



Program:

Course Number	1444
Course Title	MATHEMATICS FOR EMBEDDED SYSTEMS
Semester/Year	SUMMER/2019

Instructor	Mohsen Salahi
------------	----------------------

Lab No.	3
----------------	----------

Student Name	CHANDRA SHEKARA REDDY SAGA	SRUTHY KRISHNAN	STEPHY BABY
Student ID	C0747253	C0749122	C0753812
Signature*	CSRS	SK	SB

**By signing above you attest that you have contributed to this submission and confirm that all work you have contributed to this submission is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, an "F" in the course, or possibly more severe penalties.*

Write a program in C++ to solve any set of linear equations using “LU Decomposition” (considering any useful pivoting).

PROGRAM

```
#include <iostream>
#include<math.h>
using namespace std;
int main()
{
    int i, j, k, n;
    int matrix[3][3]= { { 1,2,4},{3,8,14},{2,6,13} } ; ;
    int A[20][20], factor, C[10], x = 0.0;
    cout << "Order of matrix: " << endl;
    cin >> n;
    cout << "Elements of matrix in row wise:" << endl;
    for (i = 1; i <= n; i++)
    {
        for (j = 1; j <= (n + 1); j++)
        {

            cin >> A[i][j];

        }
    }

    cout << "Augmented matrix:" << endl;
    for (i = 1; i <= n; i++)
    {
        for (j = 1; j <= (n + 1); j++)
        {
            if (j == 4)
                cout << ":" << " " << A[i][j] << endl;
            else
                cout << A[i][j] << " ";
        }
    }
}
```

```

        }
    }
    for (j = 1; j <= n; j++)
    {
        for (i = 1; i <= n; i++)
        {
            if (i > j)
            {
                factor = A[i][j] / A[j][j];
                for (k = 1; k <= n + 1; k++)
                {
                    A[i][k] = A[i][k] - factor * A[j][k];
                }
            }
        }
    }

    C[n] = A[n][n + 1] / A[n][n];

    for (i = n - 1; i >= 1; i--)
    {
        x = 0;
        for (j = i + 1; j <= n; j++)
        {
            x = x + A[i][j] * C[j];
        }
        C[i] = (A[i][n + 1] - x) / A[i][i];
    }

    cout << endl;
    cout << "The solution is : " << endl;
    for (i = 1; i <= n; i++)
    {

        cout << "x" << i << "=" << C[i] << endl;
    }

```

```

    }

n = 3;

int low[3][3], up[3][3];
for (int i = 0; i < n; i++)
{
    for (int l = i; l < n; l++)
    {
int sum = 0;
        for (int j = 0; j < i; j++)
            sum += (low[i][j] * up[j][l]);
        up[i][l] = matrix[i][l] - sum;
    }

    for (int l = i; l < n; l++) {
        if (i == l)
            low[i][i] = 1;
        else
        {
            int sum = 0;
            for (int j = 0; j < i; j++)
                sum += (low[l][j] * up[j][i]);
            low[l][i] = (matrix[l][i] - sum) / up[i][i];
        }
    }
}

cout << "The lower matrix is :" << endl << endl;

for (int i = 0; i < n; i++)

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

```

```

        if ((i + j) <= 3 && i < j)
            cout << "0" << " ";
        else
            cout << low[i][j] << " ";
    }
    cout << endl;
}
cout << endl;
cout << "The upper matrix is:" << endl << endl;
for (int i = 0; i < n; i++)
{

    for (int j = 0; j < n; j++)
    {
        if ((i + j) <= 3 && i > j)
            cout << "0" << " ";
        else
            cout << up[i][j] << " ";

    }
    cout << endl;
}
}

```

OUTPUT

options	compilation	execution
---------	-------------	-----------

Order of matrix:

3

Elements of matrix in row wise:

1 2 4 3

3 8 14 13

2 6 13 4

Augmented matrix:

1 2 4 : 3

3 8 14 : 13

2 6 13 : 4

The solution is :

x1=3

x2=4

x3=-2

The lower matrix is :

1 0 0

3 1 0

2 1 1

The upper matrix is:

1 2 4

0 2 2

0 0 3

Exit code: 0 (normal program termination)