NAME: Priyansh Salian

Batch: C-31

Roll No. : 2003148

EXPERIMENT NO. 6

Aim:

Programs on two dimensional arrays.

Theory:

In such case, data is stored in row and column based index (also known as matrix form).

2D array can be defined as an array of arrays. The 2D array is organized as matrices which can be represented as the collection of rows and columns.

However, 2D arrays are created to implement a relational database look alike data structure. It provides ease of holding bulk of data at once which can be passed to any number of functions wherever required. **Syntax to Declare Two Dimensional Array in Java int arr[max_rows][max_columns];** dataType[][] arrayRefVar; (or) dataType [][]arrayRefVar; (or) dataType arrayRefVar[][]; (or) dataType []arrayRefVar[];

Instantiate

Multidimensional

Array in Java int[][]

arr=new int[3][3];//3 row

and 3 column **Initialize**

Multidimensional

Array in Java

We can use two nested

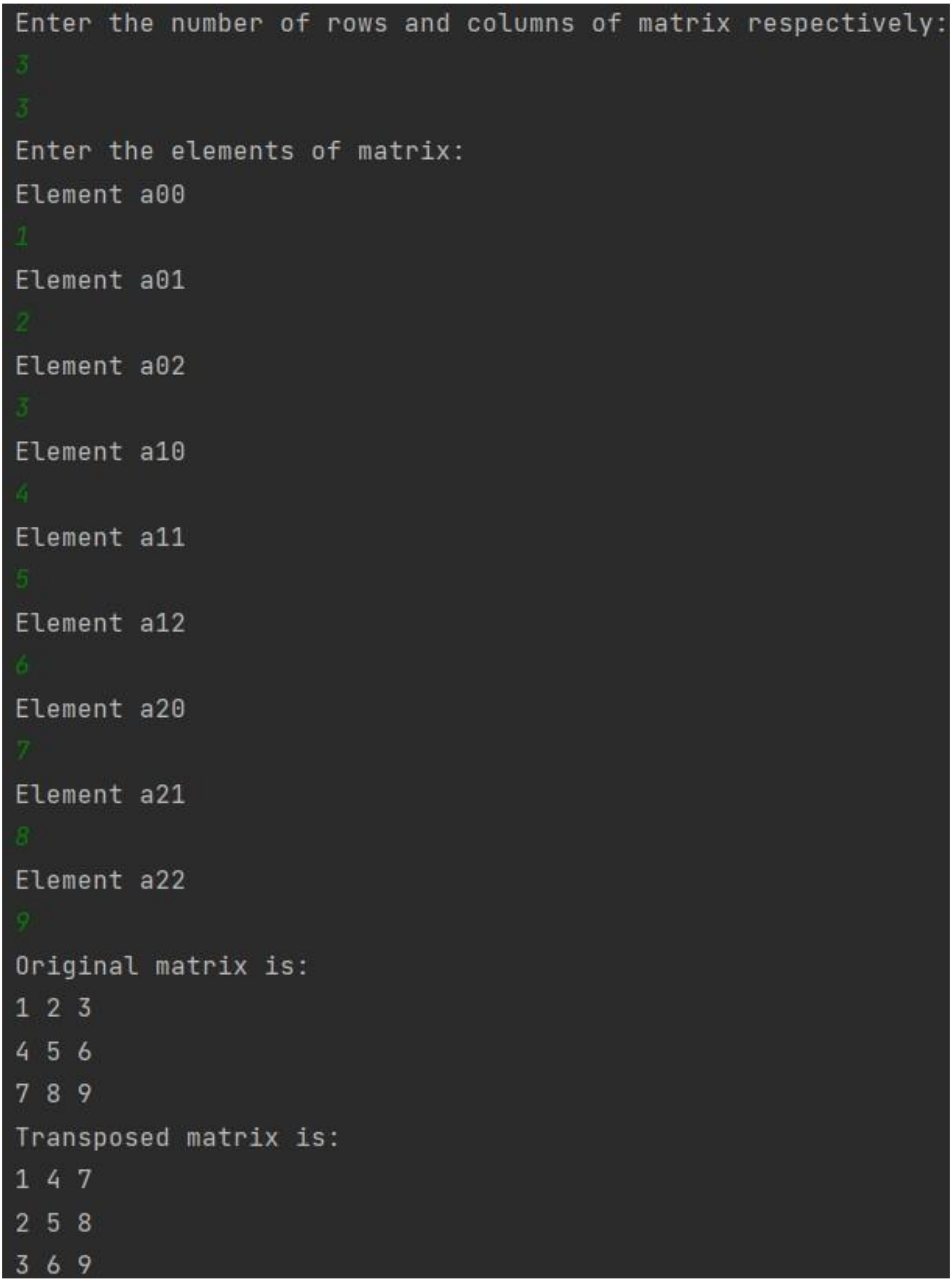
for loops

PROGRAMS:

A. WAP to find Transpose of a matrix. ***Program:***

```
import java.util.Scanner; public class
Transpose {     public static void
main(String[] args) {         Scanner s =
new Scanner(System.in);
      System.out.println("Enter the number of rows and columns of matrix
respectively:");         int row = s.nextInt();         int col = s.nextInt();
int mat[][] = new int[row][col];
      int matTranspose[][] = new int[col][row];
      System.out.println("Enter the elements of matrix:");
for(int i=0;i<row;i++){             for(int j=0;j<col;j++){
            System.out.println("Element a"+i+j);
            mat[i][j] = s.nextInt();
matTranspose[j][i] = mat[i][j];
         }
      }
      System.out.println("Original matrix is:");
for(int i=0;i<row;i++){             for(int
j=0;j<col;j++){
            System.out.print(mat[i][j]+" ");
         }
         System.out.println();
      }
      System.out.println("Transposed matrix is:");
      for(int i=0;i<col;i++){
for(int j=0;j<row;j++){
            System.out.print(matTranspose[i][j]+" ");
         }
         System.out.println();
      }


   }
}
```

Output:



```
Enter the number of rows and columns of matrix respectively:
3
3
Enter the elements of matrix:
Element a00
1
Element a01
2
Element a02
3
Element a10
4
Element a11
5
Element a12
6
Element a20
7
Element a21
8
Element a22
9
Original matrix is:
1 2 3
4 5 6
7 8 9
Transposed matrix is:
1 4 7
2 5 8
3 6 9
```

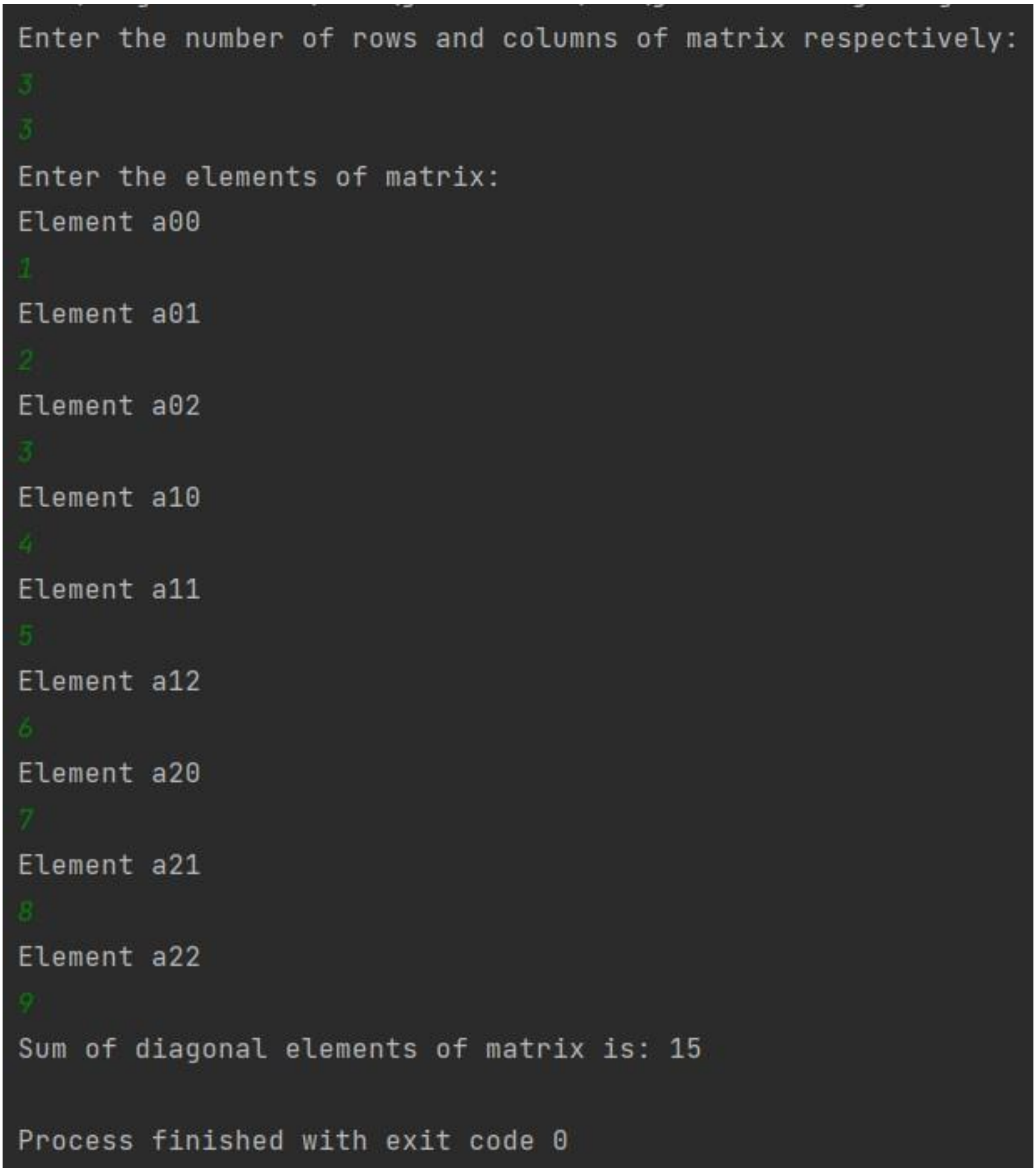
B. WAP to pass a 2-D matrix to a function which determines if it is a square matrix. If not program should come to end else the program should find the sum of all diagonal elements of matrix.

Program:

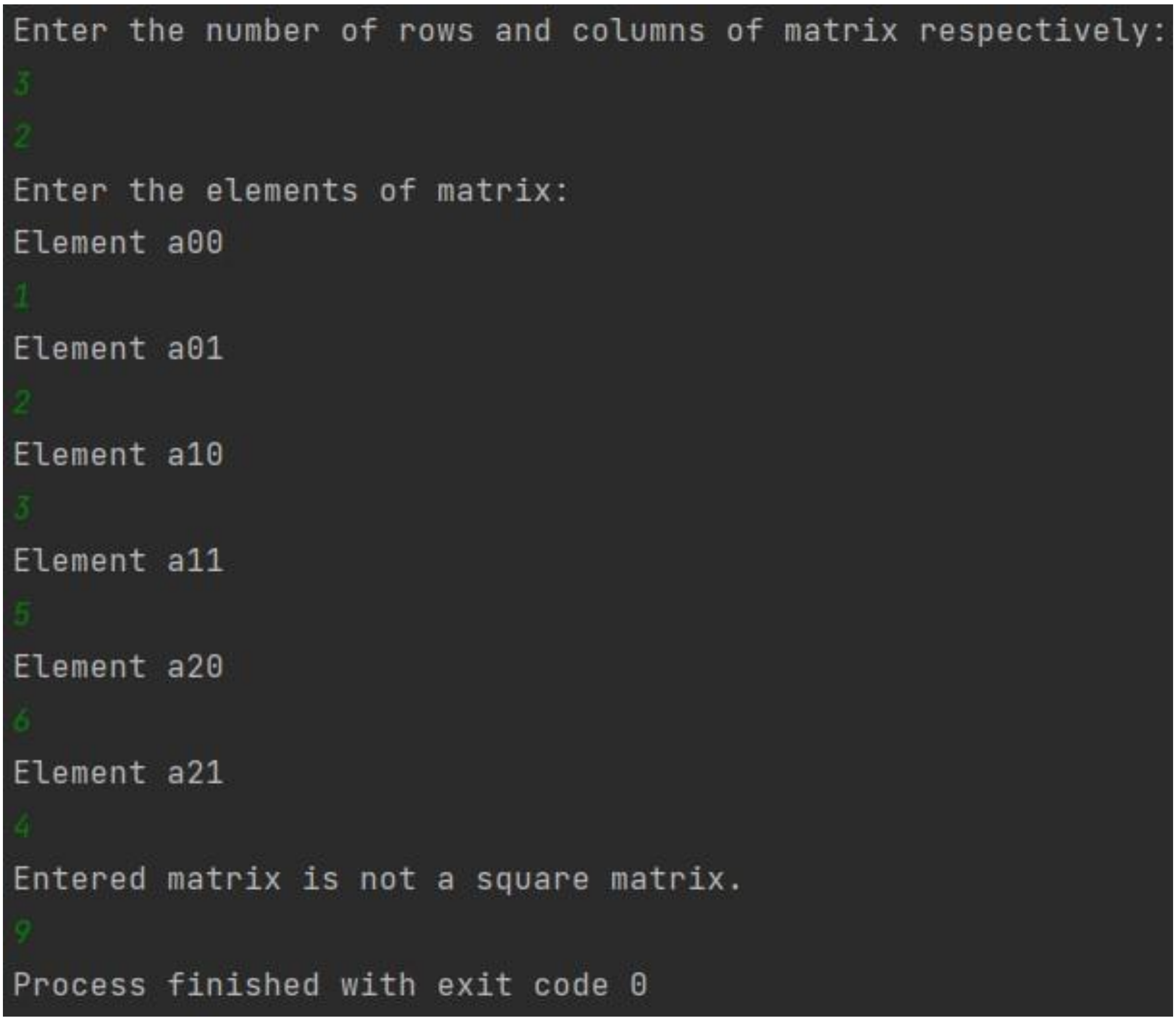
```
import java.util.Scanner;

class SumOfElements{
    public boolean checkSqMat(int a, int b){
if(a==b){             return true;
        }
else{
            return false;
        }
    }
    public int sum(int arr[][],int row,int col){
        int sum = 0;         for(int
i=0;i<row;i++){
for(int j=0;j<col;j++){
if(i==j){
                    sum = sum+arr[i][j];
                }
            }
        }
        return sum;
    }
}
public class SumOfDiagonalElements {
public static void main(String[] args) {
Scanner s = new Scanner(System.in);
        System.out.println("Enter the number of rows and columns of matrix
respectively:");        int row = s.nextInt();        int col = s.nextInt();
        int mat[][] = new int[row][col];
        System.out.println("Enter the elements of matrix:");
for(int i=0;i<row;i++){             for(int j=0;j<col;j++){
                System.out.println("Element a"+i+j);
mat[i][j] = s.nextInt();
            }
        }
        SumOfElements se = new SumOfElements();

        if(se.checkSqMat(row,col)){
            System.out.println("Sum of diagonal elements of matrix is:
"+se.sum(mat,row,col));
```

```
        }
    else{
            System.out.println("Entered matrix is not a square matrix.");
        }


      }
    }
```

<u>Output:</u>



```
Enter the number of rows and columns of matrix respectively:
3
3
Enter the elements of matrix:
Element a00
1
Element a01
2
Element a02
3
Element a10
4
Element a11
5
Element a12
6
Element a20
7
Element a21
8
Element a22
9
Sum of diagonal elements of matrix is: 15

Process finished with exit code 0
```



```
Enter the number of rows and columns of matrix respectively:
3
2
Enter the elements of matrix:
Element a00
1
Element a01
2
Element a10
3
Element a11
5
Element a20
6
Element a21
4
Entered matrix is not a square matrix.
9
Process finished with exit code 0
```