

CHRIST (Deemed to be University)

Department of Computer Science

MSc – Artificial Intelligence and Machine Learning

Name: Joel Joseph Motha

Reg No.: 2448521

Course: Java Programming

Component: Lab Practical CIA 2

Description:

- The code takes an input of a m x n matrix and then adds +1 to the odd numbers and +2 to the even numbers.
- It then swaps the columns one step ahead and then swaps the last row with the penultimate row.

Program:

```
import java.util.*;
```

```
public class MatrixTransformation {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
  
        System.out.println("Enter the number of rows for the matrix:");  
        int rows = scanner.nextInt();  
        System.out.println("Enter the number of columns for the matrix:");  
        int cols = scanner.nextInt();  
  
        int[][] inputMatrix = readMatrixInput(rows, cols, scanner);  
        System.out.println("Input Matrix:");  
        displayMatrix(inputMatrix);  
  
        int[][] transformedMatrix = transformMatrixValues(inputMatrix);  
  
        transformedMatrix = shiftColumns(transformedMatrix);  
  
        if (rows > 2) {  
            transformedMatrix = swapRows(transformedMatrix, 1, 2);  
        }  
    }  
}
```

```

    }

    System.out.println("\nOutput Matrix:");
    displayMatrix(transformedMatrix);

    scanner.close();
}

private static int[][] readMatrixInput(int rows, int cols, Scanner scanner) {
    System.out.println("Enter elements for the matrix:");
    int[][] matrix = new int[rows][cols];

    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            System.out.print("Matrix[" + (i + 1) + "][" + (j + 1) + "]: ");
            matrix[i][j] = scanner.nextInt();
        }
    }
    return matrix;
}

private static int[][] transformMatrixValues(int[][] matrix) {
    int rows = matrix.length;
    int cols = matrix[0].length;
    int[][] result = new int[rows][cols];

    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            if (matrix[i][j] % 2 == 0) {
                result[i][j] = matrix[i][j] + 2;
            } else {
                result[i][j] = matrix[i][j] + 1;
            }
        }
    }
}

```

```

        }
    }
}
return result;
}

```

```

private static int[][] shiftColumns(int[][] matrix) {
    int rows = matrix.length;
    int cols = matrix[0].length;
    int[][] shiftedMatrix = new int[rows][cols];

    for (int i = 0; i < rows; i++) {
        shiftedMatrix[i][0] = matrix[i][1];
        shiftedMatrix[i][1] = matrix[i][2];
        if (cols > 2) {
            shiftedMatrix[i][2] = matrix[i][0];
        }
    }
    return shiftedMatrix;
}

```

```

private static int[][] swapRows(int[][] matrix, int row1, int row2) {
    int[] temp = matrix[row1];
    matrix[row1] = matrix[row2];
    matrix[row2] = temp;
    return matrix;
}

```

```

private static void displayMatrix(int[][] matrix) {
    for (int[] row : matrix) {
        for (int value : row) {
            System.out.print(value + " ");

```

```
    }  
    System.out.println();  
}  
}  
}
```

Output:

```
Enter the number of rows for the matrix:  
3  
Enter the number of columns for the matrix:  
3  
Enter elements for the matrix:  
Matrix[1][1]: 4  
Matrix[1][2]: 5  
Matrix[1][3]: 6  
Matrix[2][1]: 7  
Matrix[2][2]: 5  
Matrix[2][3]: 3  
Matrix[3][1]: 5  
Matrix[3][2]: 4  
Matrix[3][3]: 4  
Input Matrix:  
4 5 6  
7 5 3  
5 4 4  
  
Output Matrix:  
6 8 6  
6 6 6  
6 4 8
```