

ASSIGNMENT – 5

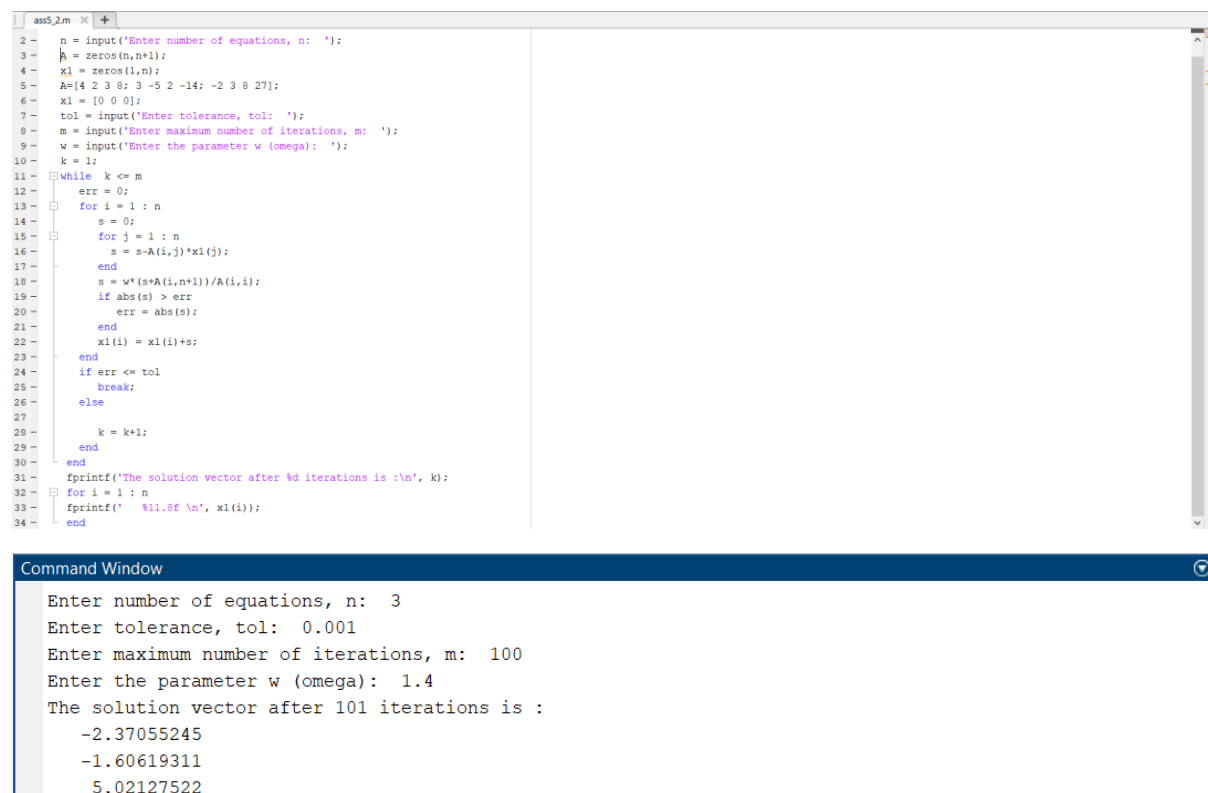
Name – Meharamt Singh

Roll No. – 102003241

Group – 2CO10

Email Id – msingh10_be20@thapar.edu

2. Write an algorithm for Successive-Over-Relaxation (SOR) method.



```
2- n = input('Enter number of equations, n: ');
3- h = zeros(n,n+1);
4- x1 = zeros(1,n);
5- A=[4 2 3 8; 3 -5 2 -14; -2 3 8 27];
6- x1 = [0 0 0];
7- tol = input('Enter tolerance, tol: ');
8- m = input('Enter maximum number of iterations, m: ');
9- w = input('Enter the parameter w (omega): ');
10- k = 1;
11- while k <= m
12-     err = 0;
13-     for i = 1 : n
14-         s = 0;
15-         for j = 1 : n
16-             s = s-A(i,j)*x1(j);
17-         end
18-         s = w*(s+A(i,n+1))/A(i,i);
19-         if abs(s) > err
20-             err = abs(s);
21-         end
22-         x1(i) = x1(i)+s;
23-     end
24-     if err <= tol
25-         break;
26-     else
27-         k = k+1;
28-     end
29- end
30- fprintf('The solution vector after %d iterations is :\n', k);
31- for i = 1 : n
32-     fprintf(' %11.8f \n', x1(i));
33- end
```

Command Window

```
Enter number of equations, n: 3
Enter tolerance, tol: 0.001
Enter maximum number of iterations, m: 100
Enter the parameter w (omega): 1.4
The solution vector after 101 iterations is :
-2.37055245
-1.60619311
5.02127522
```

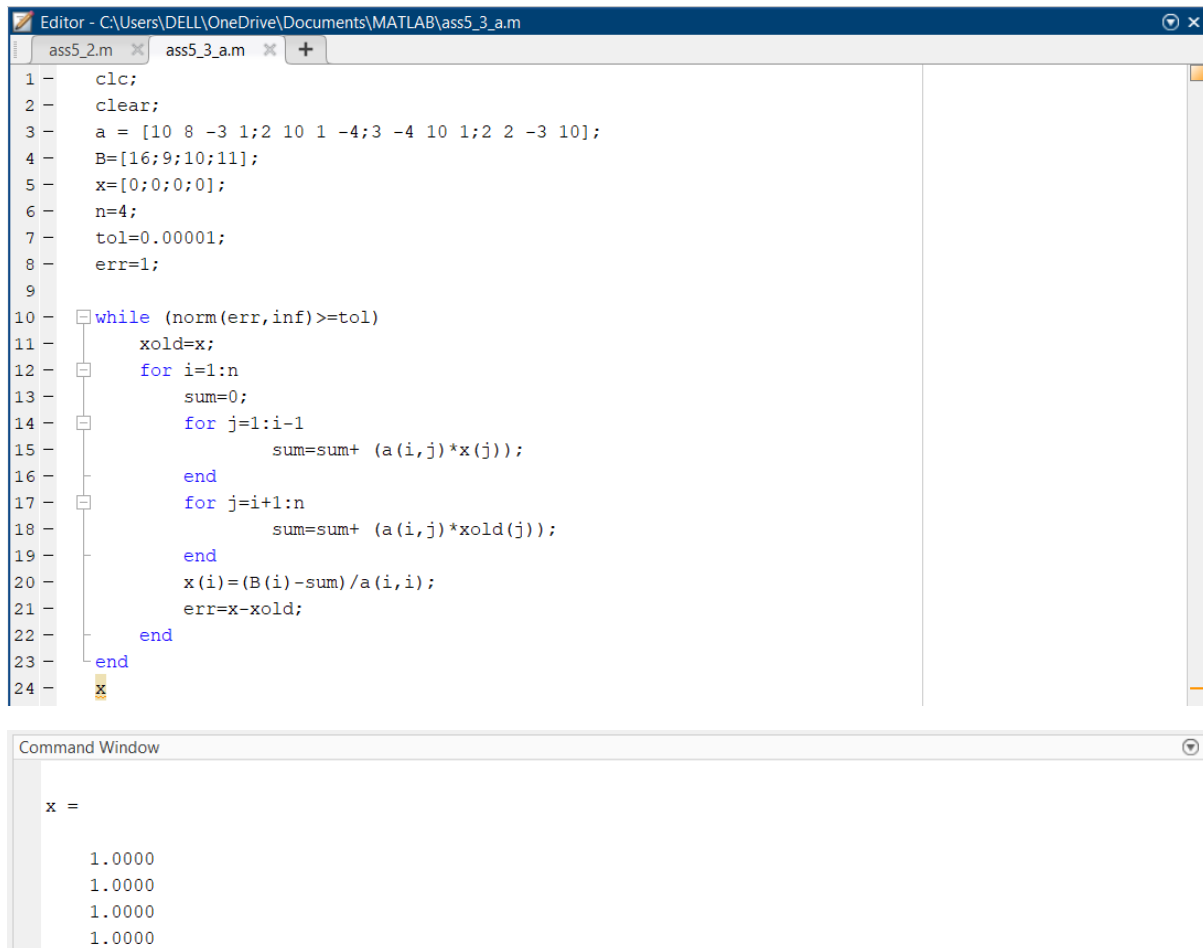
3. Use Gauss Seidel method and SOR method with $w = 1.2$ to find the solution of the following linear systems with an initial vector $[0,0,0,0]$ and tolerance value 10^{-3} in the ∞ norm:

$$(a) \ 10x + 8y - 3z + u = 16$$

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

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

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



The image shows a MATLAB Editor window with a script named 'ass5_3_a.m' and a Command Window below it. The script implements the Gauss-Seidel method for solving a system of linear equations. The matrix 'a' is a 4x4 matrix, and the vector 'B' is a 4x1 matrix. The initial vector 'x' is set to [0;0;0;0]. The tolerance 'tol' is set to 0.00001, and the error 'err' is set to 1. The method iterates until the infinity norm of the error is less than or equal to the tolerance. The final solution 'x' is displayed in the Command Window as a column vector of four 1.0000 values.

```
1 clc;
2 clear;
3 a = [10 8 -3 1; 2 10 1 -4; 3 -4 10 1; 2 2 -3 10];
4 B=[16;9;10;11];
5 x=[0;0;0;0];
6 n=4;
7 tol=0.00001;
8 err=1;
9
10 while (norm(err,inf)>=tol)
11     xold=x;
12     for i=1:n
13         sum=0;
14         for j=1:i-1
15             sum=sum+ (a(i,j)*x(j));
16         end
17         for j=i+1:n
18             sum=sum+ (a(i,j)*xold(j));
19         end
20         x(i)=(B(i)-sum)/a(i,i);
21         err=x-xold;
22     end
23 end
24
```

Command Window

```
x =
    1.0000
    1.0000
    1.0000
    1.0000
```

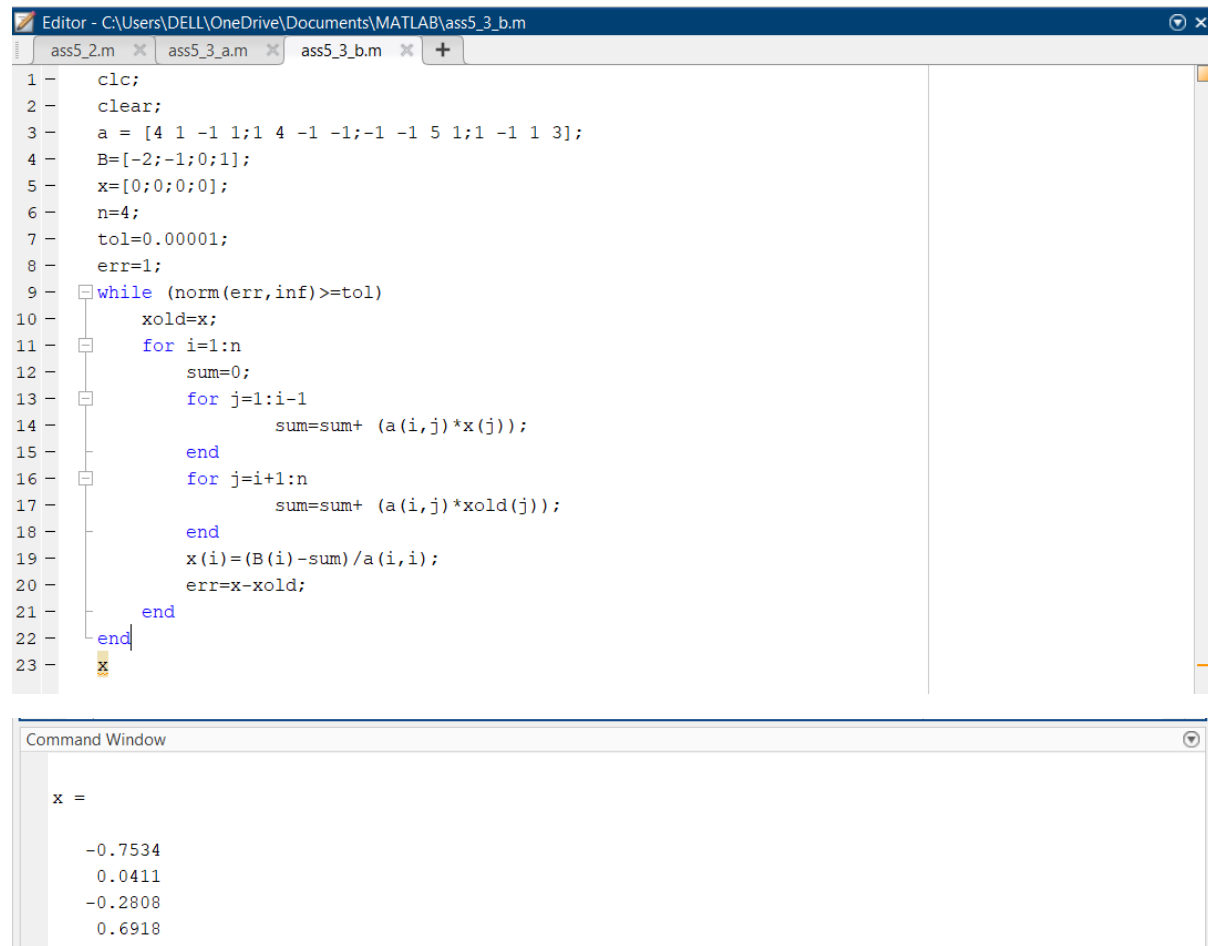
3. Use Gauss Seidel method and SOR method with $w = 1.2$ to find the solution of the following linear systems with an initial vector $[0,0,0,0]$ and tolerance value 10^{-3} in the ∞ norm:

(b) $4x_1 + x_2 - x_3 + x_4 = -2$

$$x_1 + 4x_2 - x_3 - x_4 = -1$$

$$-x_1 - x_2 + 5x_3 + x_4 = 0$$

$$1 - x_2 + x_3 + 3x_4 = 1$$



The image shows a MATLAB script in the Editor window and its execution results in the Command Window. The script implements the Gauss-Seidel method for a system of four linear equations. It initializes the coefficient matrix 'a', the right-hand side vector 'B', and the initial solution vector 'x'. A while loop iterates until the infinity norm of the error vector is less than the tolerance (0.00001). Inside the loop, the solution vector 'x' is updated sequentially for each equation, using the most recent values of the other variables. The final solution vector 'x' is displayed in the Command Window.

```
1 - clc;
2 - clear;
3 - a = [4 1 -1 1; 1 4 -1 -1; -1 -1 5 1; 1 -1 1 3];
4 - B = [-2; -1; 0; 1];
5 - x = [0; 0; 0; 0];
6 - n = 4;
7 - tol = 0.00001;
8 - err = 1;
9 - while (norm(err, inf) >= tol)
10 -     xold = x;
11 -     for i = 1:n
12 -         sum = 0;
13 -         for j = 1:i-1
14 -             sum = sum + (a(i,j)*x(j));
15 -         end
16 -         for j = i+1:n
17 -             sum = sum + (a(i,j)*xold(j));
18 -         end
19 -         x(i) = (B(i) - sum) / a(i,i);
20 -         err = x - xold;
21 -     end
22 - end
23 - x
```

Command Window

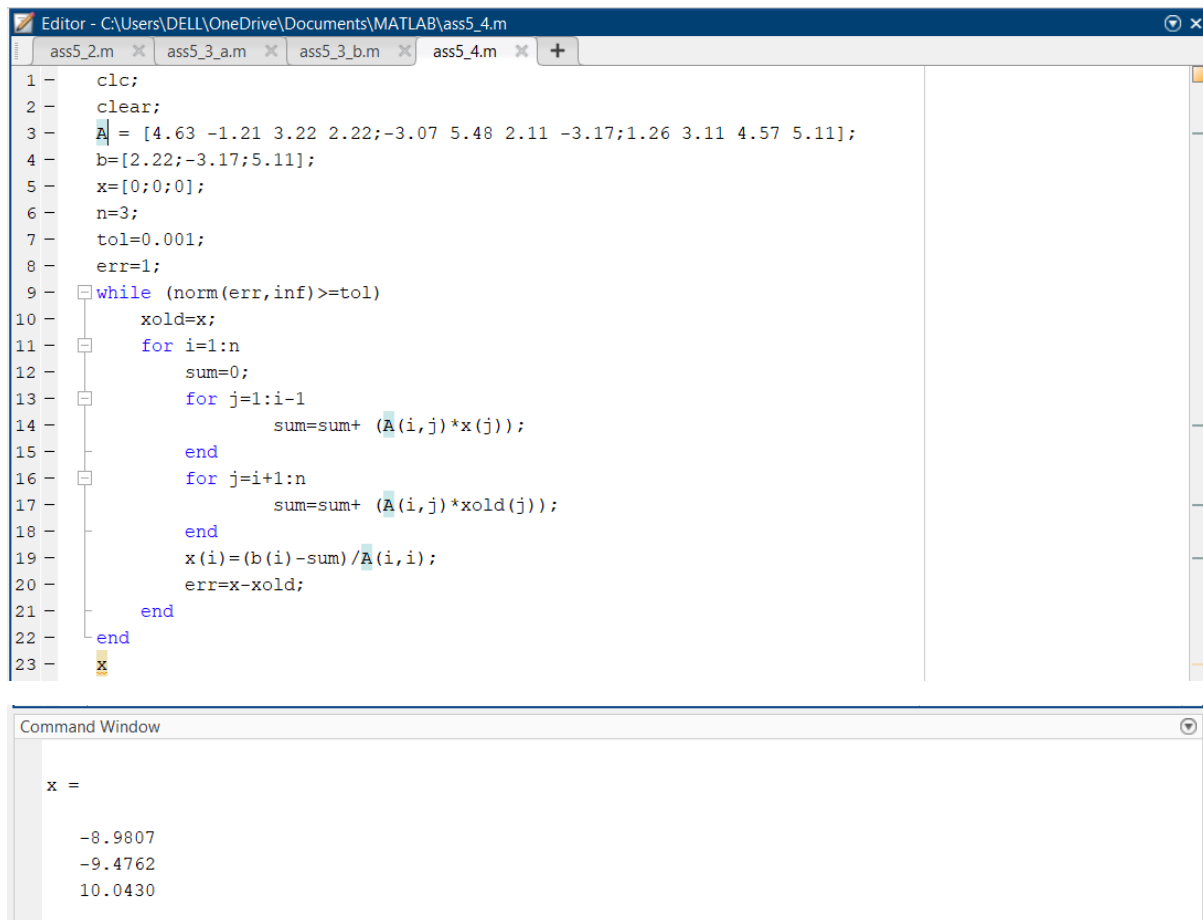
```
x =
    -0.7534
     0.0411
    -0.2808
     0.6918
```

4. Use Gauss Seidel method to solve the following linear system with an initial vector $[0,0,0]$ and tolerance value 10^{-3} in the ∞ norm:

$$4.63x_1 - 1.21x_2 + 3.22x_3 = 2.22$$

$$-3.07x_1 + 5.48x_2 + 2.11x_3 = -3.17$$

$$1.26x_1 + 3.11x_2 + 4.57x_3 = 5.11$$



The image shows a MATLAB Editor window with a script for solving a linear system using the Gauss-Seidel method. The script is as follows:

```
1 - clc;
2 - clear;
3 - A = [4.63 -1.21 3.22 2.22; -3.07 5.48 2.11 -3.17; 1.26 3.11 4.57 5.11];
4 - b = [2.22; -3.17; 5.11];
5 - x = [0; 0; 0];
6 - n = 3;
7 - tol = 0.001;
8 - err = 1;
9 - while (norm(err, inf) >= tol)
10 -     xold = x;
11 -     for i = 1:n
12 -         sum = 0;
13 -         for j = 1:i-1
14 -             sum = sum + (A(i, j) * x(j));
15 -         end
16 -         for j = i+1:n
17 -             sum = sum + (A(i, j) * xold(j));
18 -         end
19 -         x(i) = (b(i) - sum) / A(i, i);
20 -         err = x - xold;
21 -     end
22 - end
23 - x
```

The Command Window displays the final solution vector x :

```
x =
    -8.9807
    -9.4762
     10.0430
```