Name - Varin Mehta
UID - 2022300058
Class - SE Computer A (Batch D)
Date - 27-02-24

Scilab 5 - Gauss Jacobi method

Program 1
Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Jacobi method. Perform 7 iterations. Equations are:
5x - 2y + 3z = -1
-3x + 9y + z = 2
2x - y - 7z = 3

Code -

```
clc;
clear all
A = [5 -2 3; -3 9 1; 2 -1 -7]
B = [-1; 2; 3]
n = 7;
x = 0;
y = 0;
z = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*y-A(1,3)*z)/A(1,1)
   Y = (B(2)- A(2,1)*x-A(2,3)*z)/A(2,2)
   Z = (B(3)- A(3,1)*x-A(3,2)*y)/A(3,3)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
   x = X
   y = Y
   z = Z
end
```

Output -

```
Iteration number 1
Value of x = -0.2
Value of y = 0.222222
Value of z = -0.428571
Iteration number 2
Value of x = 0.146032
Value of y = 0.203175
Value of z = -0.51746
Iteration number 3
Value of x = 0.191746
Value of y = 0.328395
Value of z = -0.415873
Iteration number 4
Value of x = 0.180882
Value of y = 0.332346
Value of z = -0.4207
Iteration number 5
Value of x = 0.185359
Value of y = 0.329261
Value of z = -0.424369
Iteration number 6
Value of x = 0.186326
Value of y = 0.33116
Value of z = -0.422649
Iteration number 7
Value of x = 0.186054
Value of y = 0.331292
Value of z = -0.422644
-->
```

Program 2

Write a Scilab code to solve the following set of equations in terms of x, y, z and w by using Gauss Jacobi method. Perform 10 iterations Equations are:

10x - 2y - z - w = 3

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

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

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

Code -

```
clc;
clear all
A = [10 -2 -1 -1; -2 10 -1 -1; -1 -1 10 -2; -1 -1 -2 10]
B = [3; 15; 27; -9]
n = 10;
x = 0;
y = 0;
z = 0;
w = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*y-A(1,3)*z-A(1,4)*w)/A(1,1)
   Y = (B(2)- A(2,1)*x-A(2,3)*z-A(2,4)*w)/A(2,2)
   Z = (B(3)- A(3,1)*x-A(3,2)*y-A(3,4)*w)/A(3,3)
   W = (B(4)- A(4,1)*x-A(4,2)*y-A(4,3)*z)/A(4,4)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
   printf("\nValue of w = %g", W)
   x = X
   y = Y
   z = Z
   w = W
end
```

Output -

```
Iteration number 1
Value of x = 0.3
Value of y = 1.5
Value of z = 2.7
Value of w = -0.9
Iteration number 2
Value of x = 0.78
Value of y = 1.74
Value of z = 2.7
Value of w = -0.18
Iteration number 3
Value of x = 0.9
Value of y = 1.908
Value of z = 2.916
Value of w = -0.108
Iteration number 4
Value of x = 0.9624
Value of y = 1.9608
Value of z = 2.9592
Value of w = -0.036
Iteration number 5
Value of x = 0.98448
Value of y = 1.9848
Value of z = 2.98512
Value of w = -0.01584
Iteration number 6
Value of x = 0.993888
Value of y = 1.99382
Value of z = 2.99376
Value of w = -0.006048
Iteration number 7
Value of x = 0.997536
Value of y = 1.99755
Value of z = 2.99756
Value of w = -0.0024768
```

```
Iteration number 8
Value of x = 0.999018
Value of y = 1.99902
Value of z = 2.99901
Value of w = -0.0009792
Iteration number 9
Value of x = 0.999607
Value of y = 1.99961
Value of z = 2.99961
Value of w = -0.000393984
Iteration number 10
Value of x = 0.999843
Value of y = 1.99984
Value of z = 2.99984
Value of w = -0.000157133
```

Program 3

Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Jacobi method. Perform 7 iterations. Equations are:

$10x + y + z = 12$

$x + 10y + z = 12$

$x + y + 10z = 12$

Code -

```
clc;
clear all
A = [10 1 1; 1 10 1; 1 1 10]
B = [12; 12; 12]
n = 7;
x = 0;
y = 0;
z = 0;
for i = 1:n
    printf("\nIteration number %g", i)
    X = (B(1)- A(1,2)*y-A(1,3)*z)/A(1,1)
    Y = (B(2)- A(2,1)*x-A(2,3)*z)/A(2,2)
    Z = (B(3)- A(3,1)*x-A(3,2)*y)/A(3,3)
    printf("\nValue of x = %g", X)
    printf("\nValue of y = %g", Y)
    printf("\nValue of z = %g", Z)
    x = X
    y = Y
    z = Z
end
```

Output -

```
Iteration number 1
Value of x = 1.2
Value of y = 1.2
Value of z = 1.2
Iteration number 2
Value of x = 0.96
Value of y = 0.96
Value of z = 0.96
Iteration number 3
Value of x = 1.008
Value of y = 1.008
Value of z = 1.008
Iteration number 4
Value of x = 0.9984
Value of y = 0.9984
Value of z = 0.9984
Iteration number 5
Value of x = 1.00032
Value of y = 1.00032
Value of z = 1.00032
Iteration number 6
Value of x = 0.999936
Value of y = 0.999936
Value of z = 0.999936
Iteration number 7
Value of x = 1.00001
Value of y = 1.00001
Value of z = 1.00001
```

Program 4

Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Jacobi method. Perform 10 iterations. Equations are:

$15x + 2y + z = 18$

$2x + 20y - 3z = 19$

$3x - 6y + 25z = 22$

Code -

```
clc;
clear all
A = [15 2 1; 2 20 -3; 3 -6 25]
B = [18; 19; 22]
n = 10;
x = 0;
y = 0;
z = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*y-A(1,3)*z)/A(1,1)
   Y = (B(2)- A(2,1)*x-A(2,3)*z)/A(2,2)
   Z = (B(3)- A(3,1)*x-A(3,2)*y)/A(3,3)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
   x = X
   y = Y
   z = Z
end
```

Output -

```
Iteration number 1
Value of x = 1.2
Value of y = 0.95
Value of z = 0.88
Iteration number 2
Value of x = 1.01467
Value of y = 0.962
Value of z = 0.964
Iteration number 3
Value of x = 1.00747
Value of y = 0.993133
Value of z = 0.98912
Iteration number 4
Value of x = 1.00164
Value of y = 0.997621
Value of z = 0.997456
Iteration number 5
Value of x = 1.00049
Value of y = 0.999454
Value of z = 0.999232
Iteration number 6
Value of x = 1.00012
Value of y = 0.999836
Value of z = 0.999811
Iteration number 7
Value of x = 1.00003
Value of y = 0.999959
Value of z = 0.999946
Iteration number 8
Value of x = 1.00001
Value of y = 0.999988
Value of z = 0.999986
Iteration number 9
Value of x = 1
Value of y = 0.999997
Value of z = 0.999996
Iteration number 10
Value of x = 1
Value of y = 0.999999
Value of z = 0.999999
```

Program 5

Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Jacobi method. Perform 10 iterations. Equations are:

$20x + y - 2z = 17$

$3x + 20y - z = -18$

$2x - 3y + 20z = 25$

Code -

```
clc;
clear all
A = [20 1 -2; 3 20 -1; 2 -3 20]
B = [17; -18; 25]
n = 10;
x = 0;
y = 0;
z = 0;
for i = 1:n
    printf("\nIteration number %g", i)
    X = (B(1)- A(1,2)*y-A(1,3)*z)/A(1,1)
    Y = (B(2)- A(2,1)*x-A(2,3)*z)/A(2,2)
    Z = (B(3)- A(3,1)*x-A(3,2)*y)/A(3,3)
    printf("\nValue of x = %g", X)
    printf("\nValue of y = %g", Y)
    printf("\nValue of z = %g", Z)
    x = X
    y = Y
    z = Z
end
```

Output -

Scilab 6.0.2 Console

```
Iteration number 1
Value of x = 0.85
Value of y = -0.9
Value of z = 1.25
Iteration number 2
Value of x = 1.02
Value of y = -0.965
Value of z = 1.03
Iteration number 3
Value of x = 1.00125
Value of y = -1.0015
Value of z = 1.00325
Iteration number 4
Value of x = 1.0004
Value of y = -1.00002
Value of z = 0.99965
Iteration number 5
Value of x = 0.999966
Value of y = -1.00008
Value of z = 0.999956
Iteration number 6
Value of x = 1
Value of y = -0.999997
Value of z = 0.999992
Iteration number 7
Value of x = 0.999999
Value of y = -1
Value of z = 1
Iteration number 8
Value of x = 1
Value of y = -1
Value of z = 1
Iteration number 9
Value of x = 1
Value of y = -1
Value of z = 1
Iteration number 10
Value of x = 1
Value of y = -1
Value of z = 1
-->
```

Program 6

Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Jacobi method. Perform 11 iterations. Equations are:

$15x + 2y + z = 18$

$2x + 20y - 3z = 19$

$3x - 6y + 25z = 22$

Code -

```
clc;
clear all
A = [15 2 1; 2 20 -3; 3 -6 25]
B = [18; 19; 22]
n = 11;
x = 0;
y = 0;
z = 0;
for i = 1:n
    printf("\nIteration number %g", i)
    X = (B(1)- A(1,2)*y-A(1,3)*z)/A(1,1)
    Y = (B(2)- A(2,1)*x-A(2,3)*z)/A(2,2)
    Z = (B(3)- A(3,1)*x-A(3,2)*y)/A(3,3)
    printf("\nValue of x = %g", X)
    printf("\nValue of y = %g", Y)
    printf("\nValue of z = %g", Z)
    x = X
    y = Y
    z = Z
end
```

Output -

```
Iteration number 1
Value of x = 1.2
Value of y = 0.95
Value of z = 0.88
Iteration number 2
Value of x = 1.01467
Value of y = 0.962
Value of z = 0.964
Iteration number 3
Value of x = 1.00747
Value of y = 0.993133
Value of z = 0.98912
Iteration number 4
Value of x = 1.00164
Value of y = 0.997621
Value of z = 0.997456
Iteration number 5
Value of x = 1.00049
Value of y = 0.999454
Value of z = 0.999232
Iteration number 6
Value of x = 1.00012
Value of y = 0.999836
Value of z = 0.999811
Iteration number 7
Value of x = 1.00003
Value of y = 0.999959
Value of z = 0.999946
Iteration number 8
Value of x = 1.00001
Value of y = 0.999988
Value of z = 0.999986
Iteration number 9
Value of x = 1
Value of y = 0.999997
Value of z = 0.999996
Iteration number 10
Value of x = 1
Value of y = 0.999999
Value of z = 0.999999
```

```
Iteration number 11
Value of x = 1
Value of y = 1
Value of z = 1
-->
```

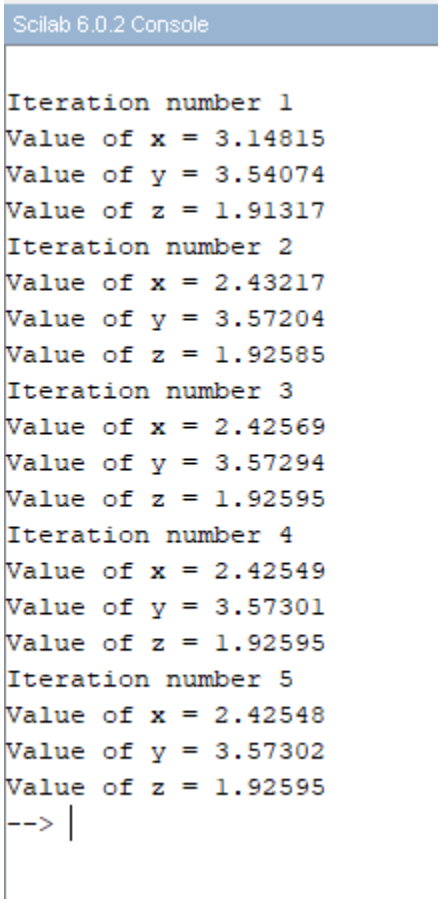
Scilab 6 - Gauss Seidel method

Program 1
Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Seidel method. Perform 5 iterations. Equations are:
27x + 6y - z = 85
6x + 15y + 2z = 72
x + y + 54z = 110

Code -

```
clc;
clear all
A = [27 6 -1; 6 15 2; 1 1 54]
B = [85; 72; 110]
n = 5;
X = 0;
Y = 0;
Z = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*Y-A(1,3)*Z)/A(1,1)
   Y = (B(2)- A(2,1)*X-A(2,3)*Z)/A(2,2)
   Z = (B(3)- A(3,1)*X-A(3,2)*Y)/A(3,3)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
end
```

Output -


Scilab 6.0.2 Console
Iteration number 1
Value of x = 3.14815
Value of y = 3.54074
Value of z = 1.91317
Iteration number 2
Value of x = 2.43217
Value of y = 3.57204
Value of z = 1.92585
Iteration number 3
Value of x = 2.42569
Value of y = 3.57294
Value of z = 1.92595
Iteration number 4
Value of x = 2.42549
Value of y = 3.57301
Value of z = 1.92595
Iteration number 5
Value of x = 2.42548
Value of y = 3.57302
Value of z = 1.92595
-->

Program 2

Write a Scilab code to solve the following set of equations in terms of x, y, z and w by using Gauss Seidel method. Perform 10 iterations. Equations are:

10x - 2y - z - w = 3

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

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

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

Code -

```
clc;
clear all
A = [10 -2 -1 -1; -2 10 -1 -1; -1 -1 10 -2; -1 -1 -2 10]
B = [3; 15; 27; -9]
n = 10;
X = 0;
Y = 0;
Z = 0;
W = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*Y-A(1,3)*Z-A(1,4)*W)/A(1,1)
   Y = (B(2)- A(2,1)*X-A(2,3)*Z-A(2,4)*W)/A(2,2)
   Z = (B(3)- A(3,1)*X-A(3,2)*Y-A(3,4)*W)/A(3,3)
   W = (B(4)- A(4,1)*X-A(4,2)*Y-A(4,3)*Z)/A(4,4)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
   printf("\nValue of w = %g", W)
end
```

Output -

```
Iteration number 1
Value of x = 0.3
Value of y = 1.56
Value of z = 2.886
Value of w = -0.1368
Iteration number 2
Value of x = 0.88692
Value of y = 1.9523
Value of z = 2.95656
Value of w = -0.0247651
Iteration number 3
Value of x = 0.983641
Value of y = 1.98991
Value of z = 2.9924
Value of w = -0.0041648
Iteration number 4
Value of x = 0.996805
Value of y = 1.99818
Value of z = 2.99867
Value of w = -0.000767789
Iteration number 5
Value of x = 0.999427
Value of y = 1.99968
Value of z = 2.99976
Value of w = -0.000138477
Iteration number 6
Value of x = 0.999897
Value of y = 1.99994
Value of z = 2.99996
Value of w = -2.49744e-05
Iteration number 7
Value of x = 0.999981
Value of y = 1.99999
Value of z = 2.99999
Value of w = -4.51086e-06
Iteration number 8
Value of x = 0.999997
Value of y = 2
Value of z = 3
Value of w = -8.14402e-07
```

```
Iteration number 9
Value of x = 0.999999
Value of y = 2
Value of z = 3
Value of w = -1.47037e-07
Iteration number 10
Value of x = 1
Value of y = 2
Value of z = 3
Value of w = -2.65476e-08
```

Program 3

Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Seidel method. Perform 7 iterations. Equations are:

$28x + 4y - z = 32$

$2x + 17y + 4z = 35$

$x + 3y + 10z = 24$

Code -

```
clc;
clear all
A = [28 4 -1; 2 17 4; 1 3 10]
B = [32; 35; 24]
n = 7;
X = 0;
Y = 0;
Z = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*Y-A(1,3)*Z)/A(1,1)
   Y = (B(2)- A(2,1)*X-A(2,3)*Z)/A(2,2)
   Z = (B(3)- A(3,1)*X-A(3,2)*Y)/A(3,3)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
end
```

Output -

```
Iteration number 1
Value of x = 1.14286
Value of y = 1.92437
Value of z = 1.7084
Iteration number 2
Value of x = 0.928962
Value of y = 1.54756
Value of z = 1.84284
Iteration number 3
Value of x = 0.987593
Value of y = 1.50903
Value of z = 1.84853
Iteration number 4
Value of x = 0.993301
Value of y = 1.50702
Value of z = 1.84857
Iteration number 5
Value of x = 0.993589
Value of y = 1.50697
Value of z = 1.84855
Iteration number 6
Value of x = 0.993595
Value of y = 1.50698
Value of z = 1.84855
Iteration number 7
Value of x = 0.993594
Value of y = 1.50698
Value of z = 1.84855
-->
```

Program 4

Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Seidel method. Perform 10 iterations. Equations are:

$4x - 2y - z = 40$

$x - 6y + 2z = -28$

$x - 2y + 12z = -86$

Code -

```
clc;
clear all
A = [4 -2 -1; 1 -6 2; 1 -2 12]
B = [40; -28; -86]
n = 10;
X = 0;
Y = 0;
Z = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*Y-A(1,3)*Z)/A(1,1)
   Y = (B(2)- A(2,1)*X-A(2,3)*Z)/A(2,2)
   Z = (B(3)- A(3,1)*X-A(3,2)*Y)/A(3,3)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
end
```

Output -

Scilab 6.0.2 Console

```
Iteration number 1
Value of x = 10
Value of y = 6.33333
Value of z = -6.94444
Iteration number 2
Value of x = 11.4306
Value of y = 4.25694
Value of z = -7.40972
Iteration number 3
Value of x = 10.276
Value of y = 3.90943
Value of z = -7.37143
Iteration number 4
Value of x = 10.1119
Value of y = 3.89483
Value of z = -7.36018
Iteration number 5
Value of x = 10.1074
Value of y = 3.89783
Value of z = -7.35931
Iteration number 6
Value of x = 10.1091
Value of y = 3.89841
Value of z = -7.35936
Iteration number 7
Value of x = 10.1094
Value of y = 3.89844
Value of z = -7.35937
Iteration number 8
Value of x = 10.1094
Value of y = 3.89844
Value of z = -7.35938
Iteration number 9
Value of x = 10.1094
Value of y = 3.89844
Value of z = -7.35938
Iteration number 10
Value of x = 10.1094
Value of y = 3.89844
Value of z = -7.35938
-->
```

Program 5

Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Seidel method. Perform 10 iterations. Equations are:

$3x - 0.1y - 0.2z = 7.85$

$0.1x + 7y - 0.3z = -19.3$

$0.3x - 0.2y + 10z = 71.4$

Code -

```
clc;
clear all
A = [3 -0.1 -0.2; 0.1 7 -0.3; 0.3 -0.2 10]
B = [7.85; -19.3; 71.4]
n = 10;
X = 0;
Y = 0;
Z = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*Y-A(1,3)*Z)/A(1,1)
   Y = (B(2)- A(2,1)*X-A(2,3)*Z)/A(2,2)
   Z = (B(3)- A(3,1)*X-A(3,2)*Y)/A(3,3)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
end
```

Output -

Scilab 6.0.2 Console

```
Iteration number 1
Value of x = 2.61667
Value of y = -2.79452
Value of z = 7.00561
Iteration number 2
Value of x = 2.99056
Value of y = -2.49962
Value of z = 7.00029
Iteration number 3
Value of x = 3.00003
Value of y = -2.49999
Value of z = 7
Iteration number 4
Value of x = 3
Value of y = -2.5
Value of z = 7
Iteration number 5
Value of x = 3
Value of y = -2.5
Value of z = 7
Iteration number 6
Value of x = 3
Value of y = -2.5
Value of z = 7
Iteration number 7
Value of x = 3
Value of y = -2.5
Value of z = 7
Iteration number 8
Value of x = 3
Value of y = -2.5
Value of z = 7
Iteration number 9
Value of x = 3
Value of y = -2.5
Value of z = 7
Iteration number 10
Value of x = 3
Value of y = -2.5
Value of z = 7
-->
```

Program 6

Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Seidel method. Perform 10 iterations. Equations are:

$110x + y + z = 187$

$2x + 140y + z = 140$

$16x + 2y + 210z = 157$

Code -

```
clc;
clear all
A = [110 1 1; 2 140 1; 16 2 210]
B = [187; 140; 157]
n = 10;
X = 0;
Y = 0;
Z = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*Y-A(1,3)*Z)/A(1,1)
   Y = (B(2)- A(2,1)*X-A(2,3)*Z)/A(2,2)
   Z = (B(3)- A(3,1)*X-A(3,2)*Y)/A(3,3)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
end
```

Output -

Scilab 6.0.2 Console

```
Iteration number 1
Value of x = 1.7
Value of y = 0.975714
Value of z = 0.608803
Iteration number 2
Value of x = 1.6856
Value of y = 0.971571
Value of z = 0.60994
Iteration number 3
Value of x = 1.68562
Value of y = 0.971563
Value of z = 0.609938
Iteration number 4
Value of x = 1.68562
Value of y = 0.971563
Value of z = 0.609938
Iteration number 5
Value of x = 1.68562
Value of y = 0.971563
Value of z = 0.609938
Iteration number 6
Value of x = 1.68562
Value of y = 0.971563
Value of z = 0.609938
Iteration number 7
Value of x = 1.68562
Value of y = 0.971563
Value of z = 0.609938
Iteration number 8
Value of x = 1.68562
Value of y = 0.971563
Value of z = 0.609938
Iteration number 9
Value of x = 1.68562
Value of y = 0.971563
Value of z = 0.609938
Iteration number 10
Value of x = 1.68562
Value of y = 0.971563
Value of z = 0.609938
-->
```

Program 7

Write a Scilab code to solve the following set of equations in terms of x, y, z and w by using Gauss Seidel method. Perform 10 iterations. Equations are:

$40x - 2y - z - w = 37$

$3x + 70y - z - w = 152$

$2x - y + 110z - 20w = 274$

$x + y - 30z + 40w = -90$

Code -

```
clc;
clear all
A = [40 -2 -1 -1; 3 70 -1 -1; 2 -1 110 -20; 1 1 -30 40]
B = [37; 152; 274; -90]
n = 10;
X = 0;
Y = 0;
Z = 0;
W = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*Y-A(1,3)*Z-A(1,4)*W)/A(1,1)
   Y = (B(2)- A(2,1)*X-A(2,3)*Z-A(2,4)*W)/A(2,2)
   Z = (B(3)- A(3,1)*X-A(3,2)*Y-A(3,4)*W)/A(3,3)
   W = (B(4)- A(4,1)*X-A(4,2)*Y-A(4,3)*Z)/A(4,4)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
   printf("\nValue of w = %g", W)
end
```

Output -

```
Iteration number 1
Value of x = 0.925
Value of y = 2.13179
Value of z = 2.49347
Value of w = -0.456317
Iteration number 2
Value of x = 1.08252
Value of y = 2.15414
Value of z = 2.40784
Value of w = -0.525034
Iteration number 3
Value of x = 1.07978
Value of y = 2.15205
Value of z = 2.39538
Value of w = -0.534261
Iteration number 4
Value of x = 1.07913
Value of y = 2.15177
Value of z = 2.39371
Value of w = -0.535489
Iteration number 5
Value of x = 1.07904
Value of y = 2.15173
Value of z = 2.39349
Value of w = -0.535652
Iteration number 6
Value of x = 1.07903
Value of y = 2.15172
Value of z = 2.39346
Value of w = -0.535674
Iteration number 7
Value of x = 1.07903
Value of y = 2.15172
Value of z = 2.39346
Value of w = -0.535677
```

```
Iteration number 8
Value of x = 1.07903
Value of y = 2.15172
Value of z = 2.39346
Value of w = -0.535677
Iteration number 9
Value of x = 1.07903
Value of y = 2.15172
Value of z = 2.39346
Value of w = -0.535677
Iteration number 10
Value of x = 1.07903
Value of y = 2.15172
Value of z = 2.39346
Value of w = -0.535677
```

Program 8

Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Seidel method. Perform 10 iterations. Equations are:

$20x + 2y - 3z = 17$

$2x + 4y - 3z = -46$

$4x + 5y + 17z = 69$

Code -

```
clc;
clear all
A = [20 2 -3; 2 4 -3; 4 5 17]
B = [17; -46; 69]
n = 10;
X = 0;
Y = 0;
Z = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*Y-A(1,3)*Z)/A(1,1)
   Y = (B(2)- A(2,1)*X-A(2,3)*Z)/A(2,2)
   Z = (B(3)- A(3,1)*X-A(3,2)*Y)/A(3,3)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
end
```

Output -

Scilab 6.0.2 Console

```
Iteration number 1
Value of x = 0.85
Value of y = -11.925
Value of z = 7.36618
Iteration number 2
Value of x = 3.14743
Value of y = -7.54908
Value of z = 5.53857
Iteration number 3
Value of x = 2.43569
Value of y = -8.56392
Value of z = 6.00452
Iteration number 4
Value of x = 2.60707
Value of y = -8.30015
Value of z = 5.88661
Iteration number 5
Value of x = 2.56301
Value of y = -8.36654
Value of z = 5.91651
Iteration number 6
Value of x = 2.57413
Value of y = -8.34968
Value of z = 5.90893
Iteration number 7
Value of x = 2.57131
Value of y = -8.35395
Value of z = 5.91085
Iteration number 8
Value of x = 2.57202
Value of y = -8.35287
Value of z = 5.91037
Iteration number 9
Value of x = 2.57184
Value of y = -8.35315
Value of z = 5.91049
Iteration number 10
Value of x = 2.57189
Value of y = -8.35308
Value of z = 5.91046
-->
```

Program 9

Write a Scilab code to solve the following set of equations in terms of x, y, z and w by using Gauss Seidel method. Perform 11 iterations. Equations are:

$276x + 42y + 3z + 8w = 467$

$22x + 115y + 142z + 17w = 526$

$219x + 7y + 422z + 8w = 637$

$43x + 2y + z + 214w = 714$

Code -

```
clc;
clear all
A = [276 42 3 8; 22 115 142 17; 219 7 422 8; 43 2 1 214]
B = [467; 526; 637; 714]
n = 11;
X = 0;
Y = 0;
Z = 0;
W = 0;
for i = 1:n
   printf("\nIteration number %g", i)
   X = (B(1)- A(1,2)*Y-A(1,3)*Z-A(1,4)*W)/A(1,1)
   Y = (B(2)- A(2,1)*X-A(2,3)*Z-A(2,4)*W)/A(2,2)
   Z = (B(3)- A(3,1)*X-A(3,2)*Y-A(3,4)*W)/A(3,3)
   W = (B(4)- A(4,1)*X-A(4,2)*Y-A(4,3)*Z)/A(4,4)
   printf("\nValue of x = %g", X)
   printf("\nValue of y = %g", Y)
   printf("\nValue of z = %g", Z)
   printf("\nValue of w = %g", W)
end
```

Output -

```
Iteration number 1
Value of x = 1.69203
Value of y = 4.25022
Value of z = 0.560887
Value of w = 2.95412
Iteration number 2
Value of x = 0.953533
Value of y = 3.26223
Value of z = 0.904521
Value of w = 3.11014
Iteration number 3
Value of x = 1.09562
Value of y = 2.78767
Value of z = 0.835697
Value of w = 3.08634
Iteration number 4
Value of x = 1.16928
Value of y = 2.86208
Value of z = 0.79669
Value of w = 3.07103
Iteration number 5
Value of x = 1.15882
Value of y = 2.91451
Value of z = 0.801537
Value of w = 3.07262
Iteration number 6
Value of x = 1.15074
Value of y = 2.90983
Value of z = 0.805776
Value of w = 3.07426
Iteration number 7
Value of x = 1.15136
Value of y = 2.90424
Value of z = 0.805517
Value of w = 3.07419
Iteration number 8
Value of x = 1.15222
Value of y = 2.9044
Value of z = 0.805071
Value of w = 3.07402
Iteration number 9
Value of x = 1.1522
Value of y = 2.90498
Value of z = 0.805073
Value of w = 3.07402
Iteration number 10
Value of x = 1.15211
Value of y = 2.905
Value of z = 0.805118
Value of w = 3.07404
Iteration number 11
Value of x = 1.15211
Value of y = 2.90494
Value of z = 0.805121
Value of w = 3.07404
```

Scilab 7 - Gauss Jordan method

Program 1
Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Jordan method. Equations are:
x + 3y + 2z = 2
2x + 7y + 7z = -1
2x + 5y + 2z = 7

Code -

```
clc;
A = [1 3 2; 2 7 7; 2 5 2];
disp(A)
B = [2; -1; 7];
disp(B)
C = [A B];
disp(C)
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
   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
disp("x = ")
disp(C(1,4))
disp("y = ")
disp(C(2,4))
disp("z = ")
disp(C(3,4))
```

Output -

```
  1.   3.   2.
  2.   7.   7.
  2.   5.   2.

  2.
 -1.
  7.

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

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

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

  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.

x =

  3.

y =

  1.

z =

 -2.
```

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. Equations are:

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

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

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

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

Code -

```
clc;
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
disp("x = ")
disp(C(1,5))
disp("y = ")
disp(C(2,5))
disp("z = ")
disp(C(3,5))
disp("w = ")
disp(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

 1.    1.125    0.25        1.125        5.25
 0.    1.       0.5263158   0.5789474    7.2631579
 0.   -1.5      5.          1.5          32.
 0.    2.75     5.5         5.75         52.5

 1.    1.125    0.25        1.125        5.25
 0.    1.       0.5263158   0.5789474    7.2631579
 0.    0.       5.7894737   2.3684211    42.894737
 0.    0.       4.0526316   4.1578947    32.526316

 1.    1.125    0.25        1.125        5.25
 0.    1.       0.5263158   0.5789474    7.2631579
 0.    0.       1.          0.4090909    7.4090909
 0.    0.       4.0526316   4.1578947    32.526316

 1.    1.125    0.25        1.125        5.25
 0.    1.       0.5263158   0.5789474    7.2631579
 0.    0.       1.          0.4090909    7.4090909
 0.    0.       0.          2.5          2.5

 1.    1.125    0.25        1.125        5.25
 0.    1.       0.5263158   0.5789474    7.2631579
 0.    0.       1.          0.4090909    7.4090909
 0.    0.       0.          1.           1.

 1.    1.125    0.25        1.125        5.25
 0.    1.       0.5263158   0.5789474    7.2631579
 0.    0.       1.          0.4090909    7.4090909
 0.    0.       0.          1.           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 set of equations in terms of x, y and z by using Gauss Jordan method. Equations are:

x + 2y + 6z = 22

3x + 4y + z = 26

6x - y - z = 19

Code -

```
clc;
A = [1 2 6; 3 4 1; 6 -1 -1];
disp(A)
B = [22; 26; 19];
disp(B)
C = [A B];
disp(C)
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
   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
disp("x = ")
disp(C(1,4))
disp("y = ")
disp(C(2,4))
disp("z = ")
disp(C(3,4))
```

Output -

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

 22.
 26.
 19.

 1.   2.   6.   22.
 3.   4.   1.   26.
 6.  -1.  -1.   19.

 1.   2.   6.   22.
 3.   4.   1.   26.
 6.  -1.  -1.   19.

 1.   2.    6.    22.
 0.  -2.   -17.  -40.
 0.  -13.  -37.  -113.

 1.   2.    6.    22.
 0.   1.    8.5   20.
 0.  -13.  -37.  -113.

 1.   2.   6.     22.
 0.   1.   8.5    20.
 0.   0.   73.5   147.

 1.   2.   6.    22.
 0.   1.   8.5   20.
 0.   0.   1.    2.

 1.   2.   6.    22.
 0.   1.   8.5   20.
 0.   0.   1.    2.

 1.   2.   0.   10.
 0.   1.   0.   3.
 0.   0.   1.   2.

   1.   0.   0.   4.
   0.   1.   0.   3.
   0.   0.   1.   2.

x =

   4.

y =

   3.

z =

   2.
```

Program 4

Write a Scilab code to solve the following set of equations in terms of x, y, z and w by using Gauss Jordan method. Equations are:

$2x + y - z + 3w = 8$

$x + y + z - w = -2$

$3x + 2y - z = 6$

$4y + 3z + 2w = -8$

Code -

```
clc;
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
disp("x = ")
disp(C(1,5))
disp("y = ")
disp(C(2,5))
disp("z = ")
disp(C(3,5))
disp("w = ")
disp(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.
 0.    1.    0.    0.   -1.
 0.    0.    1.    0.   -2.
 0.    0.    0.    1.    1.

x =

   2.

y =

  -1.

z =

  -2.

w =

   1.
```

Program 5

Write a Scilab code to solve the following set of equations in terms of x, y, z and w by using Gauss Jordan method. Equations are:

$2x + y - z + 3w = 11$

$x - 2y + z - w = 8$

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

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

Code -

```
clc;
A = [2 1 -1 3; 1 -2 1 1; 4 7 2 -1; 3 5 4 4];
disp(A)
B = [11; 8; 0; 17];
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
disp("x = ")
disp(C(1,5))
disp("y = ")
disp(C(2,5))
disp("z = ")
disp(C(3,5))
disp("w = ")
disp(C(4,5))
```

Output -

```
 2.    1.   -1.    3.
 1.   -2.    1.    1.
 4.    7.    2.   -1.
 3.    5.    4.    4.

 11.
 8.
 0.
 17.

 2.    1.   -1.    3.    11.
 1.   -2.    1.    1.    8.
 4.    7.    2.   -1.    0.
 3.    5.    4.    4.    17.

 1.    0.5  -0.5   1.5   5.5
 1.   -2.    1.    1.    8.
 4.    7.    2.   -1.    0.
 3.    5.    4.    4.    17.

 1.    0.5  -0.5   1.5   5.5
 0.   -2.5   1.5  -0.5   2.5
 0.    5.    4.   -7.   -22.
 0.    3.5   5.5  -0.5   0.5

 1.    0.5  -0.5   1.5   5.5
 0.    1.   -0.6   0.2  -1.
 0.    5.    4.   -7.   -22.
 0.    3.5   5.5  -0.5   0.5

 1.    0.5  -0.5   1.5   5.5
 0.    1.   -0.6   0.2  -1.
 0.    0.    7.   -8.   -17.
 0.    0.    7.6  -1.2   4.

 1.    0.5  -0.5   1.5         5.5
 0.    1.   -0.6   0.2        -1.
 0.    0.    1.   -1.1428571  -2.4285714
 0.    0.    7.6  -1.2         4.

1.    0.5  -0.5   1.5         5.5
0.    1.   -0.6   0.2        -1.
0.    0.    1.   -1.1428571  -2.4285714
0.    0.    0.    7.4857143   22.457143

1.    0.5  -0.5   1.5         5.5
0.    1.   -0.6   0.2        -1.
0.    0.    1.   -1.1428571  -2.4285714
0.    0.    0.    1.          3.

 1.    0.5  -0.5   1.5         5.5
 0.    1.   -0.6   0.2        -1.
 0.    0.    1.   -1.1428571  -2.4285714
 0.    0.    0.    1.          3.

 1.    0.5  -0.5   0.    1.
 0.    1.   -0.6   0.   -1.6
 0.    0.    1.    0.    1.
 0.    0.    0.    1.    3.

 1.    0.5   0.    0.    1.5
 0.    1.    0.    0.   -1.
 0.    0.    1.    0.    1.
 0.    0.    0.    1.    3.

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

x =

   2.

y =

  -1.

z =

   1.

w =

   3.
```

Program 6

Write a Scilab code to solve the following set of equations in terms of x, y, z and w by using Gauss Jordan method. Equations are:

$18x + 9y + 2z + 9w = 421$

$2x + 17y + 3z + 5w = 145$

$4x + 3y + 8z + 36w = 537$

$2x + 5y + 16z + 8w = 637$

Code -

```
clc;
A = [18 9 2 9; 2 17 3 5; 4 3 8 36; 2 5 16 8];
disp(A)
B = [421; 145; 537; 637];
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
disp("x = ")
disp(C(1,5))
disp("y = ")
disp(C(2,5))
disp("z = ")
disp(C(3,5))
disp("w = ")
disp(C(4,5))
```

Output -

```
18.    9.    2.    9.
2.    17.    3.    5.
4.    3.    8.    36.
2.    5.    16.    8.

421.
145.
537.
637.

18.    9.    2.    9.    421.
2.    17.    3.    5.    145.
4.    3.    8.    36.    537.
2.    5.    16.    8.    637.

1.    0.5    0.1111111    0.5    23.388889
2.    17.    3.           5.     145.
4.    3.     8.           36.    537.
2.    5.     16.          8.     637.

1.    0.5    0.1111111    0.5    23.388889
0.    16.    2.7777778    4.     98.222222
0.    1.     7.5555556    34.    443.44444
0.    4.     15.777778    7.     590.22222

1.    0.5    0.1111111    0.5     23.388889
0.    1.     0.1736111    0.25    6.1388889
0.    1.     7.5555556    34.     443.44444
0.    4.     15.777778    7.      590.22222

1.    0.5    0.1111111    0.5      23.388889
0.    1.     0.1736111    0.25     6.1388889
0.    0.     7.3819444    33.75    437.30556
0.    0.     15.083333    6.       565.66667

1.    0.5    0.1111111    0.5          23.388889
0.    1.     0.1736111    0.25         6.1388889
0.    0.     1.           4.5719661    59.239887
0.    0.     15.083333    6.           565.66667

1.    0.5    0.1111111    0.5          23.388889
0.    1.     0.1736111    0.25         6.1388889
0.    0.     1.           4.5719661    59.239887
0.    0.     0.          -62.960489   -327.8683

1.    0.5    0.1111111    0.5          23.388889
0.    1.     0.1736111    0.25         6.1388889
0.    0.     1.           4.5719661    59.239887
0.    0.     0.           1.           5.2075246

1.    0.5    0.1111111    0.5          23.388889
0.    1.     0.1736111    0.25         6.1388889
0.    0.     1.           4.5719661    59.239887
0.    0.     0.           1.           5.2075246

1.    0.5    0.1111111    0.    20.785127
0.    1.     0.1736111    0.    4.8370077
0.    0.     1.           0.    35.431261
0.    0.     0.           1.    5.2075246

1.    0.5    0.    0.    16.84832
0.    1.     0.    0.   -1.3142528
0.    0.     1.    0.    35.431261
0.    0.     0.    1.    5.2075246

1.    0.    0.    0.    17.505446
0.    1.    0.    0.   -1.3142528
0.    0.    1.    0.    35.431261
0.    0.    0.    1.    5.2075246

x =

  17.505446

y =

  -1.3142528

z =

  35.431261

w =

  5.2075246
```

Program 7

Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Jordan method. Equations are:

$2x + 2y + 16z = 57$

$13x + 34y + z = 62$

$36x - y + 4z = 61$

Code -

```
clc;
A = [2 2 16; 13 34 1; 36 -1 4];
disp(A)
B = [57; 62; 61];
disp(B)
C = [A B];
disp(C)
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
   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
disp("x = ")
disp(C(1,4))
disp("y = ")
disp(C(2,4))
disp("z = ")
disp(C(3,4))
```

Output -

```
2.    2.    16.
13.   34.   1.
36.  -1.    4.

57.
62.
61.

2.    2.    16.   57.
13.   34.   1.    62.
36.  -1.    4.    61.

1.    1.    8.   28.5
13.   34.   1.   62.
36.  -1.    4.   61.

1.   1.    8.     28.5
0.   21.  -103.  -308.5
0.  -37.  -284.  -965.

1.   1.    8.          28.5
0.   1.   -4.9047619  -14.690476
0.  -37.  -284.       -965.

1.   1.   8.          28.5
0.   1.  -4.9047619  -14.690476
0.   0.  -465.47619  -1508.5476

1.   1.   8.          28.5
0.   1.  -4.9047619  -14.690476
0.   0.   1.          3.2408696

1.   1.   8.          28.5
0.   1.  -4.9047619  -14.690476
0.   0.   1.          3.2408696

1.   1.   0.   2.5730435
0.   1.   0.   1.2052174
0.   0.   1.   3.2408696

1.   0.   0.   1.3678261
0.   1.   0.   1.2052174
0.   0.   1.   3.2408696

x =

   1.3678261

y =

   1.2052174

z =

   3.2408696
```

Program 8

Write a Scilab code to solve the following set of equations in terms of x, y and z by using Gauss Jordan method. Equations are:

2x + 7y - z = 10

4x + 9z = 17

3x - 5y + z = 11

Code -

```
clc;
A = [2 7 -1; 4 0 9; 3 -5 1];
disp(A)
B = [10; 17; 11];
disp(B)
C = [A B];
disp(C)
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
   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
disp("x = ")
disp(C(1,4))
disp("y = ")
disp(C(2,4))
disp("z = ")
disp(C(3,4))
```

Output -

```
 2.    7.   -1.
 4.    0.    9.
 3.   -5.    1.

 10.
 17.
 11.

 2.    7.   -1.    10.
 4.    0.    9.    17.
 3.   -5.    1.    11.

 1.    3.5  -0.5    5.
 4.    0.    9.     17.
 3.   -5.    1.     11.

 1.    3.5   -0.5    5.
 0.   -14.    11.   -3.
 0.   -15.5   2.5   -4.

 1.    3.5   -0.5          5.
 0.    1.    -0.7857143    0.2142857
 0.   -15.5   2.5          -4.

 1.    3.5  -0.5          5.             1.    0.    0.    4.0922509
 0.    1.   -0.7857143    0.2142857      0.    1.    0.    0.2693727
 0.    0.   -9.6785714   -0.6785714      0.    0.    1.    0.0701107

 1.    3.5  -0.5          5.           x =
 0.    1.   -0.7857143    0.2142857
 0.    0.    1.           0.0701107      4.0922509

 1.    3.5  -0.5          5.           y =
 0.    1.   -0.7857143    0.2142857
 0.    0.    1.           0.0701107      0.2693727

 1.    3.5   0.    5.0350554           z =
 0.    1.    0.    0.2693727
 0.    0.    1.    0.0701107             0.0701107
```

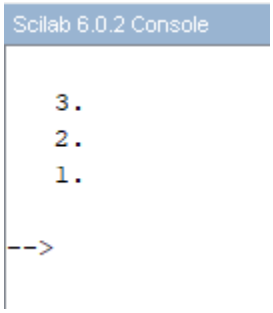
Scilab 8 - Eigenvalue

Program 1
Write a Scilab code to find the eigenvalues of the following matrix.
A = [2 -1 1 1 2 -1 1 -1 2]

Code -

```
clc
clear all
A = [2 -1 1; 1 2 -1; 1 -1 2];
a = A(1,1)+A(2,2)+A(3,3);
b = ((A(2,2)*A(3,3))-(A(3,2)*A(2,3))) + ((A(1,1)*A(3,3))-(A(3,1)*A(1,3))) +
(A(1,1)*A(2,2))-(A(2,1)*A(1,2));
m = det(A);
p = [1 -a b -m];
m = roots(p);
disp(m);
```

Output -



Program 2

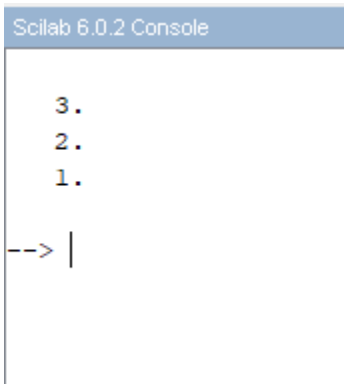
Write a Scilab code to find the eigenvalues of the following matrix.

A = [8 -8 -2 4 -3 -2 3 -4 1]

Code -

```
clc
clear all
A = [8 -8 -2; 4 -3 -2; 3 -4 1];
a = A(1,1)+A(2,2)+A(3,3);
b = ((A(2,2)*A(3,3))-(A(3,2)*A(2,3))) + ((A(1,1)*A(3,3))-(A(3,1)*A(1,3))) +
(A(1,1)*A(2,2))-(A(2,1)*A(1,2));
m = det(A);
p = [1 -a b -m];
m = roots(p);
disp(m);
```

Output -

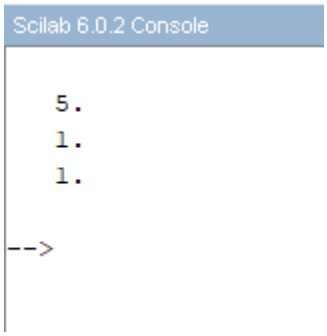


Program 3
Write a Scilab code to find the eigenvalues of the following matrix.
A = [2 2 1 1 3 1 1 2 2]

Code -

```
clc
clear all
A = [2 2 1; 1 3 1; 1 2 2];
a = A(1,1)+A(2,2)+A(3,3);
b = ((A(2,2)*A(3,3))-(A(3,2)*A(2,3))) + ((A(1,1)*A(3,3))-(A(3,1)*A(1,3))) +
(A(1,1)*A(2,2))-(A(2,1)*A(1,2));
m = det(A);
p = [1 -a b -m];
m = roots(p);
disp(m);
```

Output -

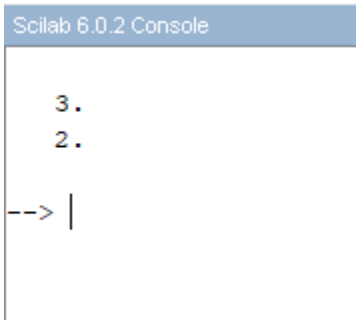


Program 4
Write a Scilab code to find the eigenvalues of the following matrix.
A = [4 -2 1 1]

Code -

```
clc
clear all
A = [4 -2; 1 1];
a = A(1,1)+A(2,2);
b = (A(1,1)*A(2,2))-(A(2,1)*A(1,2));
m = det(A);
p = [1 -a b];
m = roots(p);
disp(m);
```

Output -

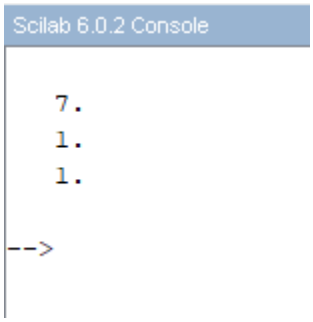


Program 5
Write a Scilab code to find the eigenvalues of the following matrix.
A =[2 1 1 2 3 2 3 3 4]

Code -

```
clc
clear all
A = [2 1 1; 2 3 2; 3 3 4];
a = A(1,1)+A(2,2)+A(3,3);
b = ((A(2,2)*A(3,3))-(A(3,2)*A(2,3))) + ((A(1,1)*A(3,3))-(A(3,1)*A(1,3))) +
(A(1,1)*A(2,2))-(A(2,1)*A(1,2));
m = det(A);
p = [1 -a b -m];
m = roots(p);
disp(m);
```

Output -



Program 6

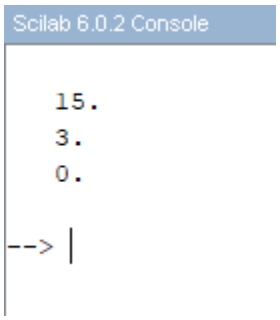
Write a Scilab code to find the eigenvalues of the following matrix.

A = [8 -6 2 -6 7 -4 2 -4 3]

Code -

```
clc
clear all
A = [8 -6 2; -6 7 -4; 2 -4 3];
a = A(1,1)+A(2,2)+A(3,3);
b = ((A(2,2)*A(3,3))-(A(3,2)*A(2,3))) + ((A(1,1)*A(3,3))-(A(3,1)*A(1,3))) +
(A(1,1)*A(2,2))-(A(2,1)*A(1,2));
m = det(A);
p = [1 -a b -m];
m = roots(p);
disp(m);
```

Output -



Scilab 9 - Eigenvalue and Eigenvector

Program 1

Write a Scilab code to find the Eigenvalue and Eigenvector of the following matrix.

A = [2 -1 1 1 2 -1 1 -1 2]

Code -

```
clc
clear all
A = [2 -1 1; 1 2 -1; 1 -1 2];
[c,d] = spec(A);
disp("The spec of A is: ");
disp(spec(A));
disp("The Eigen-values of matrix A are: ");
disp(d);
disp("The corresponding Eigen-vectors of matrix A is: ");
disp(c);
```

Output -

```
Matrix is:

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

The spec of A is:

  2.
  1.
  3.

The Eigen-values of matrix A are:

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

The corresponding Eigen-vectors of matrix A is:

  0.5773503   2.621D-16   0.7071068
  0.5773503  -0.7071068   2.604D-16
  0.5773503  -0.7071068   0.7071068

-->
```

Program 2

Write a Scilab code to find the Eigenvalue and Eigenvector of the following matrix.

A = [8 -8 -2 4 -3 -2 3 -4 1]

Code -

```
clc
clear all
A = [8 -8 -2; 4 -3 -2; 3 -4 1];
disp("Matrix is: ")
disp(A);
[c,d] = spec(A);
disp("The spec of A is: ");
disp(spec(A));
disp("The Eigen-values of matrix A are: ");
disp(d);
disp("The corresponding Eigen-vectors of matrix A is: ");
disp(c);
```

Output -

```
Matrix is:

   8.   -8.   -2.
   4.   -3.   -2.
   3.   -4.    1.

The spec of A is:

   1.
   3.
   2.

The Eigen-values of matrix A are:

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

The corresponding Eigen-vectors of matrix A is:

  -0.7427814  -0.8164966  -0.8017837
  -0.557086   -0.4082483  -0.5345225
  -0.3713907  -0.4082483  -0.2672612

-->
```

Program 3
Write a Scilab code to find the eigenvalues of the following matrix.
A = [2 2 1 1 3 1 1 2 2]

Code -

```
clc
clear all
A = [2 2 1; 1 3 1; 1 2 2];
disp("Matrix is: ")
disp(A);
[c,d] = spec(A);
disp("The spec of A is: ");
disp(spec(A));
disp("The Eigen-values of matrix A are: ");
disp(d);
disp("The corresponding Eigen-vectors of matrix A is: ");
disp(c);
```

Output -

```
Matrix is:

  2.    2.    1.
  1.    3.    1.
  1.    2.    2.

The spec of A is:

  1.
  5.
  1.

The Eigen-values of matrix A are:

  1.    0.    0.
  0.    5.    0.
  0.    0.    1.

The corresponding Eigen-vectors of matrix A is:

 -0.904534    0.5773503   0.1431312
  0.3015113   0.5773503  -0.4989347
  0.3015113   0.5773503   0.8547383
```

Program 4
Write a Scilab code to find the eigenvalues of the following matrix.
A = [4 -2 1 1]

Code -

```
clc
clear all
A = [4 -2; 1 1];
disp("Matrix is: ")
disp(A);
[c,d] = spec(A);
disp("The spec of A is: ");
disp(spec(A));
disp("The Eigen-values of matrix A are: ");
disp(d);
disp("The corresponding Eigen-vectors of matrix A is: ");
disp(c);
```

Output -

```
Matrix is:

  4.  -2.
  1.   1.

The spec of A is:

  3.
  2.

The Eigen-values of matrix A are:

  3.   0.
  0.   2.

The corresponding Eigen-vectors of matrix A is:

  0.8944272   0.7071068
  0.4472136   0.7071068
```

Program 5
Write a Scilab code to find the eigenvalues of the following matrix.
A =[2 1 1 2 3 2 3 3 4]

Code -

```
clc
clear all
A = [2 1 1; 2 3 2; 3 3 4];
disp("Matrix is: ")
disp(A);
[c,d] = spec(A);
disp("The spec of A is: ");
disp(spec(A));
disp("The Eigen-values of matrix A are: ");
disp(d);
disp("The corresponding Eigen-vectors of matrix A is: ");
disp(c);
```

Output -

```
 Matrix is:

   2.    1.    1.
   2.    3.    2.
   3.    3.    4.

 The spec of A is:

   7.
   1.
   1.

 The Eigen-values of matrix A are:

   7.    0.    0.
   0.    1.    0.
   0.    0.    1.

 The corresponding Eigen-vectors of matrix A is:

  -0.2672612  -0.8111071   0.1180346
  -0.5345225   0.3244428  -0.7586964
  -0.8017837   0.4866643   0.6406618
```

Program 6
Write a Scilab code to find the eigenvalues of the following matrix.
A = [8 -6 2 -6 7 -4 2 -4 3]

Code -

```
clc
clear all
A = [8 -6 2; -6 7 -4; 2 -4 3];
disp("Matrix is: ")
disp(A);
[c,d] = spec(A);
disp("The spec of A is: ");
disp(spec(A));
disp("The Eigen-values of matrix A are: ");
disp(d);
disp("The corresponding Eigen-vectors of matrix A is: ");
disp(c);
```

Output -

```
Matrix is:

  8.  -6.   2.
 -6.   7.  -4.
  2.  -4.   3.

The spec of A is:

  1.584D-15
  3.
  15.

The Eigen-values of matrix A are:

  2.982D-15   0.   0.
  0.          3.   0.
  0.          0.   15.

The corresponding Eigen-vectors of matrix A is:

  0.3333333   0.6666667  -0.6666667
  0.6666667   0.3333333   0.6666667
  0.6666667  -0.6666667  -0.3333333
```

Name - Varin Mehta
UID - 2022300058
Class - SE Comps A (**D**)
Date - 12-03-24

Scilab ISE 1

Program 1

Output the 11th element of the 4x4 given Matrix. Also display the rank of the matrix, trace, transpose, Upper and Lower triangular matrix.

$$A = \begin{bmatrix} 77 & 28 & 75 & 17 \\ 96 & 38 & 69 & 420 \\ 82 & 23 & 45 & 12 \\ 78 & 42 & 56 & 20 \end{bmatrix}$$

Code -

```
clc;
A = [77 28 75 17; 96 38 69 420; 82 23 45 12; 78 42 56 20]
disp(A);
disp('11th element of A:');
disp(A(11));
disp('Rank of A:');
disp(rank(A));
disp('Trace of A:');
disp(trace(A));
disp('Transpose of A:');
disp(A');
disp('Upper triangular matrix of A:');
disp(triu(A));
disp('Lower triangular matrix of A:');
disp(tril(A));
```

Output -

Scilab 6.0.2 Console

```
  77.    28.    75.    17.
  96.    38.    69.    420.
  82.    23.    45.    12.
  78.    42.    56.    20.

11th element of A:

  45.

Rank of A:

  4.

Trace of A:

  180.

Transpose of A:

  77.    96.     82.    78.
  28.    38.     23.    42.
  75.    69.     45.    56.
  17.    420.    12.    20.

Upper triangular matrix of A:

  77.    28.    75.    17.
  0.     38.    69.    420.
  0.     0.     45.    12.
  0.     0.     0.     20.

Lower triangular matrix of A:

  77.    0.     0.     0.
  96.    38.    0.     0.
  82.    23.    45.    0.
  78.    42.    56.    20.
```

Program 2

Write a Scilab code to solve the following equations in terms of x, y, z and w by using gauss elimination method.

$$A = \begin{bmatrix} 2 & 4 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 2 \\ 4 & 2 & 3 & 5 \end{bmatrix}, \quad B = \begin{bmatrix} 4 \\ 0.5 \\ 1 \\ 0 \end{bmatrix}$$

Code -

```
clc;
clear all;
A = [2 4 1 1; 1 1 1 1; 1 2 4 2; 4 2 3 5]
disp('Square Matrix of co-efficients');
disp(A);
B = [4; 0.5; 1; 0]
disp('Column Matrix of constants')
disp(B);
C = [A,B]
disp('Augmented Matrix')
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)
      end
   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
   disp(C);
end
w = C(4,5)
z = C(3,5) - C(3,4)*w
y = C(2,5) - C(2,4)*w - C(2,3)*z
x = C(1,5) - C(1,4)*w - C(1,3)*z - C(1,2)*y
```

```
disp('w:');
disp(w);
disp('z:');
disp(z);
disp('y:');
disp(y);
disp('x:');
disp(x);
```

Output -

```
Square Matrix of co-efficients

  2.   4.   1.   1.
  1.   1.   1.   1.
  1.   2.   4.   2.
  4.   2.   3.   5.

Column Matrix of constants

  4.
  0.5
  1.
  0.

Augmented Matrix

  2.   4.   1.   1.   4.
  1.   1.   1.   1.   0.5
  1.   2.   4.   2.   1.
  4.   2.   3.   5.   0.

  1.   2.   0.5   0.5   2.
  1.   1.   1.    1.    0.5
  1.   2.   4.    2.    1.
  4.   2.   3.    5.    0.

  1.   2.   0.5   0.5   2.
  0.  -1.   0.5   0.5  -1.5
  0.   0.   3.5   1.5  -1.
  0.  -6.   1.    3.   -8.

  1.   2.   0.5   0.5   2.
  0.   1.  -0.5  -0.5   1.5
  0.   0.   3.5   1.5  -1.
  0.  -6.   1.    3.   -8.
```

```
  1.    2.    0.5    0.5    2.
  0.    1.   -0.5   -0.5    1.5
  0.    0.    3.5    1.5   -1.
  0.    0.   -2.     0.     1.

  1.    2.    0.5    0.5          2.
  0.    1.   -0.5   -0.5          1.5
  0.    0.    1.     0.4285714   -0.2857143
  0.    0.   -2.     0.           1.

  1.    2.    0.5    0.5          2.
  0.    1.   -0.5   -0.5          1.5
  0.    0.    1.     0.4285714   -0.2857143
  0.    0.    0.     0.8571429    0.4285714

  1.    2.    0.5    0.5          2.
  0.    1.   -0.5   -0.5          1.5
  0.    0.    1.     0.4285714   -0.2857143
  0.    0.    0.     1.           0.5

  1.    2.    0.5    0.5          2.
  0.    1.   -0.5   -0.5          1.5
  0.    0.    1.     0.4285714   -0.2857143
  0.    0.    0.     1.           0.5

w:

  0.5

z:

 -0.5

y:

  1.5

x:

 -1.
```