Programming lab manual 09

TALHA RAFIQUE

471701

Q.No. 1

```
#include<iostream>

using namespace std;
```

```
int main()

{



        int i, j, s=0;

        int arr[3][3];

        cout<<"Enter array elements : "<<endl;

        for(i=0;i<3;i++)

        {



                for(j=0; j<3; j++)

                {

                cin>>arr[i][j];     }

        }

for(i=0; i<3;i++)

{

        for(j=0; j<3; j++)



        {

                if(i==j || i+j==2)

                {



                        s=s+arr[i][j];







                }
```

```
            }
}
cout<<"thevsum of left and right diagonal is equal to : "<<s<<endl;
return 0;

}
```

Q.No.2

```
#include<iostream>
using namespace std;
int main()
{
            int i,j,s=0;
   int arr[3][3];
   cout<<"Enter the elements of an array : "<<endl;
   for(i=0; i<3; i++)
   {
            for(j=0; j<3;j++)
            {
                        cin>>arr[i][j];

                        }

            }
            for(i=0; i<3; i++)

            {


                        for(j=0; j<3; j++)

                        {


s=s+arr[i][j];



                        }

            }




cout<<"The sum of all elements of an array : "<<s<<endl;

return 0;

}
```

Q.No.3

```
#include<iostream>

using namespace std;

int main()


{

int i,j,s ;

   int arr[3][3];
   cout<<"Enter the elements of an array : "<<endl;
   for(i=0; i<3; i++)
   {
   for(j=0; j<3;j++)
   {
      cout<<"Enter elements in Row  "
<<i+1<<" column  "<<j+1<<" : ";
   cin>>arr[i][j];
      }
   }
for(i=0; i<3; i++)
   {          for(j=0; j<3; j++)
      {
cout<<arr[j][i]<<" ";
      }
cout<<endl;
   }
}
```

Q.No.4

```
#include<iostream>
```

```
using namespace std;

int main()

{

                                                   int matrix_1[3][3]={{1,2,3},

                                                          {4,5,6},

                                                          {7,8,9}};

int matrix_2[3][3] = {{9,8,7},

        {6,5,4},

{3,2,1}};

int sum=0;

//matrix summation

for(int i=0; i<3; ++i){

for(int j=0; j<3; ++j){                            sum = matrix_1[i][j]+matrix_2[i]     }}

                                                               //display the result

                                                     cout<<"Rsesultant matrix after sum is
equal : "<<endl;

                                                   for(int i=0; i<3; ++i){

                                                     for(int j=0; j<3; ++j){

                                                          cout<<sum<< " ";}

                                                     cout<<endl;}

                                                   return 0;}
```

Q.No.5

```
#include<iostream>

using namespace std;

void printTable(int n, int i=1){

                                                            if(i<=10){

                                                               cout<< n <<" x "<< i << " = " << n * i
<<endl;

                                                               printTable(n, i+1);}        }

int main()

{

                                                            cout<<" The multiplication table of 15 is
equalis equal to : "<<endl;

                                                            printTable(15);

                                                            return 0;

}
```

Q.No.01 home task

```
#include <iostream>

#include <cmath>


using namespace std;


// Function to calculate the determinant of a 2x2 matrix

float determinant2x2(float a, float b, float c, float d) {

   return a * d - b * c;

}
```

```
// Function to calculate the determinant of a 3x3 matrix
float determinant3x3(float matrix[3][3]) {
    float det = 0;
    for (int i = 0; i < 3; ++i) {
        det += matrix[0][i] * determinant2x2(matrix[1][(i + 1) % 3], matrix[2][(i + 2) % 3],
                                             matrix[1][(i + 2) % 3], matrix[2][(i + 1) % 3]);
    }
    return det;
}

// Function to calculate the cofactor of a 3x3 matrix
void cofactor(float matrix[3][3], float cofactorMatrix[3][3]) {
    for (int i = 0; i < 3; ++i) {
        for (int j = 0; j < 3; ++j) {
            cofactorMatrix[i][j] = pow(-1, i + j) * determinant2x2(matrix[(i + 1) % 3][(j + 1) % 3],
                                                                   matrix[(i + 1) % 3][(j + 2) % 3],
                                                                   matrix[(i + 2) % 3][(j + 1) % 3],
                                                                   matrix[(i + 2) % 3][(j + 2) % 3]);
        }
    }
}

// Function to transpose a matrix
void transpose(float matrix[3][3], float transposeMatrix[3][3]) {
    for (int i = 0; i < 3; ++i) {
        for (int j = 0; j < 3; ++j) {
            transposeMatrix[i][j] = matrix[j][i];
        }
    }
}

// Function to find the inverse of a 3x3 matrix
void inverse(float matrix[3][3], float inverseMatrix[3][3]) {
    float det = determinant3x3(matrix);

    if (det == 0) {
        cout << "The matrix is singular and does not have an inverse." << endl;
        return;
    }

    float cofactorMatrix[3][3];
    cofactor(matrix, cofactorMatrix);

    float adjointMatrix[3][3];
    transpose(cofactorMatrix, adjointMatrix);

    for (int i = 0; i < 3; ++i) {
        for (int j = 0; j < 3; ++j) {
            inverseMatrix[i][j] = adjointMatrix[i][j] / det;
        }
    }
}

// Function to display a 3x3 matrix
void displayMatrix(float matrix[3][3]) {
    for (int i = 0; i < 3; ++i) {
        for (int j = 0; j < 3; ++j) {
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }
}

int main() {
    float matrix[3][3];

    cout << "Enter the elements of the 3x3 matrix:" << endl;
    for (int i = 0; i < 3; ++i) {
        for (int j = 0; j < 3; ++j) {
            cin >> matrix[i][j];
        }
    }

    float inverseMatrix[3][3];
    inverse(matrix, inverseMatrix);

    cout << "Inverse Matrix:" << endl;
    displayMatrix(inverseMatrix);

    return 0;
}
```