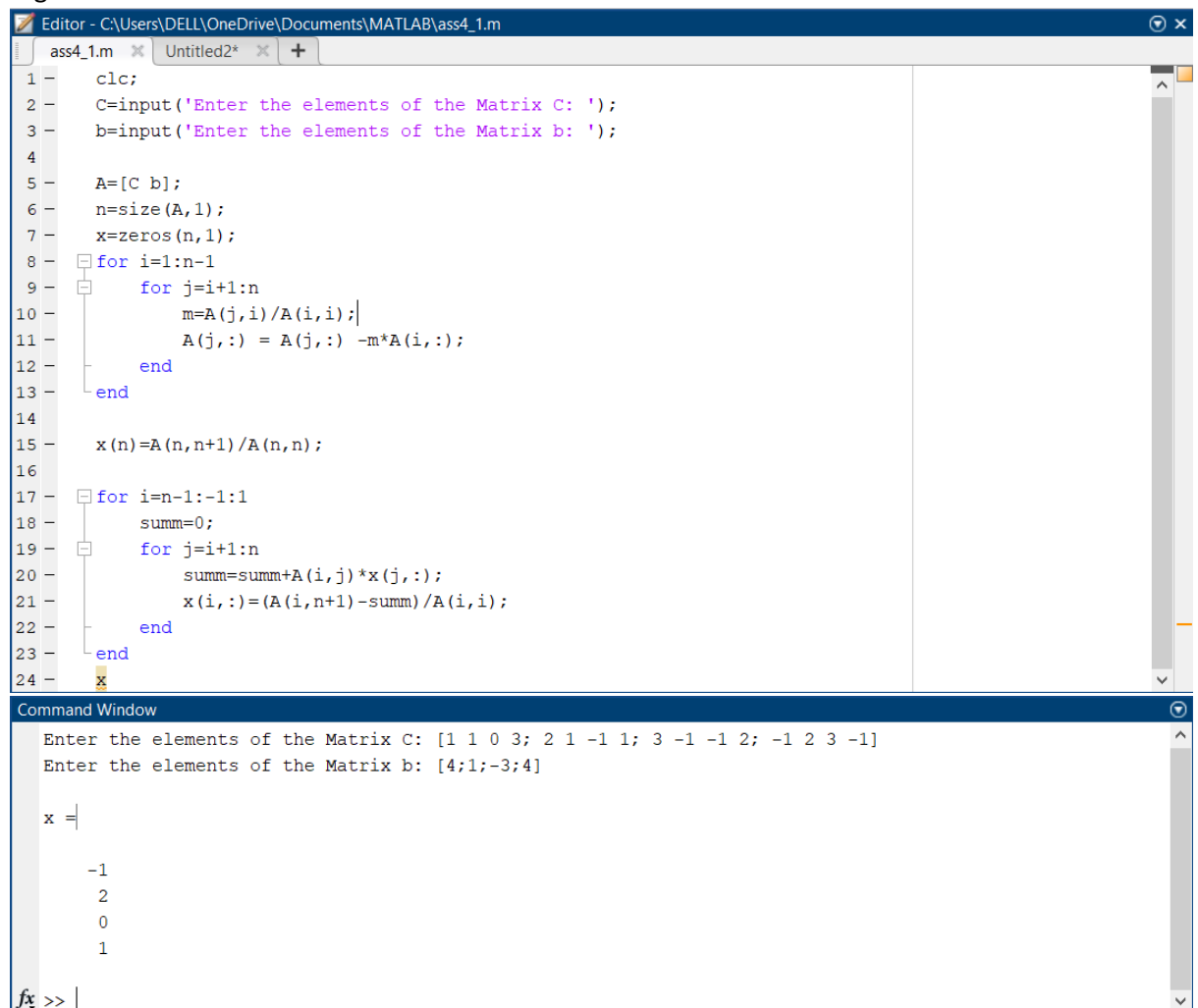


ASSIGNMENT – 4**Name – Meharamt Singh****Roll No. – 102003241****Group – 2CO10****Email Id – msingh10_be20@thapar.edu****1. Algorithm for Gauss elimination method:**

The image shows a MATLAB Editor window with a script named 'ass4_1.m' and a Command Window below it. The script implements the Gauss elimination method for solving a system of linear equations. The Command Window shows the user input for matrix C and vector b, followed by the calculated solution vector x.

```
Editor - C:\Users\DELL\OneDrive\Documents\MATLAB\ass4_1.m
ass4_1.m x Untitled2* x +
1 - clc;
2 - C=input('Enter the elements of the Matrix C: ');
3 - b=input('Enter the elements of the Matrix b: ');
4
5 - A=[C b];
6 - n=size(A,1);
7 - x=zeros(n,1);
8 - for i=1:n-1
9 -     for j=i+1:n
10 -         m=A(j,i)/A(i,i);
11 -         A(j,:)=A(j,:)-m*A(i,:);
12 -     end
13 - end
14
15 - x(n)=A(n,n+1)/A(n,n);
16
17 - for i=n-1:-1:1
18 -     summ=0;
19 -     for j=i+1:n
20 -         summ=summ+A(i,j)*x(j,:);
21 -         x(i,:)=(A(i,n+1)-summ)/A(i,i);
22 -     end
23 - end
24 - x

Command Window
Enter the elements of the Matrix C: [1 1 0 3; 2 1 -1 1; 3 -1 -1 2; -1 2 3 -1]
Enter the elements of the Matrix b: [4;1;-3;4]

x =
-1
2
0
1

fx >>
```

2. Algorithm for LU factorization method:

```

1  clc;
2  A=input('Enter the elements of the Matrix A: ');
3  b=input('Enter the elements of the Matrix b: ');
4  n=max(size(A));
5  m=eye(n);
6  for j=1:n-1
7      if A(j,j)==0
8          t=A(j,:);
9          A(j,:)=A(i,:);
10         A(i,:)=t;
11     end
12 end
13 for j=1:n-1
14     for i=j+1:n
15         if A(i,j)==0
16             fprintf('No unique solution exists');
17             break;
18         else
19             m(i,j)=A(i,j)/A(j,j);
20             A(i,:)=A(i,:)-m(i,j)*A(j,:);
21             b(i,:)=b(i,:)-m(i,j)*b(j,:);
22         end
23     end
24 end
25 U=A;
26 L=m;
27 x=zeros(n,1);
28 x(n,:)=b(n,:)/A(n,n);
29 for i=n-1:-1:1
30     x(i,:)=b(i,:)-A(i,i+1:n)*x(i+1:n,:)/A(i,i);
31 end
32

```

Command Window

Enter the elements of the Matrix A: [1 1 0 3;2 1 -1 1;3 -1 -1 2;-1 2 3 -1]

Enter the elements of the Matrix b: [4;1;-3;4]

U =

| | | | |
|---|----|----|-----|
| 1 | 1 | 0 | 3 |
| 0 | -1 | -1 | -5 |
| 0 | 0 | 3 | 13 |
| 0 | 0 | 0 | -13 |

L =

| | | | |
|----|----|---|---|
| 1 | 0 | 0 | 0 |
| 2 | 1 | 0 | 0 |
| 3 | 4 | 1 | 0 |
| -1 | -3 | 0 | 1 |

x =

| |
|----|
| -1 |
| 2 |
| 0 |
| 1 |

fx >>

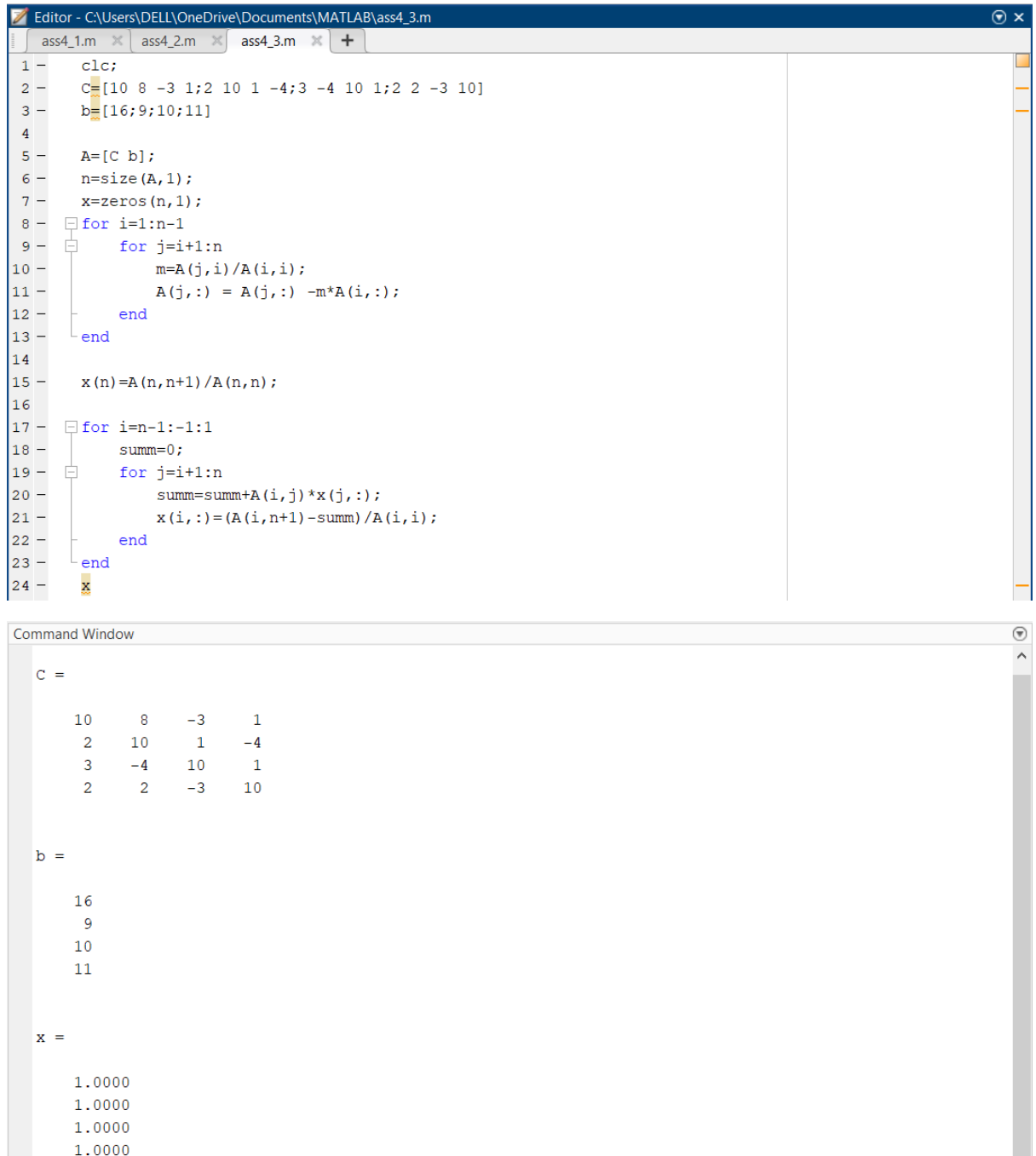
3. Use Gauss elimination method to find the solution of the following linear system of equations:

$$10x + 8y - 3z + u = 16$$

$$2x + 10y + z - 4u = 9$$

$$3x - 4y + 10z + u = 10$$

$$2x + 2y - 3z + 10u = 11$$



The image shows a MATLAB Editor window with a script named 'ass4_3.m' and a Command Window below it. The script implements the Gauss elimination method for a system of four linear equations. The matrix C is defined as [10 8 -3 1; 2 10 1 -4; 3 -4 10 1; 2 2 -3 10] and the vector b is [16; 9; 10; 11]. The script uses nested loops to perform row operations, resulting in the solution vector x = [1.0000; 1.0000; 1.0000; 1.0000].

```
1 - clc;
2 - C=[10 8 -3 1;2 10 1 -4;3 -4 10 1;2 2 -3 10]
3 - b=[16;9;10;11]
4 -
5 - A=[C b];
6 - n=size(A,1);
7 - x=zeros(n,1);
8 - for i=1:n-1
9 -     for j=i+1:n
10 -        m=A(j,i)/A(i,i);
11 -        A(j,:)=A(j,:)-m*A(i,:);
12 -     end
13 - end
14 -
15 - x(n)=A(n,n+1)/A(n,n);
16 -
17 - for i=n-1:-1:1
18 -     summ=0;
19 -     for j=i+1:n
20 -        summ=summ+A(i,j)*x(j,:);
21 -        x(i,:)=(A(i,n+1)-summ)/A(i,i);
22 -     end
23 - end
24 - x
```

Command Window

```
C =
    10     8    -3     1
     2    10     1    -4
     3    -4    10     1
     2     2    -3    10

b =
    16
     9
    10
    11

x =
    1.0000
    1.0000
    1.0000
    1.0000
```

4. Solve the following linear system of equations:

$$\pi x_1 + \sqrt{2}x_2 - x_3 + x_4 = 0$$

$$ex_1 - x_2 + x_3 + 2x_4 = 1$$

$$x_1 + x_2 - \sqrt{3}x_3 + x_4 = 2$$

$$-x_1 - x_2 + x_3 - \sqrt{5}x_4 = 3$$

```

1  clc;
2  pi=22/7; k=sqrt(2); l=sqrt(3); o=sqrt(5); e=2.718281828459045;
3  A=[pi k -1 1; e -1 1 2; 1 1 -1 1; -1 -1 1 -1];
4  b=[0;1;2;3];
5  n=max(size(A));
6  m=eye(n);
7  for j=1:n-1
8      if A(j,j)==0
9          t=A(j,:);
10         A(j,:)=A(i,:);
11         A(i,:)=t;
12     end
13 end
14 for j=1:n-1
15     for i=j+1:n
16         if A(j,j)==0
17             fprintf('No unique solution exists');
18             break;
19         else
20             m(i,j)=A(i,j)/A(j,j);
21             A(i,:)=A(i,:)-m(i,j)*A(j,:);
22             b(i,:)=b(i,:)-m(i,j)*b(j,:);
23         end
24     end
25 end
26 U=A;
27 L=m;
28 x=zeros(n,1);
29 x(n,:)=b(n,:)/A(n,n);
30 for i=n-1:-1:1
31     x(i,:)=(b(i,:)-A(i,i+1:n)*x(i+1:n,:))/A(i,i);
32 end
33 x

```

Command Window

A =

| | | | |
|---------|---------|---------|---------|
| 3.1429 | 1.4142 | -1.0000 | 1.0000 |
| 2.7183 | -1.0000 | 1.0000 | 2.0000 |
| 1.0000 | 1.0000 | -1.7321 | 1.0000 |
| -1.0000 | -1.0000 | 1.0000 | -2.2361 |

b =

| |
|---|
| 0 |
| 1 |
| 2 |
| 3 |

U =

| | | | |
|--------|---------|---------|---------|
| 3.1429 | 1.4142 | -1.0000 | 1.0000 |
| 0 | -2.2232 | 1.8649 | 1.1351 |
| 0 | 0 | -0.9525 | 0.9626 |
| 0 | 0 | 0 | -1.9759 |

L =

| | | | |
|---------|---------|---------|--------|
| 1.0000 | 0 | 0 | 0 |
| 0.8649 | 1.0000 | 0 | 0 |
| 0.3182 | -0.2474 | 1.0000 | 0 |
| -0.3182 | 0.2474 | -0.2314 | 1.0000 |

x =

| |
|---------|
| 1.3490 |
| -4.6790 |
| -4.0335 |
| -1.6563 |

5. Kirchhoff's laws of electrical circuits state that both the net flow of current through each junction and the net voltage drop around each closed loop of a circuit are zero. Suppose that a potential of V volts is applied between the points A and G in the circuit and that i_1 , i_2 , i_3 , i_4 and i_5 represent current flow as shown in the diagram. Using G as a reference point, Kirchhoff's laws imply that the currents satisfy the following system of linear equations:

$$5i_1 + 5i_2 = V$$

$$i_3 - i_4 - i_5 = 0$$

$$2i_4 - 3i_5 = 0$$

$$i_1 - i_2 - i_3 = 0$$

$$5i_2 - 7i_3 - 2i_4 = 0$$

```

4 A = [5 5 0 0 0; 0 0 1 -1 -1; 0 0 0 2 -3; 1 -1 -1 0 0; 0 5 -7 -2 0]
5 B = [V; 0; 0; 0; 0]
6 n = max(size(A));
7 m = eye(n);
8 for j=1:n-1
9     for i=j+1:n
10        if A(i,j)~=0
11            t=A(i,j);
12            A(i,:)=A(i,:)-A(j,:);
13            A(j,:)=t;
14        end
15    end
16 end
17 for j=1:n-1
18     for i=j+1:n
19        if A(i,j)~=0
20            fprintf('No unique solution exists');
21            break;
22        else
23            m(i,j)=A(i,j)/A(j,j);
24            A(i,:)=A(i,:)-m(i,j)*A(j,:);
25            B(i,:)=B(i,:)-m(i,j)*B(j,:);
26        end
27    end
28 end
29 U=A;
30 L=m;
31 x = zeros(n,1);
32 x(n,:)=B(n,:)/A(n,n);
33 for i=n-1:-1:1
34     x(i,:)=(B(i,:)-A(i,i+1:n)*x(i+1:n,:))/A(i,i);
35 end
36

```

Command Window

```

A =
     5     5     0     0     0
     0     0     1    -1    -1
     0     0     0     2    -3
     1    -1    -1     0     0
     0     5    -7    -2     0

B =

    5.0000
         0
         0
         0
         0

U =

    5.0000    5.0000         0         0         0
         0   -2.0000   -1.0000         0         0
         0         0    1.0000   -1.0000   -1.0000
         0         0         0    2.0000   -3.0000
         0         0         0         0  -26.7500

L =

    1.0000         0         0         0         0
    0.2000    1.0000         0         0         0
         0         0    1.0000         0         0
         0         0         0    1.0000         0
         0   -2.5000   -9.5000   -5.7500    1.0000

x =

    0.6785
    0.4215
    0.2570
    0.1542
    0.1028

```