**Shubhan Singh**

**2022300118**

**SE-Comps B/Batch C**

**22nd and 29th February 2024**

**Scilab no.4: Gauss Elimination Method**

**Program No.1** :- Write a sci lab code to solve the following equations in terms of x, y and z by using the Gauss Elimination Method.

Equations:

2x + y - z = 8

-3x - y + 2z = -11

-2x + y + 2z = -3

**Code :-**

clc();

A=[2 1 -1; -3 -1 2; -2 1 2];

printf("Matrix A : ");

disp(A);

B=[8; -11; -3];

printf("Matrix B : ");

disp(B);

C=[A B];

printf("Matrix C : ");

disp(C);

n=3;

for i=1:n

if C(i,i)~=0

C(i,:)=C(i,:)/C(i,i);

end

disp(C);

for j=i+1:n

C(j,:)=C(j,:)-C(j,i)\*C(i,:);

end

disp(C);

end

printf("\nMatrix C : ");

disp(C);

z=C(3,4);

y=C(2,4)-z\*C(2,3);

x=C(1,4)-z\*C(1,3)-y\*C(1,2);

printf("\nx = %g\n",x);

printf("y = %g\n",y);

printf("z = %g\n",z);

**Output :-**

**A screenshot of a math test

Description automatically generated**

**A number on a white background

Description automatically generated**

**Program No. 2:-** Write a sci lab code to solve the following equations in terms of x, y, z and w by using the Gauss Elimination Method.

10x - 2y - z - w = 3

-2x + 10y - z - w = 15

-x - y + 10z - 2w = 27

-x - y - 2z + 10w = -9

**Code**:-

clc();

A=[10 -2 -1 -1; -2 10 -1 -1; -1 -1 10 -2; -1 -1 -2 10];

printf("Matrix A : ");

disp(A);

B=[3;15;27;-9];

printf("Matrix B : ");

disp(B);

C=[A B];

printf("Matrix C : ");

disp(C);

n=4;

for i=1:n

if C(i,i)==0

C(i,:)=C(i,:)

else

C(i,:)=C(i,:)/C(i,i);

disp(C);

for j=1:n-1

if i+j<n+1

C(i+j,:)=C(i+j,:)-C(i+j,i)\*C(i,:);

else

end

end

end

end

printf("\nMatrix C : ");

disp(C);

w = C(4, 5);

z = C(3, 5) - w\*C(3, 4);

y = C(2, 5) - w\*C(2, 4) - z\*C(2, 3);

x = C(1, 5) - w\*C(1, 4) - z\*C(1, 3) - y\*C(1, 2);

printf("\nx = %g\n",x);

printf("y = %g\n",y);

printf("z = %g\n",z);

printf("w = %g\n",w);

**Output** :-

A white paper with black numbers

Description automatically generated

A screenshot of a computer

Description automatically generated

**Program No. 3 :-** Write a sci lab code to solve the following equations in terms of x, y and z by using the Gauss Elimination Method.

x - 2y + z = 0

2x + y - 3z = 5

4x - 7y + z = -1

**Code** :-

clc();

A=[1 -2 1;2 1 -3;4 -7 1];

printf("Matrix A : ");

disp(A);

B=[0;5;-1];

printf("Matrix B : ");

disp(B);

C=[A B];

printf("Matrix C : ");

disp(C);

n=3;

for i=1:n

if C(i,i)==0

C(i,:)=C(i,:)

else

C(i,:)=C(i,:)/C(i,i);

disp(C);

for j=1:n-1

if i+j<n+1

C(i+j,:)=C(i+j,:)-C(i+j,i)\*C(i,:);

else

end

end

end

end

printf("Matrix C : ");

disp(C);

z=C(3,4);

y=C(2,4)-z\*C(2,3);

x=C(1,4)-z\*C(1,3)-y\*C(1,2);

printf("\n\nx = %g\n",x);

printf("y = %g\n",y);

printf("z = %g\n",z);

**Output** :-

A screenshot of a math test

Description automatically generated

**Program no. 4** :- Write a sci lab code to solve the following equations in terms of x, y and z by using the Gauss Elimination Method.

5x -2y + 3z = -1

-3x + 9y + z = 2

2x -y -7z = 3

**Code** :-

clc();

A=[5 -2 3;-3 9 1;2 -1 -7];

printf("Matrix A : ");

disp(A);

B=[-1;2;3];

printf("Matrix B : ");

disp(B);

C=[A B];

printf("Matrix C : ");

disp(C);

n=3;

for i=1:n

if C(i,i)==0

C(i,:)=C(i,:)

else

C(i,:)=C(i,:)/C(i,i);

disp(C);

for j=1:n-1

if i+j<n+1

C(i+j,:)=C(i+j,:)-C(i+j,i)\*C(i,:);

else

end

end

end

end

printf("Matrix C : ");

disp(C);

z=C(3,4);

y=C(2,4)-z\*C(2,3);

x=C(1,4)-z\*C(1,3)-y\*C(1,2);

printf("\n\nx = %g\n",x);

printf("y = %g\n",y);

printf("z = %g\n",z);

**Output** :-

A screenshot of a computer

Description automatically generated

**Program No. 5** :- Write a sci lab code to solve the following equations in terms of x, y and z by using the Gauss Elimination Method.

5x + 4y - z = 0

10y - 3z = 11

z = 3

**Code** :-

clc();

A=[5 4 -1;0 10 -3;0 0 1];

printf("Matrix A : ");

disp(A);

B=[0;11;3];

printf("Matrix B : ");

disp(B);

C=[A B];

printf("Matrix C : ");

disp(C);

n=3;

for i=1:n

if C(i,i)==0

C(i,:)=C(i,:)

else

C(i,:)=C(i,:)/C(i,i);

disp(C);

for j=1:n-1

if i+j<n+1

C(i+j,:)=C(i+j,:)-C(i+j,i)\*C(i,:);

else

end

end

end

end

printf("Matrix C : ");

disp(C);

z=C(3,4);

y=C(2,4)-z\*C(2,3);

x=C(1,4)-z\*C(1,3)-y\*C(1,2);

printf("\n\nx = %g\n",x);

printf("y = %g\n",y);

printf("z = %g\n",z);

**Output** :-

A screenshot of a computer

Description automatically generated

**Program no. 6** :- Write a sci lab code to solve the following equations in terms of x, y and z by using the Gauss Elimination Method.

-3x + 2y - 6z = 6

5x + 7y - 5z = 6

x + 4y -2z = 8

**Code** :-

clc();

A=[-3 2 -6;5 7 -5;1 4 -2];

printf("Matrix A : ");

disp(A);

B=[6;6;8];

printf("Matrix B : ");

disp(B);

C=[A B];

printf("Matrix C : ");

disp(C);

n=3;

for i=1:n

if C(i,i)==0

C(i,:)=C(i,:)

else

C(i,:)=C(i,:)/C(i,i);

disp(C);

for j=1:n-1

if i+j<n+1

C(i+j,:)=C(i+j,:)-C(i+j,i)\*C(i,:);

else

end

end

end

end

printf("Matrix C : ");

disp(C);

z=C(3,4);

y=C(2,4)-z\*C(2,3);

x=C(1,4)-z\*C(1,3)-y\*C(1,2);

printf("\n\nx = %g\n",x);

printf("y = %g\n",y);

printf("z = %g\n",z);

**Output** :-

A screenshot of a computer

Description automatically generated

**Program No. 7** :- Write a sci lab code to solve the following equations in terms of x, y and z by using the Gauss Elimination Method.

x + 2y + 3z = 6

2x - 3y + 2z = 14

3x + y -z = -2

**Code** :-

clc();

A=[1 2 3;2 -3 2;3 1 -1];

printf("Matrix A : ");

disp(A);

B=[6;14;-2];

printf("Matrix B : ");

disp(B);

C=[A B];

printf("Matrix C : ");

disp(C);

n=3;

for i=1:n

if C(i,i)==0

C(i,:)=C(i,:)

else

C(i,:)=C(i,:)/C(i,i);

disp(C);

for j=1:n-1

if i+j<n+1

C(i+j,:)=C(i+j,:)-C(i+j,i)\*C(i,:);

else

end

end

end

end

printf("Matrix C : ");

disp(C);

z=C(3,4);

y=C(2,4)-z\*C(2,3);

x=C(1,4)-z\*C(1,3)-y\*C(1,2);

printf("\n\nx = %g\n",x);

printf("y = %g\n",y);

printf("z = %g\n",z);

**Output** :-

A screenshot of a math test

Description automatically generated

**Program No. 8:** Write a sci lab code to solve the following equations in terms of x, y, z and w by using the Gauss Elimination Method.

A = B =

**Code:**

clc();

A=[2 1 0 1;5 -4 1 0;3 0 2 0;1 1 -1 1];

printf("Matrix A : ");

disp(A);

B=[2;1;-2;1];

printf("Matrix B : ");

disp(B);

C=[A B];

printf("Matrix C : ");

disp(C);

n=4;

for i=1:n

if C(i,i)==0

C(i,:)=C(i,:)

else

C(i,:)=C(i,:)/C(i,i);

disp(C);

for j=1:n-1

if i+j<n+1

C(i+j,:)=C(i+j,:)-C(i+j,i)\*C(i,:);

else

end

end

end

end

printf("Matrix C : ");

disp(C);

w = C(4, 5);

z = C(3, 5) - w\*C(3, 4);

y = C(2, 5) - w\*C(2, 4) - z\*C(2, 3);

x = C(1, 5) - w\*C(1, 4) - z\*C(1, 3) - y\*C(1, 2);

printf("\n\nx = %g\n",x);

printf("y = %g\n",y);

printf("z = %g\n",z);

printf("w = %g\n",w);

**Output:**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated