Name: Amal S. Thundiyil
UID: 2020400066
Batch: IT-D

Scilab No. : 6

Title: **Gauss Jordan Method**

Program 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]
disp(A)
B = [2;-1;7]
disp(B)
C = [A B]
n = 3
for i = 1:n
   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, :)
      end
   end
end

for j = n:-1:2
   for i = 1:j-1
      C(i, :) = C(i, :) - C(i, j)*C(j, :)
   end
   disp(C)
end

disp("Z=")
disp(C(3, 4))
disp("y=")
disp(C(2, 4))
disp("X=")
disp(C(1, 4))
```

Output:

```
  2.
 -1.
  7.

  1.   3.   2.   2.
  2.   7.   7.  -1.
  2.   5.   2.   7.

  1.   3.   2.   2.
  0.   1.   3.  -5.
  0.  -1.  -2.   3.

  1.   3.   2.   2.
  0.   1.   3.  -5.
  0.   0.   1.  -2.

  1.   3.   0.   6.
  0.   1.   0.   1.
  0.   0.   1.  -2.

  1.   0.   0.   3.
  0.   1.   0.   1.
  0.   0.   1.  -2.

Z=

 -2.

y=

  1.

X=

  3.
```

```
Z=

 -2.

y=

  1.

X=
```

Program 2: Write a scilab code to solve the following set of equations in terms of x, y, z and 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];
disp(a);
b = [42; 45; 53; 63];
disp(b);
c = [a b];
disp(c);
n = 4;

for i = 1 : n
      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, :)
         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("x : %.f\n", c(1, 5));
printf("y : %.f\n", c(2, 5));
printf("z : %.f\n", c(3, 5));
printf("w : %.f\n", c(4, 5));
```

Output:

```
8.   9.   2.   9.
2.   7.   3.   5.
4.   3.   6.   6.
2.   5.   6.   8.

42.
45.
53.
63.

8.   9.   2.   9.   42.
2.   7.   3.   5.   45.
4.   3.   6.   6.   53.
2.   5.   6.   8.   63.

1.   1.125   0.25   1.125   5.25
2.   7.      3.     5.      45.
4.   3.      6.     6.      53.
2.   5.      6.     8.      63.

1.   1.125   0.25   1.125   5.25
0.   4.75    2.5    2.75    34.5
0.  -1.5     5.     1.5     32.
0.   2.75    5.5    5.75    52.5

      column 1 to 4

1.   1.125   0.25        1.125
0.   1.      0.5263158   0.5789474
0.  -1.5     5.          1.5
0.   2.75    5.5         5.75

      column 1 to 4

1.   1.125   0.25        1.125
0.   1.      0.5263158   0.5789474
0.   0.      1.          0.4090909
0.   0.      4.0526316   4.1578947

      column 5

5.25
7.2631579
7.4090909
32.526316

      column 1 to 4

1.   1.125   0.25        1.125
0.   1.      0.5263158   0.5789474
0.   0.      1.          0.4090909
0.   0.      0.          2.5

      column 5

5.25
7.2631579
7.4090909
2.5

      column 5

5.25
7.2631579
32.
52.5

      column 1 to 4

1.   1.125   0.25        1.125
0.   1.      0.5263158   0.5789474
0.   0.      5.7894737   2.3684211
0.   0.      4.0526316   4.1578947

      column 5

5.25
7.2631579
42.894737
32.526316

      column 1 to 4

1.   1.125   0.25        1.125
0.   1.      0.5263158   0.5789474
0.   0.      1.          0.4090909
0.   0.      0.          1.

      column 5

5.25
7.2631579
7.4090909
1.

      column 1 to 4

1.   1.125   0.25        1.125
0.   1.      0.5263158   0.5789474
0.   0.      1.          0.4090909
0.   0.      0.          1.

      column 5

5.25
7.2631579
7.4090909
1.

      column 5

5.25
7.2631579
7.4090909
1.

1.   1.125   0.25        0.   4.125
0.   1.      0.5263158   0.   6.6842105
0.   0.      1.          0.   7.
0.   0.      0.          1.   1.

1.   1.125   0.   0.   2.375
0.   1.      0.   0.   3.
0.   0.      1.   0.   7.
0.   0.      0.   1.   1.

1.   0.   0.   0.  -1.
0.   1.   0.   0.   3.
0.   0.   1.   0.   7.
0.   0.   0.   1.   1.
x : -1
y : 3
z : 7
w : 1
```

Program 3: 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]
disp(A)
B = [44;52;38]
disp(B)
C = [A B]
n = 3
for i = 1:n
   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, :)
      end
   end
end

for j = n:-1:2
   for i = 1:j-1
      C(i, :) = C(i, :) - C(i, j)*C(j, :)
   end
   disp(C)
end

disp("Z=")
disp(C(3, 4))
disp("y=")
disp(C(2, 4))
disp("X=")
disp(C(1, 4))
```

Output:

```
   1.   2.   6.
   3.   4.   1.
   6.  -1.  -1.

   44.
   52.
   38.

   1.   2.   6.   44.
   3.   4.   1.   52.
   6.  -1.  -1.   38.

   1.   2.    6.    44.
   0.   1.    8.5   40.
   0.  -13.  -37.  -226.

   1.   2.   6.    44.
   0.   1.   8.5   40.
   0.   0.   1.    4.

   1.   2.   0.   20.
   0.   1.   0.   6.
   0.   0.   1.   4.

   1.   0.   0.   8.
   0.   1.   0.   6.
   0.   0.   1.   4.

Z=

   4.
```

```
   0.   0.   1.   4.

   1.   0.   0.   8.
   0.   1.   0.   6.
   0.   0.   1.   4.

Z=

   4.

y=

   6.

X=

   8.
```

Program 4: Write a scilab code to solve the following equations in terms of a, b, c, d by using gauss jordan method

$$2a + b + c + 3d = 8,\ a + b + c + d = -2,\ 3a + 2b - c = 6,\ 4b + 3c + 2d = -8$$

Code:

```
clc;
clear all;

a = [ 2 1 -1 3;1 1 1 -1; 3 2 -1 0; 0 4 3 2];
disp(a);
b = [8;-2;6;-8];
disp(b);
c = [a b];
disp(c);
n = 4;

for i = 1 : n
      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, :)
         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("a : %.f\n", c(1, 5));
printf("b : %.f\n", c(2, 5));
printf("c : %.f\n", c(3, 5));
printf("d : %.f\n", c(4, 5));
```

Output:

```
 2.   1.  -1.   3.
 1.   1.   1.  -1.
 3.   2.  -1.   0.
 0.   4.   3.   2.

 8.
-2.
 6.
-8.

 2.   1.  -1.   3.   8.
 1.   1.   1.  -1.  -2.
 3.   2.  -1.   0.   6.
 0.   4.   3.   2.  -8.

 1.   0.5  -0.5   1.5   4.
 1.   1.    1.   -1.   -2.
 3.   2.   -1.    0.    6.
 0.   4.    3.    2.   -8.

 1.   0.5  -0.5   1.5   4.
 0.   0.5   1.5  -2.5  -6.
 0.   0.5   0.5  -4.5  -6.
 0.   4.    3.    2.   -8.

 1.   0.5  -0.5   1.5   4.
 0.   1.    3.   -5.   -12.
 0.   0.5   0.5  -4.5  -6.
 0.   4.    3.    2.   -8.

 1.   0.5  -0.5   1.5   4.
 0.   1.    3.   -5.   -12.
 0.   0.   -1.   -2.    0.
 0.   0.   -9.    22.   40.

 1.   0.5  -0.5   1.5   4.
 0.   1.    3.   -5.   -12.
 0.   0.    1.    2.    0.
 0.   0.   -9.    22.   40.

 1.   0.5  -0.5   1.5   4.
 0.   1.    3.   -5.   -12.
 0.   0.    1.    2.    0.
 0.   0.    0.    40.   40.

 1.   0.5  -0.5   1.5   4.
 0.   1.    3.   -5.   -12.
 0.   0.    1.    2.    0.
 0.   0.    0.    1.    1.

 1.   0.5  -0.5   1.5   4.
 0.   1.    3.   -5.   -12.
 0.   0.    1.    2.    0.
 0.   0.    0.    1.    1.

 1.   0.5  -0.5   0.   2.5
 0.   1.    3.    0.  -7.
 0.   0.    1.    0.  -2.
 0.   0.    0.    1.   1.

 1.   0.5   0.   0.   1.5
 0.   1.    0.   0.  -1.
 0.   0.    1.   0.  -2.
 0.   0.    0.   1.   1.

 1.   0.   0.   0.   2.
```

```
   1.    0.5   0.    0.    1.5
   0.    1.    0.    0.   -1.
   0.    0.    1.    0.   -2.
   0.    0.    0.    1.    1.

   1.    0.    0.    0.    2.
   0.    1.    0.    0.   -1.
   0.    0.    1.    0.   -2.
   0.    0.    0.    1.    1.

a : 2
b : -1
c : -2
d : 1
```