**Shubhan Singh**

**2022300118**

**SE-Comps B/Batch C**

**28th March 2024**

**Scilab no.7 : Gauss Jordan Method**

**Program No.1** :- Write a scilab code to solve the following equations in terms of x,y,z by using gauss jordan method

x+3y+2z = 2

2x+7y+7z = -1

2x+5y+2z = 7

**Code :-**

clc;

clear all;

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

printf("Matrix A : ");

disp(A)

printf("Matrix B : ")

B = [2 ; -1 ; 7]

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

for j = 1:n-1

if i+j < n+1

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

end

end

disp(C)

end

for j = n:-1:2

for i = 1 : j-1

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

end

disp(C);

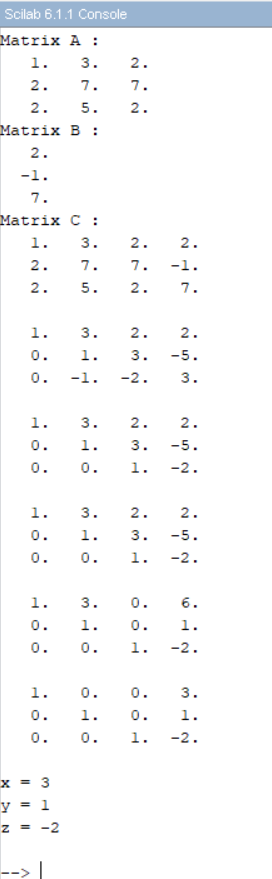
end

printf("\nx = %g\n",C(1,4));

printf("y = %g\n",C(2,4));

printf("z = %g\n",C(3,4));

**Output :-**

****

**Program No. 2:-** Write a scilab code to solve the following equations in terms of x,y,z,w by using gauss jordan method

8x+9y+2z+9w = 42

2x+7y+3z+5w = 45

4x+3y+6z+6w = 53

2x+5y+6z+8w = 63

**Code**:-

clc;

clear all;

A = [8 9 2 9 ; 2 7 3 5 ; 4 3 6 6 ; 2 5 6 8];

printf("Matrix A : ");

disp(A)

printf("Matrix B : ")

B = [42;45;53;63];

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 , :)/C(i , i);

end

for j = 1:n-1

if i+j < n+1

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

end

end

disp(C)

end

for j = n:-1:2

for i = 1 : j-1

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

end

disp(C);

end

printf("\nx = %g\n",C(1,5));

printf("y = %g\n",C(2,5));

printf("z = %g\n",C(3,5));

printf("w = %g\n",C(4,5));

**Output** :-

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**Program No. 3 :-** Write a scilab code to solve the following equations in terms of x,y,z by using gauss jordan method

15x-y = 97

3x-5y+z = 47

3x+5y+15z = 67

**Code** :-

clc;

clear all;

A = [15 -1 0 ; 3 -5 1 ; 3 5 15];

printf("Matrix A : ");

disp(A)

printf("Matrix B : ")

B = [97 ; 47 ; 67]

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

for j = 1:n-1

if i+j < n+1

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

end

end

disp(C)

end

for j = n:-1:2

for i = 1 : j-1

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

end

disp(C);

end

printf("\nx = %g\n",C(1,4));

printf("y = %g\n",C(2,4));

printf("z = %g\n",C(3,4));

**Output** :-

A screenshot of a computer

Description automatically generated

**Program no. 4** :- Write a scilab code to solve the following equations in terms of x,y,z by using gauss jordan method

x+2y+6z = 22

3x+4y+z = 26

6x-y-z = 19

**Code** :-

clc;

clear all;

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

printf("Matrix A : ");

disp(A)

printf("Matrix B : ")

B = [22; 26; 19]

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

for j = 1:n-1

if i+j < n+1

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

end

end

disp(C)

end

for j = n:-1:2

for i = 1 : j-1

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

end

disp(C);

end

printf("\nx = %g\n",C(1,4));

printf("y = %g\n",C(2,4));

printf("z = %g\n",C(3,4));

**Output** :-

A screenshot of a math test

Description automatically generated

**Program No. 5** :- Write a scilab code to solve the following equations in terms of x,y,z by using gauss jordan method

x+2y+6z = 44

3x+4y+z = 52

6x-y-z = 38

**Code** :-

clc;

clear all;

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

printf("Matrix A : ");

disp(A)

printf("Matrix B : ")

B = [44; 52; 38]

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

for j = 1:n-1

if i+j < n+1

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

end

end

disp(C)

end

for j = n:-1:2

for i = 1 : j-1

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

end

disp(C);

end

printf("\nx = %g\n",C(1,4));

printf("y = %g\n",C(2,4));

printf("z = %g\n",C(3,4));

**Output** :-

A screenshot of a math test

Description automatically generated

**Program No. 6** :- Write a scilab code to solve the following equations in terms of x,y,z by using gauss jordan method

2x+y-z+3w = 11

x-2y+z+w = 8

4x+7y+2z-w = 0

3x+5y+4z+4w = 17

**Code** :-

clc;

clear all;

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

printf("Matrix A : ");

disp(A)

printf("Matrix B : ")

B = [11; 8; 0; 17]

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 , :)/C(i , i);

end

for j = 1:n-1

if i+j < n+1

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

end

end

disp(C)

end

for j = n:-1:2

for i = 1 : j-1

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

end

disp(C);

end

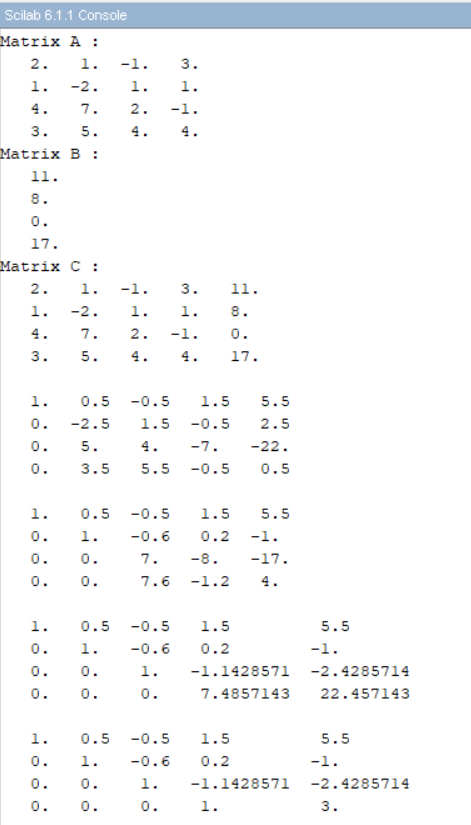
printf("\nx = %g\n",C(1,5));

printf("y = %g\n",C(2,5));

printf("z = %g\n",C(3,5));

printf("w = %g\n",C(4,5));

**Output** :-



A screenshot of a math test

Description automatically generated

**Program No. 7** :- Write a scilab code to solve the following equations in terms of x,y,z,w by using gauss jordan method

2x+4y-z = 3

5x-7y+2z = 5

7x+8y-13z = 17

**Code** :-

clc;

clear all;

A = [2 4 -1 ; 5 -7 2 ; 7 8 -13];

printf("Matrix A : ");

disp(A)

printf("Matrix B : ")

B = [3 ; 5 ; 17]

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

for j = 1:n-1

if i+j < n+1

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

end

end

disp(C)

end

for j = n:-1:2

for i = 1 : j-1

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

end

disp(C);

end

printf("\nx = %g\n",C(1,4));

printf("y = %g\n",C(2,4));

printf("z = %g\n",C(3,4));

**Output** :-

A screenshot of a computer

Description automatically generated