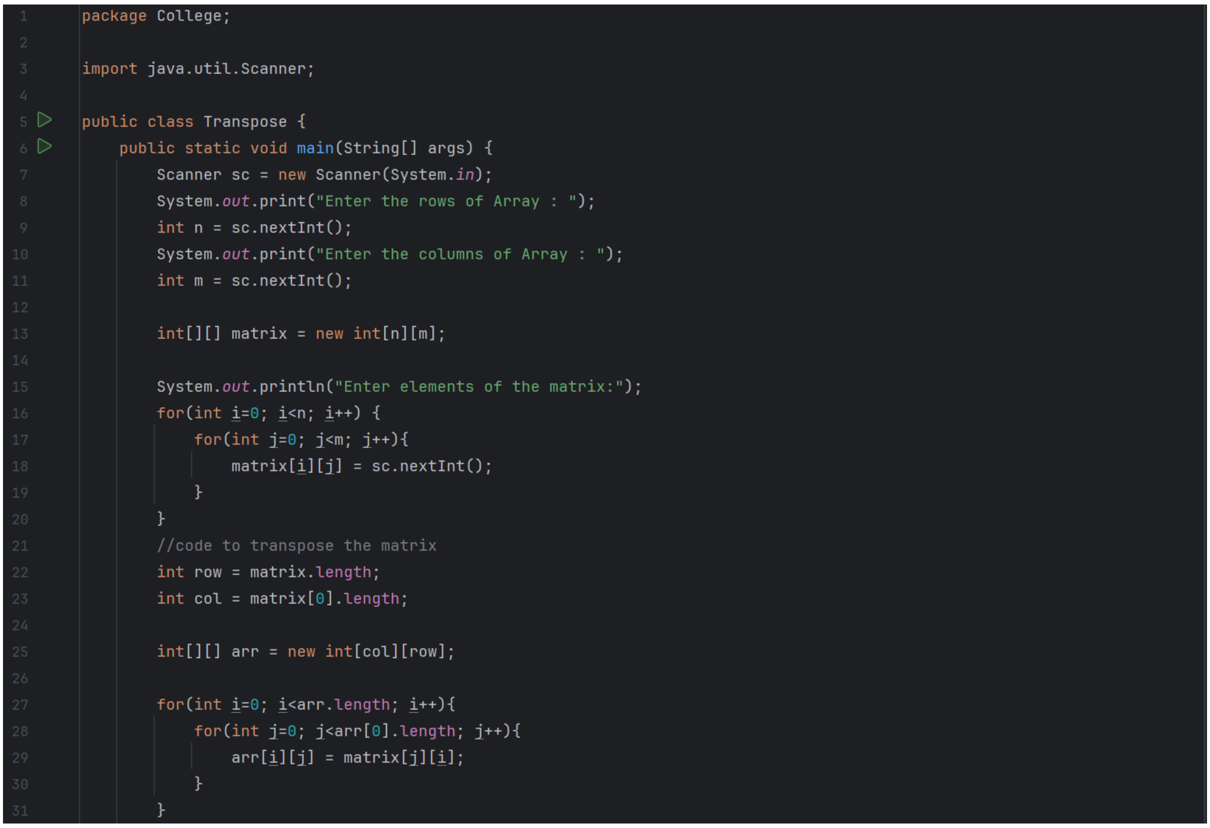
OUTPUT :



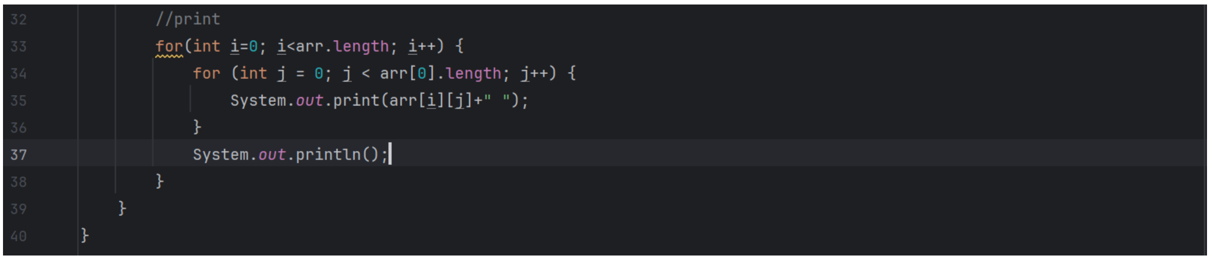
```
Enter the rows of Array : 3
Enter the columns of Array : 4
1 2 3 4
5 6 7 8
9 10 11 12
1 5 9 10 6 2 3 7 11 12 8 4
Process finished with exit code 0
```

Problem Statement – Transpose of a Matrix

SOLUTION :

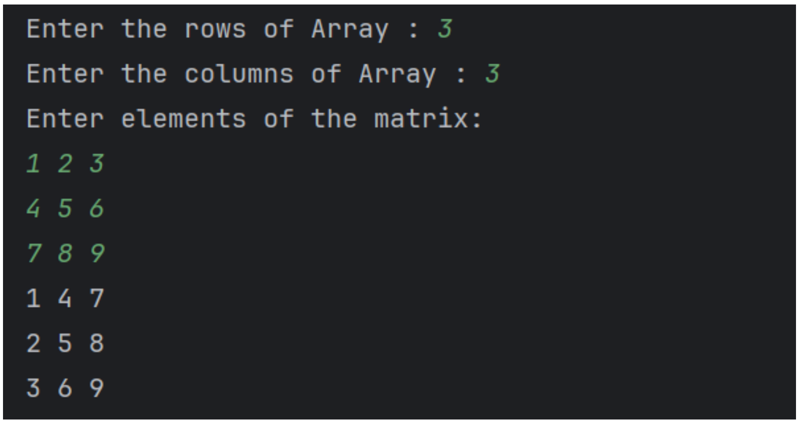


```java
package College;

import java.util.Scanner;

public class Transpose {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the rows of Array : ");
        int n = sc.nextInt();
        System.out.print("Enter the columns of Array : ");
        int m = sc.nextInt();

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

        System.out.println("Enter elements of the matrix:");
        for(int i=0; i<n; i++) {
            for(int j=0; j<m; j++){
                matrix[i][j] = sc.nextInt();
            }
        }
        //code to transpose the matrix
        int row = matrix.length;
        int col = matrix[0].length;

        int[][] arr = new int[col][row];

        for(int i=0; i<arr.length; i++){
            for(int j=0; j<arr[0].length; j++){
                arr[i][j] = matrix[j][i];
            }
        }
```



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

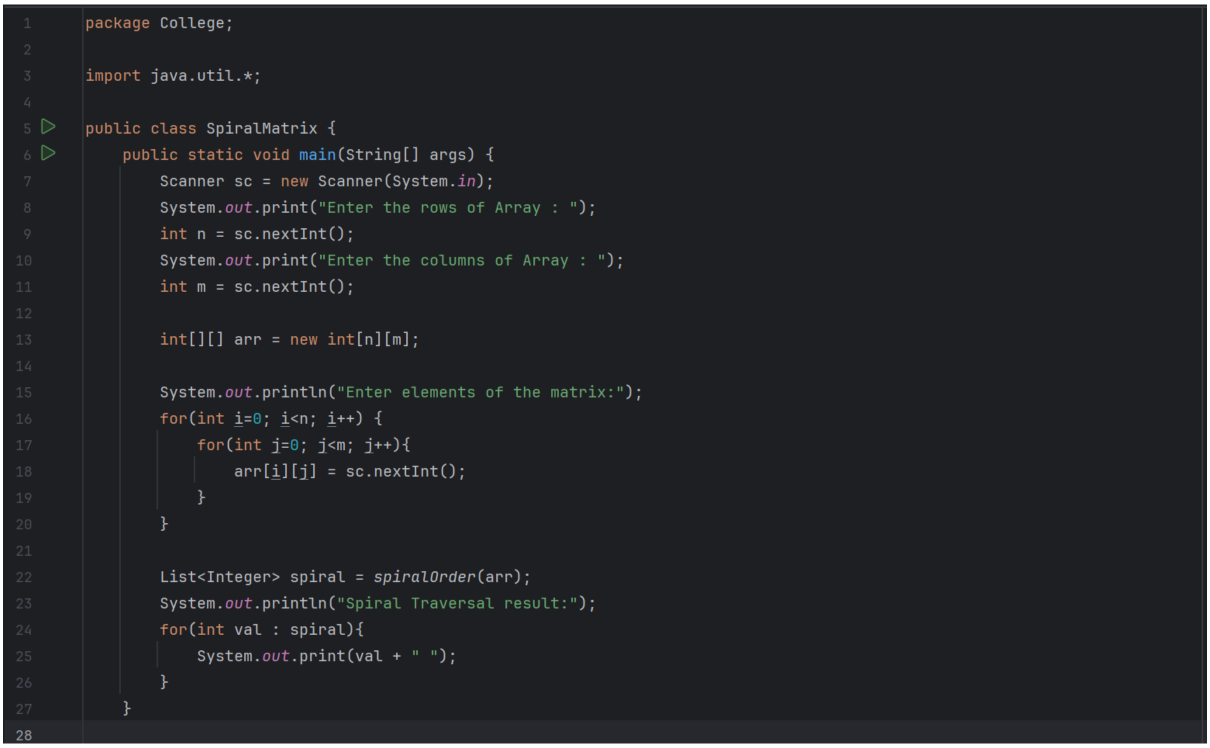
OUTPUT :



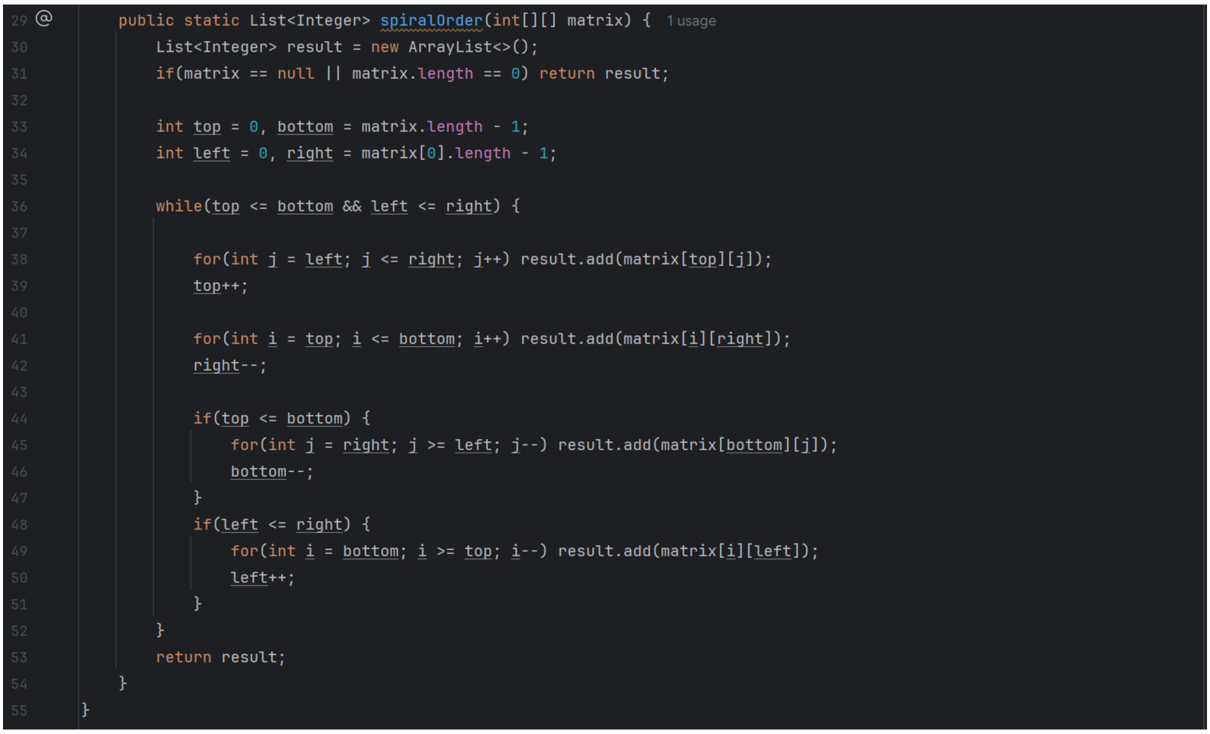
```
Enter the rows of Array : 3
Enter the columns of Array : 3
Enter elements of the matrix:
1 2 3
4 5 6
7 8 9
1 4 7
2 5 8
3 6 9
```

Problem Statement – Spiral Traversal of a Matrix

SOLUTION :

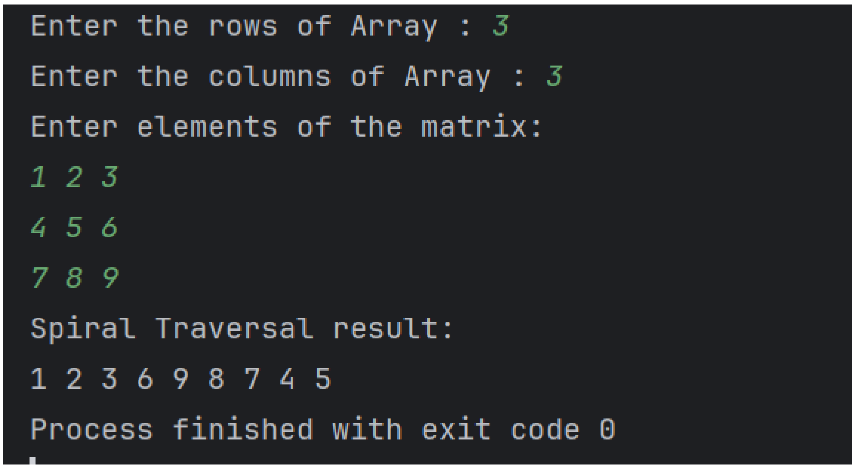


```java
package College;

import java.util.*;

public class SpiralMatrix {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the rows of Array : ");
        int n = sc.nextInt();
        System.out.print("Enter the columns of Array : ");
        int m = sc.nextInt();

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

        System.out.println("Enter elements of the matrix:");
        for(int i=0; i<n; i++) {
            for(int j=0; j<m; j++){
                arr[i][j] = sc.nextInt();
            }
        }

        List<Integer> spiral = spiralOrder(arr);
        System.out.println("Spiral Traversal result:");
        for(int val : spiral){
            System.out.print(val + " ");
        }
    }

```



```java
    public static List<Integer> spiralOrder(int[][] matrix) {
        List<Integer> result = new ArrayList<>();
        if(matrix == null || matrix.length == 0) return result;

        int top = 0, bottom = matrix.length - 1;
        int left = 0, right = matrix[0].length - 1;

        while(top <= bottom && left <= right) {

            for(int j = left; j <= right; j++) result.add(matrix[top][j]);
            top++;

            for(int i = top; i <= bottom; i++) result.add(matrix[i][right]);
            right--;

            if(top <= bottom) {
                for(int j = right; j >= left; j--) result.add(matrix[bottom][j]);
                bottom--;
            }
            if(left <= right) {
                for(int i = bottom; i >= top; i--) result.add(matrix[i][left]);
                left++;
            }
        }
        return result;
    }
}
```

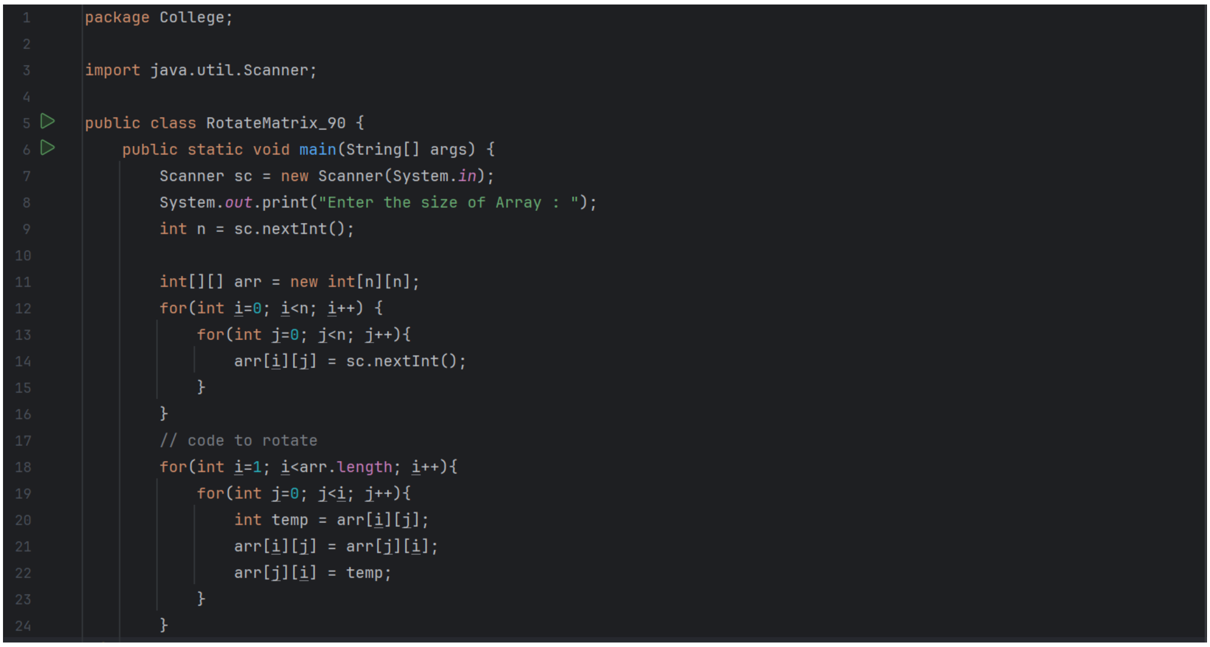
OUTPUT :



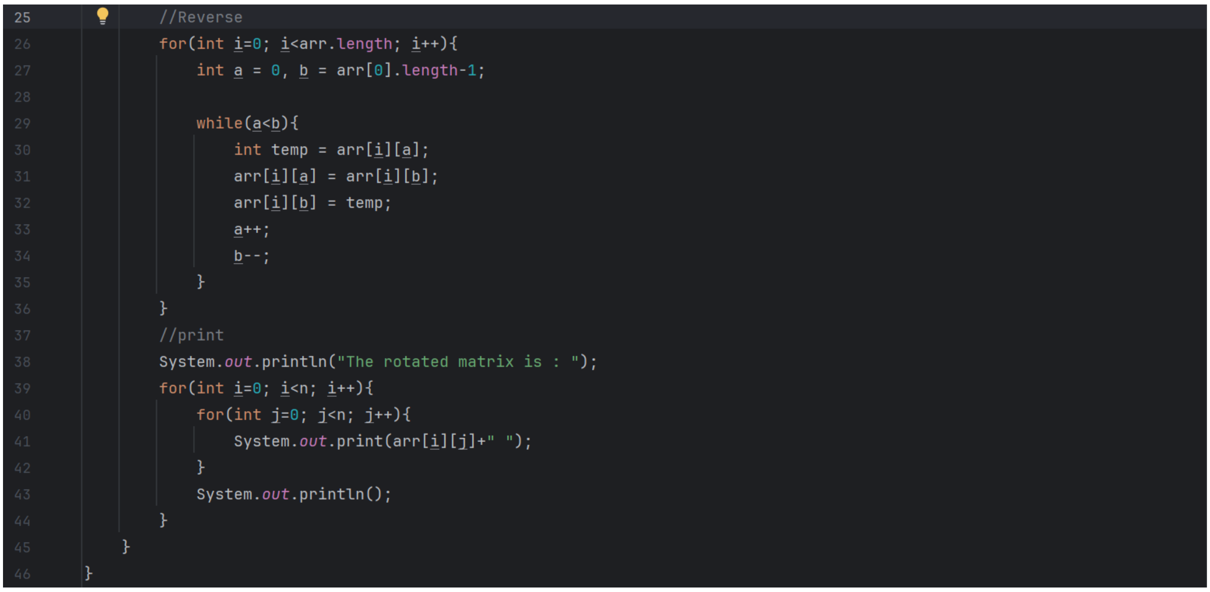
```
Enter the rows of Array : 3
Enter the columns of Array : 3
Enter elements of the matrix:
1 2 3
4 5 6
7 8 9
Spiral Traversal result:
1 2 3 6 9 8 7 4 5
Process finished with exit code 0
```

Problem Statement – Rotate Matrix by 90° Clockwise

SOLUTION :

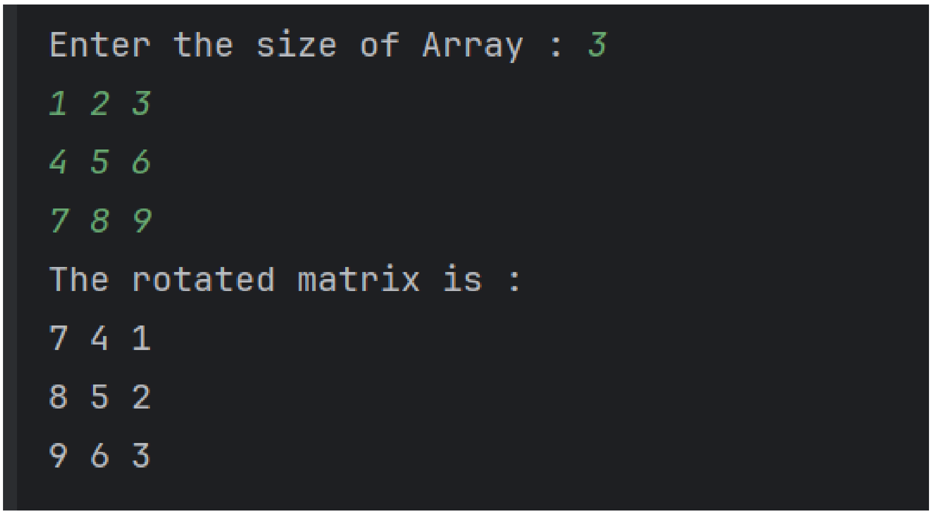


```java
package College;

import java.util.Scanner;

public class RotateMatrix_90 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the size of Array : ");
        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();
            }
        }
        // code to rotate
        for(int i=1; i<arr.length; i++){
            for(int j=0; j<i; j++){
                int temp = arr[i][j];
                arr[i][j] = arr[j][i];
                arr[j][i] = temp;
            }
        }
```



```
        //Reverse
        for(int i=0; i<arr.length; i++){
            int a = 0, b = arr[0].length-1;

            while(a<b){
                int temp = arr[i][a];
                arr[i][a] = arr[i][b];
                arr[i][b] = temp;
                a++;
                b--;
            }
        }
        //print
        System.out.println("The rotated matrix is : ");
        for(int i=0; i<n; i++){
            for(int j=0; j<n; j++){
                System.out.print(arr[i][j]+" ");
            }
            System.out.println();
        }
    }
}
```

OUTPUT :



```
Enter the size of Array : 3
1 2 3
4 5 6
7 8 9
The rotated matrix is :
7 4 1
8 5 2
9 6 3
```