

Slip No. 29(Total Marks: 30)

**1. Write a Java program to accept 5 numbers using command line arguments, sort, and display them – 10 Marks**

```
java
Copy code
import java.util.Arrays;

public class SortCommandLineArgs {
    public static void main(String[] args) {
        if (args.length != 5) {
            System.out.println("Please provide exactly 5 numbers as command
line arguments.");
            return;
        }

        int[] numbers = new int[5];
        for (int i = 0; i < args.length; i++) {
            numbers[i] = Integer.parseInt(args[i]);
        }

        Arrays.sort(numbers);
        System.out.println("Sorted numbers:");
        for (int number : numbers) {
            System.out.print(number + " ");
        }
    }
}
```

**2. Write a menu-driven program to perform operations on a multidimensional array (matrix) – 20 Marks**

```
java
Copy code
import java.util.Scanner;

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

        int[][] matrix1 = new int[rows][cols];
        int[][] matrix2 = new int[rows][cols];
        int[][] result = new int[rows][cols];

        System.out.println("Enter elements of Matrix 1:");
        inputMatrix(matrix1, sc);

        System.out.println("Enter elements of Matrix 2:");
        inputMatrix(matrix2, sc);
    }
}
```

```

boolean exit = false;

while (!exit) {
    System.out.println("\nMenu:");
    System.out.println("1. Addition");
    System.out.println("2. Multiplication");
    System.out.println("3. Transpose of Matrix 1");
    System.out.println("4. Exit");
    System.out.print("Enter your choice: ");
    int choice = sc.nextInt();

    switch (choice) {
        case 1:
            addMatrices(matrix1, matrix2, result);
            System.out.println("Result of Addition:");
            displayMatrix(result);
            break;
        case 2:
            multiplyMatrices(matrix1, matrix2, result);
            System.out.println("Result of Multiplication:");
            displayMatrix(result);
            break;
        case 3:
            System.out.println("Transpose of Matrix 1:");
            transposeMatrix(matrix1);
            break;
        case 4:
            exit = true;
            break;
        default:
            System.out.println("Invalid choice. Please try again.");
    }
}
sc.close();
}

private static void inputMatrix(int[][] matrix, Scanner sc) {
    for (int i = 0; i < matrix.length; i++) {
        for (int j = 0; j < matrix[i].length; j++) {
            System.out.print("Enter element [" + i + "][" + j + "]: ");
            matrix[i][j] = sc.nextInt();
        }
    }
}

private static void addMatrices(int[][] mat1, int[][] mat2, int[][]
result) {
    for (int i = 0; i < mat1.length; i++) {
        for (int j = 0; j < mat1[i].length; j++) {
            result[i][j] = mat1[i][j] + mat2[i][j];
        }
    }
}

private static void multiplyMatrices(int[][] mat1, int[][] mat2, int[][]
result) {

```

```

        for (int i = 0; i < mat1.length; i++) {
            for (int j = 0; j < mat2[0].length; j++) {
                result[i][j] = 0;
                for (int k = 0; k < mat1[0].length; k++) {
                    result[i][j] += mat1[i][k] * mat2[k][j];
                }
            }
        }
    }

    private static void transposeMatrix(int[][] matrix) {
        for (int i = 0; i < matrix[0].length; i++) {
            for (int j = 0; j < matrix.length; j++) {
                System.out.print(matrix[j][i] + " ");
            }
            System.out.println();
        }
    }

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

```