Division/ Roll number/ Batch	B/37/B	Name of Student	Gaurav Mishra
Date of Performance	12/05/2022	Date of Submission	16/05/2022

Experiment No. 4 B

Title	A program to calculate the transpose of a matrix
Learning Outcomes	Concept of array Single dimensional array Multidimensional array
Theory	An array is defined as the collection of single type of data items stored at contiguous memory location A multidimensional array is an array of array data. In multidimensional array is stored in the tabular form
Problem Definition	A transpose of the matrix is to be created using two multidimensional array
Algorithm	1. Start 2. Declare matrix a[m][n] of order mxn 3. Read matrix a[m][n] from user 4. Declare matrix b[m][n] of order mxn 5. Transpose the matrix 6. Print matrix b 7. end
Flowchart	

```
                    ┌──────────────┐         ╭─────────╮         ┌──────────────┐
                    │  Scan a[i][j] │        │  End for │        │  i=0; j=0;   │
                    └──────┬───────┘         ╰────┬────╯         └──────┬───────┘
                           │                      │                     │
                           ▼                      ▼                     │
                      ┌─────────┐            ┌─────────┐                │
                      │   j++   │            │   i++   │                │
                      └─────────┘            └─────────┘                │
                                                                        │
                                                                        ▼
                                         ◇ i< 3 ◇ ──── false ────────┐
                                           │ true                    │
                                           ▼                         ▼
                                         ◇ j< 3 ◇                ╭─────────╮
                                    true │      │ false          │ End for │
                                         ▼      ▼                ╰─────────╯
                                   ┌──────────┐  ┌──────────┐
                                   │  print   │  │ Print "\n"│
                                   │  a[i][j] │  └──────────┘
                                   └──────────┘
                                   ┌─────────┐       ┌─────────┐
                                   │   j++   │       │   i++   │
                                   └─────────┘       └─────────┘

                                         ◇ i< 3 ◇ ──── false ────────┐
                                           │ true                    │
                                           ▼                         ▼
                                         ◇ j< 3 ◇                ╭─────────╮
                                    true │      │ false          │ End for │
                                         ▼      ▼                ╰─────────╯
                                ┌──────────────┐  ┌──────────┐
                                │ B[i][j]=a[j][i]│ │ Print "\n"│
                                └──────────────┘  └──────────┘
```

Flowchart

```
        j++            i++


              i< 3 ──── false ──────────┐
               │                         │
              true                       ▼
               │                      ( End for )
               ▼
              j< 3
          true │    false
               ▼        │
          ┌─────────┐   ▼
          │  print  │  Print "\n"
          │ b[i][j] │
          └─────────┘
               │        │
              j++       ▼
               │       i++
               ▼
           ( stop )
```
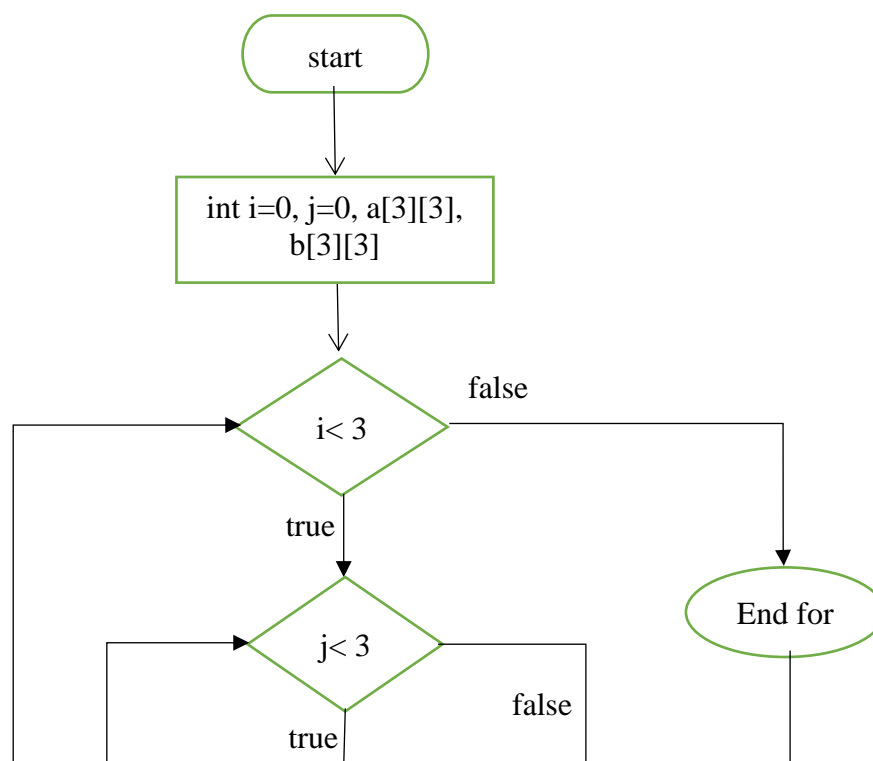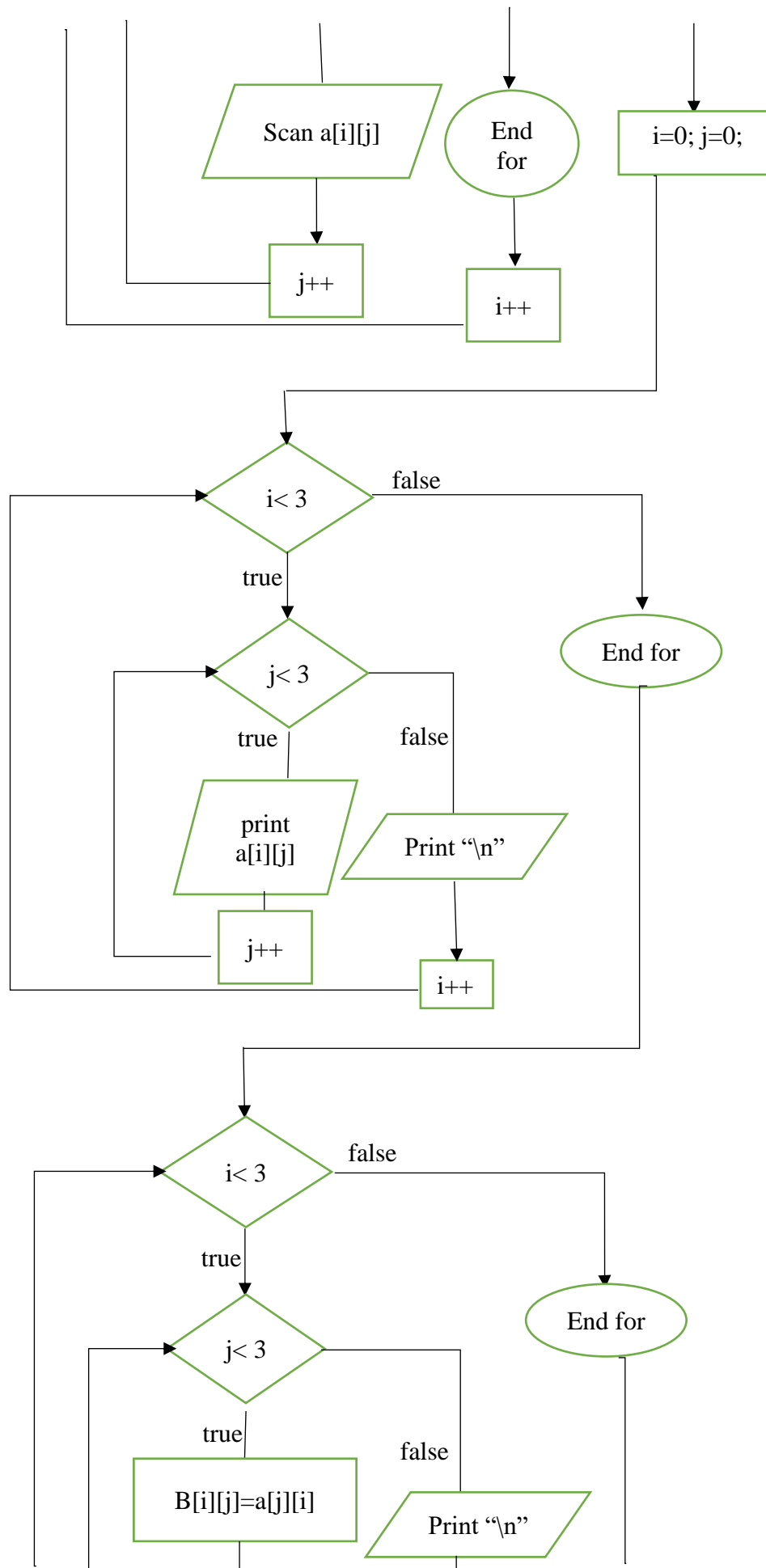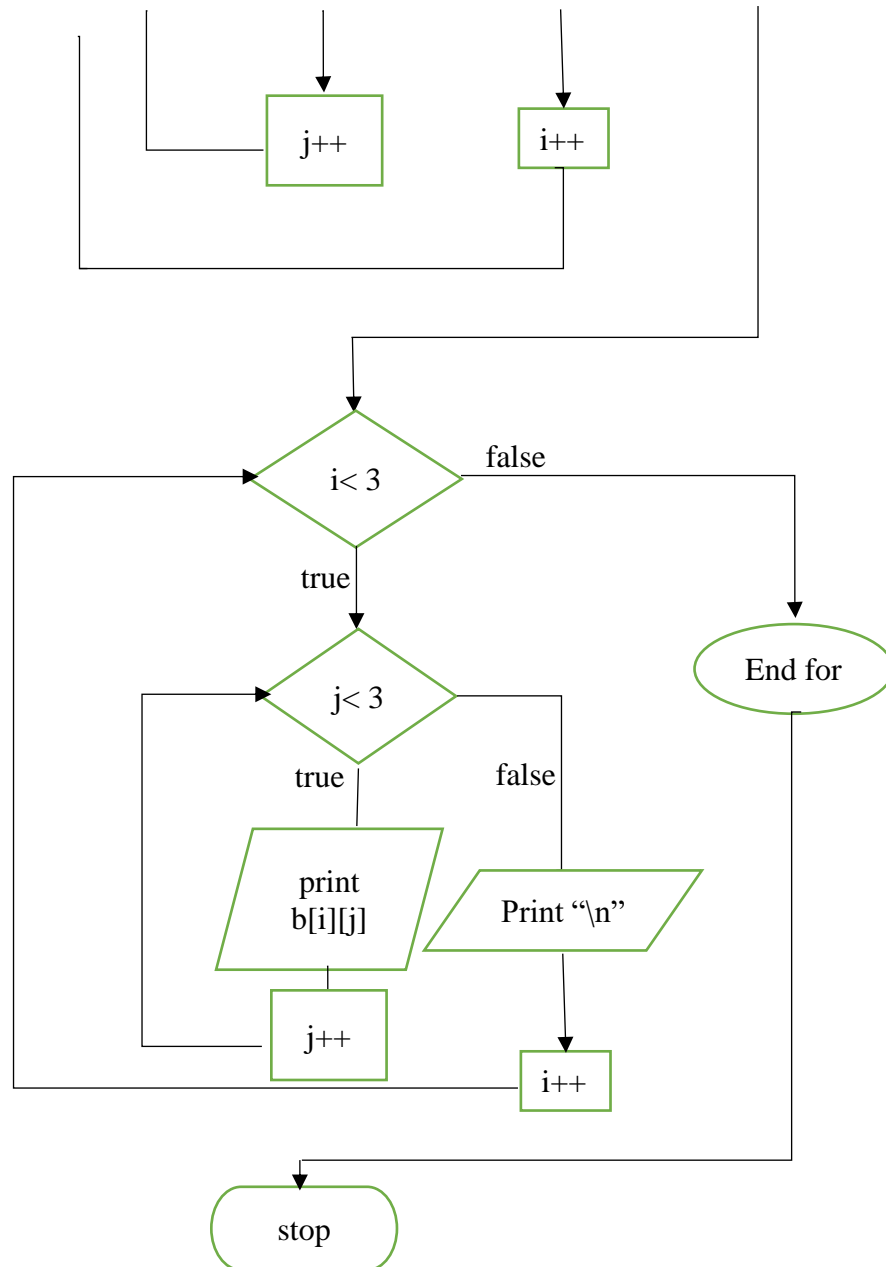
Code	

```c
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int m,n,i,j,mat[10][10],tra[10][10];
    printf("enter  rows and columns \n");
    scanf("%d%d",&m,&n);
    printf("enter the elements of the matrix\n");
    for(i=0;i<m;i++)
    for(j=0;j<n;j++)
      scanf("%d",&mat[i][j]);
     for(i=0;i<m;i++)
    for(j=0;j<n;j++)
     tra[j][i]=mat[i][j];
     printf("transpose of the matrix \n");
      for(i=0;i<m;i++){
```

| | ```
      for(j=0;j<n;j++){
        printf("%d\t",tra[i][j]);
        }
        printf("\n");
      }
      return 0;
    }
``` |
| Output | enter rows and columns
3
3
enter the elements of the matrix
1
2
3
4
5
6
7
8
9
transpose of the matrix
1   4   7
2   5   8
3   6   9 |
| Conclusion | The transpose of the matrix has been made by taking the elements of the first matrix and storing them as a multidimensional array |