LAB MANUAL 9

NAME HAMDAN HAFEEZ MALIK
ROLL. 480469

LAB TASK 1

```
#include <iostream>
using namespace std;
const int N = 3;
int main() {
int matrix[N][N];
for (int i = 0; i < N; i++) {
for (int j = 0; j < N; j++) {
cout << "Enter element (" << i + 1 << "," << j + 1 << "): ";
cin >> matrix[i][j];
 }
 }
int leftDiagonalSum = 0;
for (int i = 0; i < N; i++) {
leftDiagonalSum += matrix[i][i];
 }
for (int i = 0; i < N; i++) {
rightDiagonalSum += matrix[i][N - 1 - i];
 }
cout << "Left diagonal sum: " << leftDiagonalSum << endl;
cout << "Right diagonal sum: " << rightDiagonalSum << endl;
return 0
}
```

Lab task 2

```
#include <iostream>
using namespace std;
void addMatrices(int matrix1[3][3], int matrix2[3][3], int result[3][3]) {
 for (int i = 0; i < 3; i++) {
 for (int j = 0; j < 3; j++) {
 result[i][j] = matrix1[i][j] + matrix2[i][j];
 }
 }
}
int main() {
 int matrix1[3][3] = { {1, 2, 3},
 {4, 5, 6},
 {7, 8, 9} };
 int matrix2[3][3] = { {9, 8, 7},
 {6, 5, 4},
 {3, 2, 1} };
 int result[3][3];
 addMatrices(matrix1, matrix2, result);
 cout << "Resultant Matrix after addition:" << endl;
 for (int i = 0; i < 3; i++) {
 for (int j = 0; j < 3; j++) {
 cout << result[i][j] << " ";
 }
cout << endl;
 }
 return 0;
}
```

Lab task 3

```
#include <iostream>
using namespace std;
int transpose(int a[3][3])
{
int result[3][3];
for (int i = 0; i < 3;i++)
{
for (int j = 0; j < 3; j++)
{
result[j][i] = a[i][j];
}
}
cout << "While the transpose of the matrix is: " << endl;
for (int i = 0; i < 3;i++)
{
for (int j = 0; j < 3; j++)
{
cout << result[i][j]<<" ";
}
cout << "\n";
}
return 0;
}
int main()
{
int mat[3][3];
for (int i = 0; i < 3;i++)
{
cout << "Enter data of row " << i + 1 << endl;
for (int j = 0; j < 3; j++)
{
cin >> mat[i][j];
```

```
}
}
cout << "The resultant matrix is " << endl;
for (int i = 0; i < 3;i++)
{
for (int j = 0; j < 3; j++)
{
cout<< mat[i][j]<<" ";
}
cout << endl;
}
cout << endl;
transpose(mat);
return 0;
}
```

Lab task 4

Using 2D arrays in C++, implement 3x3 matrix multiplication. Make a function.

Code:

```
#include <iostream>
using namespace std;
void multiplyMatrices(int matrix1[3][3], int matrix2[3][3], int
result[3][3]) {
 for (int i = 0; i < 3; i++) {
```

```
 for (int j = 0; j < 3; j++) {
 result[i][j] = 0;
 for (int k = 0; k < 3; k++) {
 result[i][j] += matrix1[i][k] * matrix2[k][j];
}
 }
 }
}
int main() {
 int matrix1[3][3] = { {1, 2, 3},
 {4, 5, 6},
 {7, 8, 9} };
 int matrix2[3][3] = { {9, 8, 7},
 {6, 5, 4},
 {3, 2, 1} };
 int result[3][3];
 multiplyMatrices(matrix1, matrix2, result);
 cout << "Resultant Matrix after multiplication:" << endl;
 for (int i = 0; i < 3; i++) {
 for (int j = 0; j < 3; j++) {
 cout << result[i][j] << " ";
 }
 cout << endl;
 }
 return 0;
}
```

Lab task 5

```
#include <iostream>
using namespace std;
int table(int num, int i)
```

```
{
if (i <= 10)
{
cout << num << "*" << i << "=" << num * i << endl;
return table(num, i + 1);
}
else
{
return 1;
}
}
int main()
{
table(15, 1);
return 0;
}
```

HOME TASK

No 1

```
#include <iostream>
using namespace std;
int Determinant(int a[2][2])
{
return (a[0][0] * a[1][1]) - (a[0][1] * a[1][0]);
}
int Adjoint(int b[2][2])
{
int temp = 0;
temp = b[0][0];
b[0][0] = b[1][1];
```

```
b[1][1] = temp;
temp = -b[0][1];
b[0][1] = -b[1][0];
b[1][0] = temp;
return 0;
}
int Inverse(int a[2][2])
{
cout << "The inverse is :" << endl;
cout << 1 << "/" << Determinant(a) << " multiplied by the matrix :"
<< endl;
for (int i = 0; i < 2;i++)
{
for (int j = 0; j < 2; j++)
{
cout << a[i][j] << " ";
}
cout << endl;
}
return 0;
}
int InputMatrix(int b[2][2])
{
for (int i = 0; i < 2; i++)
{
cout << "Enter data for row " << i + 1 << endl;
for (int j = 0; j < 2; j++)
{
cin >> b[i][j];
}
}
return 0;
}
int main()
{
```

```
int mat[2][2];
InputMatrix(mat);
cout << "The input matrix :" << endl;
for (int i = 0; i < 2;i++) {
for (int j = 0; j < 2; j++) {
cout << mat[i][j] << " "; }
cout << endl; }
Determinant(mat);
Adjoint(mat);
Inverse(mat);
return 0;
}}
```