

```

function [A,b,x0] = Test(n)
n = 10;
A = zeros(n,n);
b = zeros(n,1);
x0 = zeros(n,1);
for i = 1:n-1
    A(i,i) = 2;
    A(i+1,i) = -0.5;
    A(i,i+1) = -0.5;
    A(n,n) = 2;
    b(1) = 1;
    b(i+1) = 2;
end
b
x0

```

b =

```

1
2
2
2
2
2
2
2
2
2
2

```

x0 =

```

0
0
0
0
0
0
0
0
0
0

```

ans =

Columns 1 through 7

2.0000	-0.5000	0	0	0	0	0
-0.5000	2.0000	-0.5000	0	0	0	0
0	-0.5000	2.0000	-0.5000	0	0	0
0	0	-0.5000	2.0000	-0.5000	0	0
0	0	0	-0.5000	2.0000	-0.5000	0
0	0	0	0	-0.5000	2.0000	-0.5000
0	0	0	0	0	-0.5000	2.0000
0	0	0	0	0	0	-0.5000
0	0	0	0	0	0	0

0 0 0 0 0 0

Columns 8 through 10

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
-0.5000	0	0
2.0000	-0.5000	0
-0.5000	2.0000	-0.5000
0	-0.5000	2.0000

---

```

function [Results,T,rho] = myjacobi(A,b,x0,tol,Niter)
[A,b,x0] = Test(50);
tol = 10^-5;
Niter = 30;
[n,~] = size(A);
X = zeros(n,Niter);
X(:,1) = x0;
D = zeros(n);
U = -triu(A);
L = -tril(A);
for i = 1:n
    D(i,i) = A(i,i);
    U(i,i) = 0;
    L(i,i) = 0;
end
Dinv = D\eye(n);
T = Dinv*(L+U);
rho = max(abs(eig(T)));
for k = 2:Niter
    X(:,k) = Dinv*b + T*X(:,k-1);
    if norm(X(:,k)-X(:,k-1),inf)/norm(X(:,k),inf) < tol
        t = X(:,1:k);
        n = size(t,2);
        Iteration = zeros(n,1);
        Residual = zeros(n,1);
        Error = zeros(n,1);
        for i = 1:n
            Iteration(i) = i;
            Residual(i) = norm(A*X(:,i)-b,inf);
            Error(i) = norm(X(:,i)-A\b);
        end
        R = table(Iteration,Residual>Error);
        Results = table(R,'VariableNames',{'Results for all Iterations'});
        return
    end
end
Results = 'N/A';
disp('Jacobi failed to converge after maximum iterations')

```

ans =

18x1 table

Results for all Iterations

Iteration	Residual	Error
-----------	----------	-------

1.0000000000000e+00	2.0000000000000e+00	1.38838897897155e+01
2.0000000000000e+00	1.0000000000000e+00	6.87989126139758e+00
3.0000000000000e+00	5.0000000000000e-01	3.41599719618496e+00
4.0000000000000e+00	2.5000000000000e-01	1.69768166775527e+00
5.0000000000000e+00	1.2500000000000e-01	8.44216750309025e-01
6.0000000000000e+00	6.2500000000000e-02	4.19989100286033e-01
7.0000000000000e+00	3.1250000000000e-02	2.09008974685922e-01
8.0000000000000e+00	1.5625000000000e-02	1.04041359442112e-01
9.0000000000000e+00	7.8125000000000e-03	5.18013514569107e-02
1.0000000000000e+01	3.9062500000000e-03	2.57961735950219e-02

1.10000000000000e+01	1.95312500000000e-03	1.28480520773979e-02
1.20000000000000e+01	9.76562500000000e-04	6.39997386963789e-03
1.30000000000000e+01	4.88281250000000e-04	3.18838612167856e-03
1.40000000000000e+01	2.44140625000000e-04	1.58858188080560e-03
1.50000000000000e+01	1.22070312500000e-04	7.91570254741955e-04
1.60000000000000e+01	6.10351562500000e-05	3.94463156211175e-04
1.70000000000000e+01	3.05175781250000e-05	1.96588043596275e-04
1.80000000000000e+01	1.52587890625000e-05	9.79802351361221e-05

---

```

function [Results,T,rho] = myjacobi(A,b,x0,tol,Niter)
[A,b,x0] = Test(100);
tol = 10^-5;
Niter = 30;
[n,~] = size(A);
X = zeros(n,Niter);
X(:,1) = x0;
D = zeros(n);
U = -triu(A);
L = -tril(A);
for i = 1:n
    D(i,i) = A(i,i);
    U(i,i) = 0;
    L(i,i) = 0;
end
Dinv = D\eye(n);
T = Dinv*(L+U);
rho = max(abs(eig(T)));
for k = 2:Niter
    X(:,k) = Dinv*b + T*X(:,k-1);
    if norm(X(:,k)-X(:,k-1),inf)/norm(X(:,k),inf) < tol
        t = X(:,1:k);
        n = size(t,2);
        Iteration = zeros(n,1);
        Residual = zeros(n,1);
        Error = zeros(n,1);
        for i = 1:n
            Iteration(i) = i;
            Residual(i) = norm(A*X(:,i)-b,inf);
            Error(i) = norm(X(:,i)-A\b);
        end
        R = table(Iteration,Residual>Error);
        Results = table(R,'VariableNames',{'Results for all Iterations'});
        return
    end
end
Results = 'N/A';
disp('Jacobi failed to converge after maximum iterations')

```

ans =

18x1 table

Results for all Iterations

Iteration	Residual	Error
-----------	----------	-------

1.0000000000000e+00	2.0000000000000e+00	1.98182339196248e+01
2.0000000000000e+00	1.0000000000000e+00	9.86574395414024e+00
3.0000000000000e+00	5.0000000000000e-01	4.91620146498733e+00
4.0000000000000e+00	2.5000000000000e-01	2.45094329698431e+00
5.0000000000000e+00	1.2500000000000e-01	1.22227325975100e+00
6.0000000000000e+00	6.2500000000000e-02	6.09674785733404e-01
7.0000000000000e+00	3.1250000000000e-02	3.04159294612643e-01
8.0000000000000e+00	1.5625000000000e-02	1.51761772935620e-01
9.0000000000000e+00	7.8125000000000e-03	7.57306927557276e-02
1.0000000000000e+01	3.9062500000000e-03	3.77939416476968e-02

1.10000000000000e+01	1.95312500000000e-03	1.88628551779624e-02
1.20000000000000e+01	9.76562500000000e-04	9.41506140990925e-03
1.30000000000000e+01	4.88281250000000e-04	4.69965264844027e-03
1.40000000000000e+01	2.44140625000000e-04	2.34602315222888e-03
1.50000000000000e+01	1.22070312500000e-04	1.17117100710566e-03
1.60000000000000e+01	6.10351562500000e-05	5.84692822370271e-04
1.70000000000000e+01	3.05175781250000e-05	2.91912636958136e-04
1.80000000000000e+01	1.52587890625000e-05	1.45745184002962e-04

---

Published with MATLAB® R2021b

```

function [Results,T,rho] = myjacobi(A,b,x0,tol,Niter)
[A,b,x0] = Test(200);
tol = 10^-5;
Niter = 30;
[n,~] = size(A);
X = zeros(n,Niter);
X(:,1) = x0;
D = zeros(n);
U = -triu(A);
L = -tril(A);
for i = 1:n
    D(i,i) = A(i,i);
    U(i,i) = 0;
    L(i,i) = 0;
end
Dinv = D\eye(n);
T = Dinv*(L+U);
rho = max(abs(eig(T)));
for k = 2:Niter
    X(:,k) = Dinv*b + T*X(:,k-1);
    if norm(X(:,k)-X(:,k-1),inf)/norm(X(:,k),inf) < tol
        t = X(:,1:k);
        n = size(t,2);
        Iteration = zeros(n,1);
        Residual = zeros(n,1);
        Error = zeros(n,1);
        for i = 1:n
            Iteration(i) = i;
            Residual(i) = norm(A*X(:,i)-b,inf);
            Error(i) = norm(X(:,i)-A\b);
        end
        R = table(Iteration,Residual>Error);
        Results = table(R,'VariableNames',{'Results for all Iterations'});
        return
    end
end
Results = 'N/A';
disp('Jacobi failed to converge after maximum iterations')

```

ans =

18x1 table

Results for all Iterations

Iteration	Residual	Error
-----------	----------	-------

1.0000000000000e+00	2.0000000000000e+00	2.81560365764247e+01
2.0000000000000e+00	1.0000000000000e+00	1.40475230474506e+01
3.0000000000000e+00	5.0000000000000e-01	7.01206366516616e+00
4.0000000000000e+00	2.5000000000000e-01	3.50101742998122e+00
5.0000000000000e+00	1.2500000000000e-01	1.74827112356818e+00
6.0000000000000e+00	6.2500000000000e-02	8.73114164562156e-01
7.0000000000000e+00	3.1250000000000e-02	4.36083852600920e-01
8.0000000000000e+00	1.5625000000000e-02	2.17820334736137e-01
9.0000000000000e+00	7.8125000000000e-03	1.08805576374846e-01
1.0000000000000e+01	3.9062500000000e-03	5.43531133562698e-02

1.10000000000000e+01	1.95312500000000e-03	2.71528457445492e-02
1.20000000000000e+01	9.76562500000000e-04	1.35650585326045e-02
1.30000000000000e+01	4.88281250000000e-04	6.77706373927145e-03
1.40000000000000e+01	2.44140625000000e-04	3.38589561391561e-03
1.50000000000000e+01	1.22070312500000e-04	1.69167303202175e-03
1.60000000000000e+01	6.10351562500000e-05	8.45218744691281e-04
1.70000000000000e+01	3.05175781250000e-05	4.22309418646329e-04
1.80000000000000e+01	1.52587890625000e-05	2.11008822150744e-04

---

Published with MATLAB® R2021b

```

function [Results,T,rho] = mygauseidel(A,b,x0,tol,Niter)
[A,b,x0] = Test(50);
tol = 10^-5;
Niter = 30;
[n,~] = size(A);
X = zeros(n,Niter);
X(:,1) = x0;
D = zeros(n);
U = -triu(A);
L = -tril(A);
for i = 1:n
    D(i,i) = A(i,i);
    U(i,i) = 0;
    L(i,i) = 0;
end
Dinv = (D-L)\eye(n);
T = Dinv*U;
rho = max(abs(eig(T)));
for k = 2:Niter
    X(:,k) = Dinv*b + T*X(:,k-1);
    if norm(X(:,k)-X(:,k-1),inf)/norm(X(:,k),inf) < tol
        t = X(:,1:k);
        n = size(t,2);
        Iteration = zeros(n,1);
        Residual = zeros(n,1);
        Error = zeros(n,1);
        for i = 1:n
            Iteration(i) = i;
            Residual(i) = norm(A*X(:,i)-b,inf);
            Error(i) = norm(X(:,i)-A\b);
        end
        R = table(Iteration,Residual>Error);
        Results = table(R,'VariableNames',{'Results for all Iterations'});
        return
    end
end
Results = 'N/A';
disp('Gauss-Seidel failed to converge after maximum iterations')

```

ans =

13x1 table

Results for all Iterations

Iteration	Residual	Error
-----------	----------	-------

1.0000000000000e+00	2.0000000000000e+00	1.38838897897155e+01
2.0000000000000e+00	6.6666666666667e-01	4.59876788629031e+00
3.0000000000000e+00	2.2222222222222e-01	1.52041181248428e+00
4.0000000000000e+00	7.40740740740744e-02	5.02584669479238e-01
5.0000000000000e+00	2.46913580246915e-02	1.66127770865071e-01
6.0000000000000e+00	8.23045267489730e-03	5.49110088516634e-02
7.0000000000000e+00	2.74348422496606e-03	1.81490105834213e-02
8.0000000000000e+00	9.14494741655503e-04	5.99812042716191e-03
9.0000000000000e+00	3.04831580552056e-04	1.98216402416572e-03
1.0000000000000e+01	1.01610526850981e-04	6.54968887001546e-04

1.10000000000000e+01	3.38701756172899e-05	2.16398205260611e-04
1.20000000000000e+01	1.12900585391706e-05	7.14881877222995e-05
1.30000000000000e+01	3.76335284690832e-06	2.36134072042124e-05

---

Published with MATLAB® R2021b

```

function [Results,T,rho] = mygauseidel(A,b,x0,tol,Niter)
[A,b,x0] = Test(100);
tol = 10^-5;
Niter = 30;
[n,~] = size(A);
X = zeros(n,Niter);
X(:,1) = x0;
D = zeros(n);
U = -triu(A);
L = -tril(A);
for i = 1:n
    D(i,i) = A(i,i);
    U(i,i) = 0;
    L(i,i) = 0;
end
Dinv = (D-L)\eye(n);
T = Dinv*U;
rho = max(abs(eig(T)));
for k = 2:Niter
    X(:,k) = Dinv*b + T*X(:,k-1);
    if norm(X(:,k)-X(:,k-1),inf)/norm(X(:,k),inf) < tol
        t = X(:,1:k);
        n = size(t,2);
        Iteration = zeros(n,1);
        Residual = zeros(n,1);
        Error = zeros(n,1);
        for i = 1:n
            Iteration(i) = i;
            Residual(i) = norm(A*X(:,i)-b,inf);
            Error(i) = norm(X(:,i)-A\b);
        end
        R = table(Iteration,Residual>Error);
        Results = table(R,'VariableNames',{'Results for all Iterations'});
        return
    end
end
Results = 'N/A';
disp('Gauss-Seidel failed to converge after maximum iterations')

```

ans =

13x1 table

Results for all Iterations

Iteration	Residual	Error
-----------	----------	-------

1.0000000000000e+00	2.0000000000000e+00	1.98182339196248e+01
2.0000000000000e+00	6.6666666666667e-01	6.58565777232596e+00
3.0000000000000e+00	2.2222222222222e-01	2.18650128790515e+00
4.0000000000000e+00	7.40740740740744e-02	7.25906173339314e-01
5.0000000000000e+00	2.46913580246915e-02	2.41001233000523e-01
6.0000000000000e+00	8.23045267489730e-03	8.00139766215484e-02
7.0000000000000e+00	2.74348422496606e-03	2.65654262113177e-02
8.0000000000000e+00	9.14494741655503e-04	8.82000455127531e-03
9.0000000000000e+00	3.04831580552056e-04	2.92832526263460e-03
1.0000000000000e+01	1.01610526850981e-04	9.72223843199042e-04

1.10000000000000e+01	3.38701756172899e-05	3.22781076050997e-04
1.20000000000000e+01	1.12900585391706e-05	1.07162643093285e-04
1.30000000000000e+01	3.76335284690832e-06	3.55771588543805e-05

---

Published with MATLAB® R2021b

```

function [Results,T,rho] = mygauseidel(A,b,x0,tol,Niter)
[A,b,x0] = Test(200);
tol = 10^-5;
Niter = 30;
[n,~] = size(A);
X = zeros(n,Niter);
X(:,1) = x0;
D = zeros(n);
U = -triu(A);
L = -tril(A);
for i = 1:n
    D(i,i) = A(i,i);
    U(i,i) = 0;
    L(i,i) = 0;
end
Dinv = (D-L)\eye(n);
T = Dinv*U;
rho = max(abs(eig(T)));
for k = 2:Niter
    X(:,k) = Dinv*b + T*X(:,k-1);
    if norm(X(:,k)-X(:,k-1),inf)/norm(X(:,k),inf) < tol
        t = X(:,1:k);
        n = size(t,2);
        Iteration = zeros(n,1);
        Residual = zeros(n,1);
        Error = zeros(n,1);
        for i = 1:n
            Iteration(i) = i;
            Residual(i) = norm(A*X(:,i)-b,inf);
            Error(i) = norm(X(:,i)-A\b);
        end
        R = table(Iteration,Residual>Error);
        Results = table(R,'VariableNames',{'Results for all Iterations'});
        return
    end
end
Results = 'N/A';
disp('Gauss-Seidel failed to converge after maximum iterations')

```

ans =

13x1 table

Results for all Iterations

Iteration	Residual	Error
-----------	----------	-------

1.0000000000000e+00	2.0000000000000e+00	2.81560365764247e+01
2.0000000000000e+00	6.6666666666667e-01	9.37098355236214e+00
3.0000000000000e+00	2.2222222222222e-01	3.11754061512423e+00
4.0000000000000e+00	7.40740740740744e-02	1.03712902644043e+00
5.0000000000000e+00	2.46913580246915e-02	3.45033201906892e-01
6.0000000000000e+00	8.23045267489730e-03	1.14787941780337e-01
7.0000000000000e+00	2.74348422496606e-03	3.81889046589793e-02
8.0000000000000e+00	9.14494741655503e-04	1.27052171778397e-02
9.0000000000000e+00	3.04831580552056e-04	4.22697505244080e-03
1.0000000000000e+01	1.01610526850981e-04	1.40630335204521e-03

1.10000000000000e+01	3.38701756172899e-05	4.67874451845515e-04
1.20000000000000e+01	1.12900585391706e-05	1.55661087804656e-04
1.30000000000000e+01	3.76335284690832e-06	5.17881907083929e-05

---

Published with MATLAB® R2021b

```

function [T,rho,w_opt,values] = mySOR(A)
[A] = Test(20);
[n,~] = size(A);
D = zeros(n);
U = -triu(A);
L = -tril(A);
for i = 1:n
    D(i,i) = A(i,i);
    U(i,i) = 0;
    L(i,i) = 0;
end
W = zeros(21,1);
for w = 0:20
    Dinv = (D-w/10*L)\eye(n);
    T = Dinv*((1-w/10)*D+w/10*U);
    W(w+1) = max(abs(eig(T)));
end
[rho,I] = min(W);
w_opt = (I-1)/10
x = 0:0.1:2;
plot(x,W)
Dinv = (D-w_opt*L)\eye(n);
T = Dinv*((1-w_opt)*D+w_opt*U);
Omega = zeros(21,1);
Rho = zeros(21,1);
for i = 1:21
    Omega(i) = (i-1)/10;
    Rho(i) = W(i);
end
values = table(Omega,Rho)

```

w\_opt =  
1.10000000000000e+00

values =

21×2 table

Omega	Rho
0.00000000000000e+00	1.00000000000000e+00
1.00000000000000e-01	9.48142519234887e-01
2.00000000000000e-01	8.93467670317662e-01
3.00000000000000e-01	8.35583956261145e-01
4.00000000000000e-01	7.73987914370515e-01
5.00000000000000e-01	7.08008589333076e-01
6.00000000000000e-01	6.36708313913634e-01
7.00000000000000e-01	5.58686782844217e-01
8.00000000000000e-01	4.71634361971493e-01
9.00000000000000e-01	3.71051294888880e-01
1.00000000000000e+00	2.44446600723268e-01
1.10000000000000e+00	1.0000000023006e-01
1.20000000000000e+00	2.0000000000182e-01
1.30000000000000e+00	3.000000000002e-01

1.40000000000000e+00	4.0000000000012e-01
1.50000000000000e+00	5.000000000007e-01
1.60000000000000e+00	6.000000000002e-01
1.70000000000000e+00	7.000000000001e-01
1.80000000000000e+00	8.000000000001e-01
1.90000000000000e+00	9.000000000000e-01
2.00000000000000e+00	1.000000000000e+00

ans =

Columns 1 through 3

-1.00000000000001e-01	2.75000000000000e-01	0
-2.75000000000003e-02	-2.43750000000008e-02	2.75000000000000e-01
-7.56250000000008e-03	-6.703125000000021e-03	-2.43750000000008e-02
-2.07968750000003e-03	-1.84335937500005e-03	-6.703125000000021e-03
-5.71914062500007e-04	-5.069238281250017e-04	-1.84335937500005e-03
-1.572763671875002e-04	-1.394040527343755e-04	-5.069238281250017e-04
-4.325100097656256e-05	-3.833611450195325e-05	-1.394040527343755e-04
-1.189402526855471e-05	-1.054243148803715e-05	-3.833611450195325e-05
-3.270856948852544e-06	-2.899168659210217e-06	-1.054243148803715e-05
-8.994856609344498e-07	-7.972713812828095e-07	-2.899168659210217e-06
-2.473585567569737e-07	-2.192496298527726e-07	-7.972713812828095e-07
-6.802360310816778e-08	-6.029364820951247e-08	-2.192496298527726e-07
-1.870649085474614e-08	-1.658075325761593e-08	-6.029364820951247e-08
-5.144284985055189e-09	-4.559707145844382e-09	-1.658075325761593e-08
-1.414678370890177e-09	-1.253919465107206e-09	-4.559707145844382e-09
-3.890365519947987e-10	-3.448278529044816e-10	-1.253919465107206e-09
-1.069850517985696e-10	-9.482765954873243e-11	-3.448278529044816e-10
-2.942088924460665e-11	-2.607760637590141e-11	-9.482765954873243e-11
-8.090744542266830e-12	-7.171341753372891e-12	-2.607760637590141e-11
-2.224954749123378e-12	-1.972118982177545e-12	-7.171341753372891e-12

Columns 4 through 6

0	0	0
0	0	0
2.75000000000000e-01	0	0
-2.43750000000008e-02	2.75000000000000e-01	0
-6.703125000000021e-03	-2.43750000000008e-02	2.75000000000000e-01
-1.84335937500005e-03	-6.703125000000021e-03	-2.43750000000008e-02
-5.069238281250017e-04	-1.84335937500005e-03	-6.703125000000021e-03
-1.394040527343755e-04	-5.069238281250017e-04	-1.84335937500005e-03
-3.833611450195325e-05	-1.394040527343755e-04	-5.069238281250017e-04
-1.054243148803715e-05	-3.833611450195325e-05	-1.394040527343755e-04
-2.899168659210217e-06	-1.054243148803715e-05	-3.833611450195325e-05
-7.972713812828095e-07	-2.899168659210217e-06	-1.054243148803715e-05
-2.192496298527726e-07	-7.972713812828095e-07	-2.899168659210217e-06
-6.029364820951247e-08	-2.192496298527726e-07	-7.972713812828095e-07
-1.658075325761593e-08	-6.029364820951247e-08	-2.192496298527726e-07
-4.559707145844382e-09	-1.658075325761593e-08	-6.029364820951247e-08
-1.253919465107206e-09	-4.559707145844383e-09	-1.658075325761593e-08
-3.448278529044816e-10	-1.253919465107206e-09	-4.559707145844383e-09
-9.482765954873243e-11	-3.448278529044816e-10	-1.253919465107206e-09
-2.607760637590141e-11	-9.482765954873243e-11	-3.448278529044816e-10

Columns 7 through 9

0	0	0
0	0	0

0	0	0
0	0	0
0	0	0
2.75000000000000e-01	0	0
-2.43750000000008e-02	2.75000000000000e-01	0
-6.703125000000021e-03	-2.43750000000008e-02	2.75000000000000e-01
-1.84335937500005e-03	-6.703125000000021e-03	-2.43750000000008e-02
-5.069238281250017e-04	-1.84335937500005e-03	-6.703125000000021e-03
-1.394040527343755e-04	-5.069238281250017e-04	-1.84335937500005e-03
-3.833611450195325e-05	-1.394040527343755e-04	-5.069238281250017e-04
-1.054243148803715e-05	-3.833611450195325e-05	-1.394040527343755e-04
-2.899168659210217e-06	-1.054243148803715e-05	-3.833611450195325e-05
-7.972713812828095e-07	-2.899168659210217e-06	-1.054243148803715e-05
-2.192496298527726e-07	-7.972713812828095e-07	-2.899168659210217e-06
-6.029364820951248e-08	-2.192496298527726e-07	-7.972713812828093e-07
-1.658075325761593e-08	-6.029364820951248e-08	-2.192496298527726e-07
-4.559707145844383e-09	-1.658075325761593e-08	-6.029364820951248e-08
-1.253919465107206e-09	-4.559707145844383e-09	-1.658075325761593e-08

Columns 10 through 12

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
2.75000000000000e-01	0	0
-2.43750000000008e-02	2.75000000000000e-01	0
-6.703125000000021e-03	-2.43750000000008e-02	2.75000000000000e-01
-1.84335937500005e-03	-6.703125000000021e-03	-2.43750000000008e-02
-5.069238281250017e-04	-1.84335937500005e-03	-6.703125000000021e-03
-1.394040527343755e-04	-5.069238281250017e-04	-1.84335937500005e-03
-3.833611450195325e-05	-1.394040527343755e-04	-5.069238281250017e-04
-1.054243148803715e-05	-3.833611450195325e-05	-1.394040527343755e-04
-2.899168659210217e-06	-1.054243148803715e-05	-3.833611450195325e-05
-7.972713812828093e-07	-2.899168659210217e-06	-1.054243148803715e-05
-2.192496298527726e-07	-7.972713812828093e-07	-2.899168659210217e-06
-6.029364820951248e-08	-2.192496298527726e-07	-7.972713812828093e-07

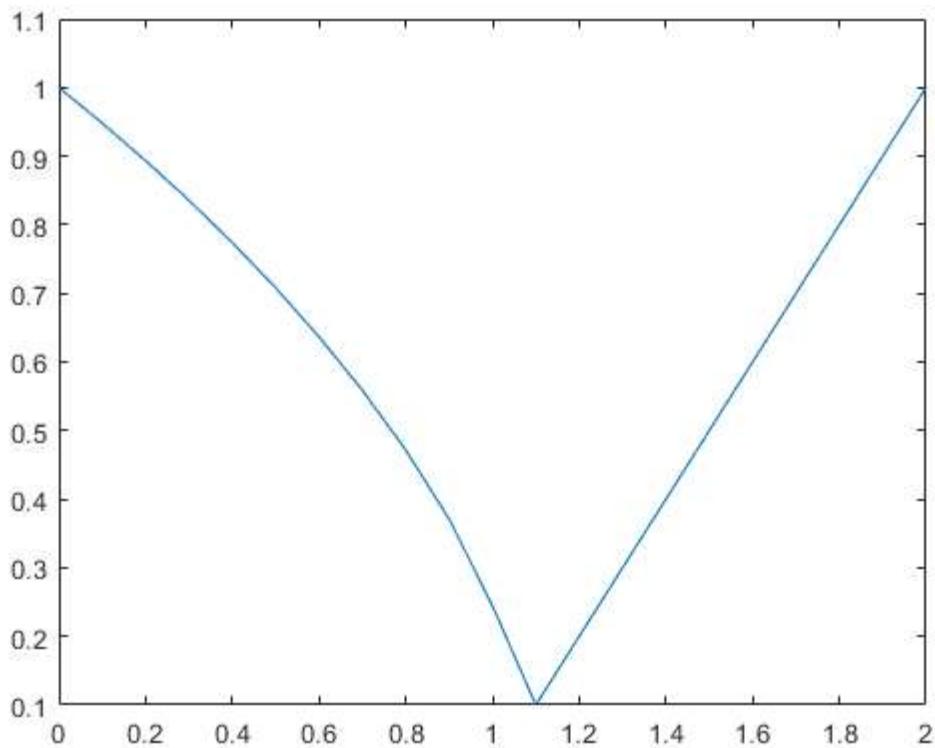
Columns 13 through 15

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
2.75000000000000e-01	0	0
-2.43750000000008e-02	2.75000000000000e-01	0
-6.703125000000021e-03	-2.43750000000008e-02	2.75000000000000e-01
-1.84335937500005e-03	-6.703125000000021e-03	-2.43750000000008e-02
-5.069238281250017e-04	-1.84335937500005e-03	-6.703125000000021e-03
-1.394040527343754e-04	-5.069238281250018e-04	-1.84335937500005e-03

$$\begin{aligned} -3.833611450195325e-05 & \quad -1.394040527343754e-04 & -5.069238281250018e-04 \\ -1.054243148803715e-05 & \quad -3.833611450195325e-05 & -1.394040527343754e-04 \\ -2.899168659210217e-06 & \quad -1.054243148803715e-05 & -3.833611450195325e-05 \end{aligned}$$

Columns 16 through 18

Columns 19 through 20



---

Published with MATLAB® R2021b

### Numerical Analysis Project 3:

1a.) Using the Jacobi method on the linear test system for N = 50,100,200, we get the following rho values.

$$\rho(T_J(50)) = 0.49905166 < 1$$

$$\rho(T_J(100)) = 0.49975814 < 1$$

$$\rho(T_J(200)) = 0.49993893 < 1$$

Since each of these values is less than one, the Jacobi matrix will converge for each N value. We can also note that given any integers N<sub>1</sub> and N<sub>2</sub>, if N<sub>1</sub> < N<sub>2</sub>, then  $\rho(T_J(N_1)) < \rho(T_J(N_2))$ , which implies that the Jacobi matrix will converge faster for lesser N values.

1b.) Although we have the same number of iterations for each N value, the results from myjacobi are to be expected since the error term is smaller for lesser values of N. This is consistent with our previous result that lesser N values converge faster.

2a.) Using the Gauss-Seidel method on the linear test system for N = 50,100,200, we get the following rho values.

$$\rho(T_{GS}(50)) = 0.24905256 < 1$$

$$\rho(T_{GS}(100)) = 0.24976105 < 1$$

$$\rho(T_{GS}(200)) = 0.25055881 < 1$$

Similar to the Jacobi matrix, since all of these rho values are less than one, the Gauss-Seidel matrix will converge for each N value. We can also note that given any integers N<sub>1</sub> and N<sub>2</sub>, if N<sub>1</sub> < N<sub>2</sub>, then  $\rho(T_{GS}(N_1)) < \rho(T_{GS}(N_2))$ , which implies that the Gauss-Seidel matrix will converge faster for lesser N values. We can also note that the Gauss-Seidel iteration matrices have smaller rho values than the Jacobi matrices, so we can conclude that the Gauss-Seidel method will converge faster.

2b.) Once again, although we have the same number of iterations for each N value, the results from mygauseidel are to be expected since the error term is smaller for lesser values of N. This is consistent with our previous result that lesser N values converge faster. Also, compared to the Jacobi method, Gauss-Seidel took fewer iterations for the same N values, which is once again consistent with the relative values of rho.

3c.) Plotting rho(w) versus w gets us a V-shaped graph that represents the efficiency of the SOR method with respect to the relaxation factor. To maximize the efficiency of the SOR method, we want to take the minimum value of rho(w). For our linear test case with N = 20, rho(w) reaches a minimum of approximately 0.1, when w = 1.1.

```

function [Results,T,w_opt,rho] = completeSOR(A,b,x0,tol,Niter)
[A,b,x0] = Test(50);
tol = 10^-10;
Niter = 20;
[n,~] = size(A);
X = zeros(n,Niter);
X(:,1) = x0;
D = zeros(n);
U = -triu(A);
L = -tril(A);
for i = 1:n
    D(i,i) = A(i,i);
    U(i,i) = 0;
    L(i,i) = 0;
end
W = zeros(20001,1);
for w = 0:20000
    Dinv = (D-w/10000*L)\eye(n);
    T = Dinv*((1-w/10000)*D+w/10000*U);
    W(w+1) = max(abs(eig(T)));
end
[rho,I] = min(W);
w_opt = (I-1)/10000
rho
x = 0:0.0001:2;
plot(x,W)
Dinv = (D-w_opt*L)\eye(n);
T = Dinv*((1-w_opt)*D+w_opt*U)
for k = 2:Niter
    X(:,k) = Dinv*w*b + T*X(:,k-1);
    if norm(X(:,k)-X(:,k-1),inf)/norm(X(:,k),inf) < tol
        t = X(:,1:k);
        n = size(t,2);
        Iteration = zeros(n,1);
        Residual = zeros(n,1);
        Error = zeros(n,1);
        for i = 1:n
            Iteration(i) = i;
            Residual(i) = norm(A*X(:,i)-b,inf);
            Error(i) = norm(X(:,i)-A\b);
        end
        R = table(Iteration,Residual>Error);
        Results = table(R,'VariableNames',{'Results for all Iterations'});
        return
    end
end
Results = 'N/A';
disp('SOR failed to converge after maximum iterations')

```

w\_opt =  
1.07190000000000e+00

rho =  
8.249802538362505e-02

T =

Columns 1 through 3

-7.190000000000007e-02	2.679750000000000e-01	0
-1.926740250000002e-02	-8.939937500006268e-05	2.679750000000000e-01
-5.163182184937506e-03	-2.395679751564036e-05	-8.939937500006268e-05
-1.383603746008629e-03	-6.419822814254244e-06	-2.395679751564036e-05
-3.707712138366622e-04	-1.720352018649796e-06	-6.419822814254244e-06
-9.935741602787956e-05	-4.610113321976784e-07	-1.720352018649796e-06
-2.662530356007103e-05	-1.235395117456799e-07	-4.610113321976784e-07
-7.134915721510034e-06	-3.310550066004786e-08	-1.235395117456799e-07
-1.911979040471652e-06	-8.871446539376254e-09	-3.310550066004786e-08
-5.123625833703908e-07	-2.377325886389375e-09	-8.871446539376254e-09
-1.373003632786805e-07	-6.370639044052446e-10	-2.377325886389375e-09
-3.679306484960440e-08	-1.707171997829940e-10	-6.370639044052446e-10
-9.859621553072741e-09	-4.574794161184361e-11	-1.707171997829940e-10
-2.642132085684668e-09	-1.225930465343307e-11	-4.574794161184361e-11
-7.080253456613489e-10	-3.285187164503634e-12	-1.225930465343307e-11
-1.897330920036000e-10	-8.803480304078765e-13	-3.285187164503634e-12
-5.084372532966471e-11	-2.359112634485593e-13	-8.803480304078765e-13
-1.362484729521690e-11	-6.321832082262964e-14	-2.359112634485593e-13
-3.651118453935750e-12	-1.694092952244400e-14	-6.321832082262964e-14
-9.784084676934327e-13	-4.539745588776551e-15	-1.694092952244400e-14
-2.621890091301476e-13	-1.216538324152392e-15	-4.539745588776551e-15
-7.026009972165132e-14	-3.260018574147609e-16	-1.216538324152392e-15
-1.882795022290951e-14	-8.736034774071704e-17	-3.260018574147609e-16
-5.045419960984177e-15	-2.341038918581727e-17	-8.736034774071704e-17
-1.352046414044735e-15	-6.273399042070083e-18	-2.341038918581727e-17
-3.623146378036379e-16	-1.681114108298834e-18	-6.273399042070083e-18
-9.709126506542987e-17	-4.504965531713658e-19	-1.681114108298834e-18
-2.601803175590857e-17	-1.207218138360967e-19	-4.504965531713658e-19
-6.972182059789600e-18	-3.235042806272687e-20	-1.207218138360967e-19
-1.868370487472118e-18	-8.669105960109478e-21	-3.235042806272687e-20
-5.006765813803409e-19	-2.323103669660220e-21	-8.669105960109478e-21
-1.341688068953969e-19	-6.225337058771261e-22	-2.323103669660220e-21
-3.595388602779398e-20	-1.668234698324298e-22	-6.225337058771261e-22
-9.634742608298092e-21	-4.470451932834690e-23	-1.668234698324298e-22
-2.581870150458682e-21	-1.197969356701343e-23	-4.470451932834690e-23
-6.918766535691654e-22	-3.210258383620587e-24	-1.197969356701343e-23
-1.854056462401971e-22	-8.602689903507339e-25	-3.210258383620587e-24
-4.968407805121682e-23	-2.305305826892380e-25	-8.602689903507339e-25
-1.331409081577483e-23	-6.177643289615351e-26	-2.305305826892380e-25
-3.567843486357259e-24	-1.655453960534754e-26	-6.177643289615351e-26
-9.560928582565866e-25	-4.436202750742578e-27	-1.655453960534754e-26
-2.562089836913089e-25	-1.188791432130132e-27	-4.436202750742578e-27
-6.865760240467850e-26	-3.185663840251117e-28	-1.188791432130132e-27
-1.839852100439372e-26	-8.536782675912768e-29	-3.185663840251117e-28
-4.930343666152408e-27	-2.287644337577472e-29	-8.536782675912768e-29
-1.321208843937192e-27	-6.130314913623660e-30	-2.287644337577472e-29
-3.540509399540689e-28	-1.642771138978433e-30	-6.130314913623660e-30
-9.487680063419163e-29	-4.402215959677392e-31	-1.642771138978433e-30
-2.542461064994751e-29	-1.179683821794592e-31	-4.402215959677371e-31
-6.813160038919682e-30	-3.161257721454195e-32	-1.179683821794592e-31

Columns 4 through 6

0	0	0
0	0	0

2.679750000000000e-01	0	0
-8.939937500006268e-05	2.679750000000000e-01	0
-2.395679751564036e-05	-8.939937500006268e-05	2.679750000000000e-01
-6.419822814254244e-06	-2.395679751564036e-05	-8.939937500006268e-05
-1.720352018649796e-06	-6.419822814254244e-06	-2.395679751564036e-05
-4.610113321976784e-07	-1.720352018649796e-06	-6.419822814254244e-06
-1.235395117456799e-07	-4.610113321976784e-07	-1.720352018649796e-06
-3.310550066004786e-08	-1.235395117456799e-07	-4.610113321976784e-07
-8.871446539376254e-09	-3.310550066004786e-08	-1.235395117456799e-07
-2.377325886389375e-09	-8.871446539376254e-09	-3.310550066004786e-08
-6.370639044052446e-10	-2.377325886389375e-09	-8.871446539376254e-09
-1.707171997829940e-10	-6.370639044052446e-10	-2.377325886389375e-09
-4.574794161184361e-11	-1.707171997829940e-10	-6.370639044052446e-10
-1.225930465343307e-11	-4.574794161184361e-11	-1.707171997829940e-10
-3.285187164503634e-12	-1.225930465343307e-11	-4.574794161184361e-11
-8.803480304078765e-13	-3.285187164503634e-12	-1.225930465343307e-11
-2.359112634485593e-13	-8.803480304078765e-13	-3.285187164503634e-12
-6.321832082262964e-14	-2.359112634485593e-13	-8.803480304078765e-13
-1.694092952244400e-14	-6.321832082262964e-14	-2.359112634485593e-13
-4.539745588776551e-15	-1.694092952244400e-14	-6.321832082262964e-14
-1.216538324152392e-15	-4.539745588776551e-15	-1.694092952244400e-14
-3.260018574147609e-16	-1.216538324152392e-15	-4.539745588776551e-15
-8.736034774071704e-17	-3.260018574147609e-16	-1.216538324152392e-15
-2.341038918581727e-17	-8.736034774071704e-17	-3.260018574147609e-16
-6.273399042070083e-18	-2.341038918581727e-17	-8.736034774071704e-17
-1.681114108298834e-18	-6.273399042070083e-18	-2.341038918581727e-17
-4.504965531713658e-19	-1.681114108298834e-18	-6.273399042070083e-18
-1.207218138360967e-19	-4.504965531713658e-19	-1.681114108298834e-18
-3.235042806272687e-20	-1.207218138360967e-19	-4.504965531713658e-19
-8.669105960109478e-21	-3.235042806272687e-20	-1.207218138360967e-19
-2.323103669660220e-21	-8.669105960109478e-21	-3.235042806272687e-20
-6.225337058771261e-22	-2.323103669660220e-21	-8.669105960109478e-21
-1.668234698324298e-22	-6.225337058771261e-22	-2.323103669660220e-21
-4.470451932834690e-23	-1.668234698324298e-22	-6.225337058771261e-22
-1.197969356701343e-23	-4.470451932834690e-23	-1.668234698324298e-22
-3.210258383620587e-24	-1.197969356701343e-23	-4.470451932834690e-23
-8.602689903507339e-25	-3.210258383620587e-24	-1.197969356701343e-23
-2.305305826892380e-25	-8.602689903507339e-25	-3.210258383620587e-24
-6.177643289615351e-26	-2.305305826892380e-25	-8.602689903507339e-25
-1.655453960534754e-26	-6.177643289615351e-26	-2.305305826892380e-25
-4.436202750742578e-27	-1.655453960534754e-26	-6.177643289615351e-26
-1.188791432130132e-27	-4.436202750742578e-27	-1.655453960534754e-26
-3.185663840251117e-28	-1.188791432130132e-27	-4.436202750742578e-27
-8.536782675912768e-29	-3.185663840251117e-28	-1.188791432130132e-27
-2.287644337577472e-29	-8.536782675912768e-29	-3.185663840251117e-28
-6.130314913623660e-30	-2.287644337577472e-29	-8.536782675912768e-29
-1.642771138978345e-30	-6.130314913623592e-30	-2.287644337577518e-29
-4.402215959677371e-31	-1.642771138978345e-30	-6.130314913623592e-30

Columns 7 through 9

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
2.679750000000000e-01	0	0
-8.939937500006268e-05	2.679750000000000e-01	0
-2.395679751564036e-05	-8.939937500006268e-05	2.679750000000000e-01
-6.419822814254244e-06	-2.395679751564036e-05	-8.939937500006268e-05
-1.720352018649796e-06	-6.419822814254244e-06	-2.395679751564036e-05

-4.610113321976784e-07	-1.720352018649796e-06	-6.419822814254244e-06
-1.235395117456799e-07	-4.610113321976784e-07	-1.720352018649796e-06
-3.310550066004786e-08	-1.235395117456799e-07	-4.610113321976784e-07
-8.871446539376254e-09	-3.310550066004786e-08	-1.235395117456799e-07
-2.377325886389375e-09	-8.871446539376254e-09	-3.310550066004786e-08
-6.370639044052446e-10	-2.377325886389375e-09	-8.871446539376254e-09
-1.707171997829940e-10	-6.370639044052446e-10	-2.377325886389375e-09
-4.574794161184361e-11	-1.707171997829940e-10	-6.370639044052446e-10
-1.225930465343307e-11	-4.574794161184361e-11	-1.707171997829940e-10
-3.285187164503634e-12	-1.225930465343307e-11	-4.574794161184361e-11
-8.803480304078765e-13	-3.285187164503634e-12	-1.225930465343307e-11
-2.359112634485593e-13	-8.803480304078765e-13	-3.285187164503634e-12
-6.321832082262964e-14	-2.359112634485593e-13	-8.803480304078765e-13
-1.694092952244400e-14	-6.321832082262964e-14	-2.359112634485593e-13
-4.539745588776551e-15	-1.694092952244400e-14	-6.321832082262964e-14
-1.216538324152392e-15	-4.539745588776551e-15	-1.694092952244400e-14
-3.260018574147609e-16	-1.216538324152392e-15	-4.539745588776551e-15
-8.736034774071704e-17	-3.260018574147609e-16	-1.216538324152392e-15
-2.341038918581727e-17	-8.736034774071704e-17	-3.260018574147609e-16
-6.273399042070083e-18	-2.341038918581727e-17	-8.736034774071704e-17
-1.681114108298834e-18	-6.273399042070083e-18	-2.341038918581727e-17
-4.504965531713658e-19	-1.681114108298834e-18	-6.273399042070083e-18
-1.207218138360967e-19	-4.504965531713658e-19	-1.681114108298834e-18
-3.235042806272687e-20	-1.207218138360967e-19	-4.504965531713658e-19
-8.669105960109478e-21	-3.235042806272687e-20	-1.207218138360967e-19
-2.323103669660220e-21	-8.669105960109478e-21	-3.235042806272687e-20
-6.225337058771261e-22	-2.323103669660220e-21	-8.669105960109478e-21
-1.668234698324298e-22	-6.225337058771261e-22	-2.323103669660220e-21
-4.470451932834690e-23	-1.668234698324298e-22	-6.225337058771261e-22
-1.197969356701343e-23	-4.470451932834690e-23	-1.668234698324298e-22
-3.210258383620587e-24	-1.197969356701343e-23	-4.470451932834690e-23
-8.602689903507339e-25	-3.210258383620587e-24	-1.197969356701343e-23
-2.305305826892380e-25	-8.602689903507339e-25	-3.210258383620587e-24
-6.177643289615351e-26	-2.305305826892380e-25	-8.602689903507339e-25
-1.655453960534754e-26	-6.177643289615351e-26	-2.305305826892380e-25
-4.436202750742578e-27	-1.655453960534754e-26	-6.177643289615351e-26
-1.188791432130132e-27	-4.436202750742578e-27	-1.655453960534754e-26
-3.185663840251117e-28	-1.188791432130132e-27	-4.436202750742578e-27
-8.536782675912648e-29	-3.185663840251293e-28	-1.188791432130100e-27
-2.287644337577518e-29	-8.536782675912648e-29	-3.185663840251293e-28

Columns 10 through 12

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
2.679750000000000e-01	0	0
-8.939937500006268e-05	2.679750000000000e-01	0
-2.395679751564036e-05	-8.939937500006268e-05	2.679750000000000e-01
-6.419822814254244e-06	-2.395679751564036e-05	-8.939937500006268e-05
-1.720352018649796e-06	-6.419822814254244e-06	-2.395679751564036e-05
-4.610113321976784e-07	-1.720352018649796e-06	-6.419822814254244e-06
-1.235395117456799e-07	-4.610113321976784e-07	-1.720352018649796e-06
-3.310550066004786e-08	-1.235395117456799e-07	-4.610113321976784e-07
-8.871446539376254e-09	-3.310550066004786e-08	-1.235395117456799e-07
-2.377325886389375e-09	-8.871446539376254e-09	-3.310550066004786e-08

-6.370639044052446e-10	-2.377325886389375e-09	-8.871446539376254e-09
-1.707171997829940e-10	-6.370639044052446e-10	-2.377325886389375e-09
-4.574794161184361e-11	-1.707171997829940e-10	-6.370639044052446e-10
-1.225930465343307e-11	-4.574794161184361e-11	-1.707171997829940e-10
-3.285187164503634e-12	-1.225930465343307e-11	-4.574794161184361e-11
-8.803480304078765e-13	-3.285187164503634e-12	-1.225930465343307e-11
-2.359112634485593e-13	-8.803480304078765e-13	-3.285187164503634e-12
-6.321832082262964e-14	-2.359112634485593e-13	-8.803480304078765e-13
-1.694092952244400e-14	-6.321832082262964e-14	-2.359112634485593e-13
-4.539745588776551e-15	-1.694092952244400e-14	-6.321832082262964e-14
-1.216538324152392e-15	-4.539745588776551e-15	-1.694092952244400e-14
-3.260018574147609e-16	-1.216538324152392e-15	-4.539745588776551e-15
-8.736034774071704e-17	-3.260018574147609e-16	-1.216538324152392e-15
-2.341038918581727e-17	-8.736034774071704e-17	-3.260018574147609e-16
-6.273399042070083e-18	-2.341038918581727e-17	-8.736034774071704e-17
-1.681114108298834e-18	-6.273399042070083e-18	-2.341038918581727e-17
-4.504965531713658e-19	-1.681114108298834e-18	-6.273399042070083e-18
-1.207218138360967e-19	-4.504965531713658e-19	-1.681114108298834e-18
-3.235042806272687e-20	-1.207218138360967e-19	-4.504965531713658e-19
-8.669105960109478e-21	-3.235042806272687e-20	-1.207218138360967e-19
-2.323103669660220e-21	-8.669105960109478e-21	-3.235042806272687e-20
-6.225337058771261e-22	-2.323103669660220e-21	-8.669105960109478e-21
-1.668234698324298e-22	-6.225337058771261e-22	-2.323103669660220e-21
-4.470451932834690e-23	-1.668234698324298e-22	-6.225337058771261e-22
-1.197969356701343e-23	-4.470451932834690e-23	-1.668234698324298e-22
-3.210258383620587e-24	-1.197969356701343e-23	-4.470451932834690e-23
-8.602689903507339e-25	-3.210258383620587e-24	-1.197969356701343e-23
-2.305305826892380e-25	-8.602689903507339e-25	-3.210258383620587e-24
-6.177643289615351e-26	-2.305305826892380e-25	-8.602689903507339e-25
-1.655453960534754e-26	-6.177643289615351e-26	-2.305305826892380e-25
-4.436202750742248e-27	-1.655453960534731e-26	-6.177643289615456e-26
-1.188791432130100e-27	-4.436202750742248e-27	-1.655453960534731e-26

Columns 13 through 15

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
2.679750000000000e-01	0	0
-8.939937500006268e-05	2.679750000000000e-01	0
-2.395679751564036e-05	-8.939937500006268e-05	2.679750000000000e-01
-6.419822814254244e-06	-2.395679751564036e-05	-8.939937500006268e-05
-1.720352018649796e-06	-6.419822814254244e-06	-2.395679751564036e-05
-4.610113321976784e-07	-1.720352018649796e-06	-6.419822814254244e-06
-1.235395117456799e-07	-4.610113321976784e-07	-1.720352018649796e-06
-3.310550066004786e-08	-1.235395117456799e-07	-4.610113321976784e-07
-8.871446539376254e-09	-3.310550066004786e-08	-1.235395117456799e-07
-2.377325886389375e-09	-8.871446539376254e-09	-3.310550066004786e-08
-6.370639044052446e-10	-2.377325886389375e-09	-8.871446539376254e-09
-1.707171997829940e-10	-6.370639044052446e-10	-2.377325886389375e-09
-4.574794161184361e-11	-1.707171997829940e-10	-6.370639044052446e-10
-1.225930465343307e-11	-4.574794161184361e-11	-1.707171997829940e-10
-3.285187164503634e-12	-1.225930465343307e-11	-4.574794161184361e-11

-8.803480304078765e-13	-3.285187164503634e-12	-1.225930465343307e-11
-2.359112634485593e-13	-8.803480304078765e-13	-3.285187164503634e-12
-6.321832082262964e-14	-2.359112634485593e-13	-8.803480304078765e-13
-1.694092952244400e-14	-6.321832082262964e-14	-2.359112634485593e-13
-4.53974558776551e-15	-1.694092952244400e-14	-6.321832082262964e-14
-1.216538324152392e-15	-4.53974558776551e-15	-1.694092952244400e-14
-3.260018574147609e-16	-1.216538324152392e-15	-4.53974558776551e-15
-8.736034774071704e-17	-3.260018574147609e-16	-1.216538324152392e-15
-2.341038918581727e-17	-8.736034774071704e-17	-3.260018574147609e-16
-6.273399042070083e-18	-2.341038918581727e-17	-8.736034774071704e-17
-1.681114108298834e-18	-6.273399042070083e-18	-2.341038918581727e-17
-4.504965531713658e-19	-1.681114108298834e-18	-6.273399042070083e-18
-1.207218138360967e-19	-4.504965531713658e-19	-1.681114108298834e-18
-3.235042806272687e-20	-1.207218138360967e-19	-4.504965531713658e-19
-8.669105960109478e-21	-3.235042806272687e-20	-1.207218138360967e-19
-2.323103669660220e-21	-8.669105960109478e-21	-3.235042806272687e-20
-6.225337058771261e-22	-2.323103669660220e-21	-8.669105960109478e-21
-1.668234698324298e-22	-6.225337058771261e-22	-2.323103669660220e-21
-4.470451932834690e-23	-1.668234698324298e-22	-6.225337058771261e-22
-1.197969356701343e-23	-4.470451932834690e-23	-1.668234698324298e-22
-3.210258383620587e-24	-1.197969356701343e-23	-4.470451932834690e-23
-8.602689903507339e-25	-3.210258383620587e-24	-1.197969356701343e-23
-2.305305826892467e-25	-8.602689903507759e-25	-3.210258383620424e-24
-6.177643289615456e-26	-2.305305826892467e-25	-8.602689903507759e-25

Columns 16 through 18

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
2.679750000000000e-01	0	0
-8.939937500006268e-05	2.679750000000000e-01	0
-2.395679751564036e-05	-8.939937500006268e-05	2.679750000000000e-01
-6.419822814254244e-06	-2.395679751564036e-05	-8.939937500006268e-05
-1.720352018649796e-06	-6.419822814254244e-06	-2.395679751564036e-05
-4.610113321976784e-07	-1.720352018649796e-06	-6.419822814254244e-06
-1.235395117456799e-07	-4.610113321976784e-07	-1.720352018649796e-06
-3.310550066004786e-08	-1.235395117456799e-07	-4.610113321976784e-07
-8.871446539376254e-09	-3.310550066004786e-08	-1.235395117456799e-07
-2.377325886389375e-09	-8.871446539376254e-09	-3.310550066004786e-08
-6.370639044052446e-10	-2.377325886389375e-09	-8.871446539376254e-09
-1.707171997829940e-10	-6.370639044052446e-10	-2.377325886389375e-09
-4.574794161184361e-11	-1.707171997829940e-10	-6.370639044052446e-10
-1.225930465343307e-11	-4.574794161184361e-11	-1.707171997829940e-10
-3.285187164503634e-12	-1.225930465343307e-11	-4.574794161184361e-11
-8.803480304078765e-13	-3.285187164503634e-12	-1.225930465343307e-11
-2.359112634485593e-13	-8.803480304078765e-13	-3.285187164503634e-12
-6.321832082262964e-14	-2.359112634485593e-13	-8.803480304078765e-13
-1.694092952244400e-14	-6.321832082262964e-14	-2.359112634485593e-13
-4.53974558776551e-15	-1.694092952244400e-14	-6.321832082262964e-14

-1.216538324152392e-15	-4.539745588776551e-15	-1.694092952244400e-14
-3.260018574147609e-16	-1.216538324152392e-15	-4.539745588776551e-15
-8.736034774071704e-17	-3.260018574147609e-16	-1.216538324152392e-15
-2.341038918581727e-17	-8.736034774071704e-17	-3.260018574147609e-16
-6.273399042070083e-18	-2.341038918581727e-17	-8.736034774071704e-17
-1.681114108298834e-18	-6.273399042070083e-18	-2.341038918581727e-17
-4.504965531713658e-19	-1.681114108298834e-18	-6.273399042070083e-18
-1.207218138360967e-19	-4.504965531713658e-19	-1.681114108298834e-18
-3.235042806272687e-20	-1.207218138360967e-19	-4.504965531713658e-19
-8.669105960109478e-21	-3.235042806272687e-20	-1.207218138360967e-19
-2.323103669660220e-21	-8.669105960109478e-21	-3.235042806272687e-20
-6.225337058771261e-22	-2.323103669660220e-21	-8.669105960109478e-21
-1.668234698324298e-22	-6.225337058771261e-22	-2.323103669660220e-21
-4.470451932834690e-23	-1.668234698324298e-22	-6.225337058771261e-22
-1.197969356701289e-23	-4.470451932834496e-23	-1.668234698324192e-22
-3.210258383620424e-24	-1.197969356701289e-23	-4.470451932834496e-23

Columns 19 through 21

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
2.679750000000000e-01	0	0
-8.939937500006268e-05	2.679750000000000e-01	0
-2.395679751564036e-05	-8.939937500006268e-05	2.679750000000000e-01
-6.419822814254244e-06	-2.395679751564036e-05	-8.939937500006268e-05
-1.720352018649796e-06	-6.419822814254244e-06	-2.395679751564036e-05
-4.610113321976784e-07	-1.720352018649796e-06	-6.419822814254244e-06
-1.235395117456799e-07	-4.610113321976784e-07	-1.720352018649796e-06
-3.310550066004786e-08	-1.235395117456799e-07	-4.610113321976784e-07
-8.871446539376254e-09	-3.310550066004786e-08	-1.235395117456799e-07
-2.377325886389375e-09	-8.871446539376254e-09	-3.310550066004786e-08
-6.370639044052446e-10	-2.377325886389375e-09	-8.871446539376254e-09
-1.707171997829940e-10	-6.370639044052446e-10	-2.377325886389375e-09
-4.574794161184361e-11	-1.707171997829940e-10	-6.370639044052446e-10
-1.225930465343307e-11	-4.574794161184361e-11	-1.707171997829940e-10
-3.285187164503634e-12	-1.225930465343307e-11	-4.574794161184361e-11
-8.803480304078765e-13	-3.285187164503634e-12	-1.225930465343307e-11
-2.359112634485593e-13	-8.803480304078765e-13	-3.285187164503634e-12
-6.321832082262964e-14	-2.359112634485593e-13	-8.803480304078765e-13
-1.694092952244400e-14	-6.321832082262964e-14	-2.359112634485593e-13
-4.539745588776551e-15	-1.694092952244400e-14	-6.321832082262964e-14
-1.216538324152392e-15	-4.539745588776551e-15	-1.694092952244400e-14
-3.260018574147609e-16	-1.216538324152392e-15	-4.539745588776551e-15
-8.736034774071704e-17	-3.260018574147609e-16	-1.216538324152392e-15
-2.341038918581727e-17	-8.736034774071704e-17	-3.260018574147609e-16
-6.273399042070083e-18	-2.341038918581727e-17	-8.736034774071704e-17

-1.681114108298834e-18	-6.273399042070083e-18	-2.341038918581727e-17
-4.504965531713658e-19	-1.681114108298834e-18	-6.273399042070083e-18
-1.207218138360967e-19	-4.504965531713658e-19	-1.681114108298834e-18
-3.235042806272687e-20	-1.207218138360967e-19	-4.504965531713658e-19
-8.669105960109478e-21	-3.235042806272687e-20	-1.207218138360967e-19
-2.323103669660220e-21	-8.669105960109478e-21	-3.235042806272687e-20
-6.225337058771557e-22	-2.323103669660318e-21	-8.669105960109864e-21
-1.668234698324192e-22	-6.225337058771557e-22	-2.323103669660318e-21

Columns 22 through 24

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
2.679750000000000e-01	0	0
-8.939937500006268e-05	2.679750000000000e-01	0
-2.395679751564036e-05	-8.939937500006268e-05	2.679750000000000e-01
-6.419822814254244e-06	-2.395679751564036e-05	-8.939937500006268e-05
-1.720352018649796e-06	-6.419822814254244e-06	-2.395679751564036e-05
-4.610113321976784e-07	-1.720352018649796e-06	-6.419822814254244e-06
-1.235395117456799e-07	-4.610113321976784e-07	-1.720352018649796e-06
-3.310550066004786e-08	-1.235395117456799e-07	-4.610113321976784e-07
-8.871446539376254e-09	-3.310550066004786e-08	-1.235395117456799e-07
-2.377325886389375e-09	-8.871446539376254e-09	-3.310550066004786e-08
-6.370639044052446e-10	-2.377325886389375e-09	-8.871446539376254e-09
-1.707171997829940e-10	-6.370639044052446e-10	-2.377325886389375e-09
-4.574794161184361e-11	-1.707171997829940e-10	-6.370639044052446e-10
-1.225930465343307e-11	-4.574794161184361e-11	-1.707171997829940e-10
-3.285187164503634e-12	-1.225930465343307e-11	-4.574794161184361e-11
-8.803480304078765e-13	-3.285187164503634e-12	-1.225930465343307e-11
-2.359112634485593e-13	-8.803480304078765e-13	-3.285187164503634e-12
-6.321832082262964e-14	-2.359112634485593e-13	-8.803480304078765e-13
-1.694092952244400e-14	-6.321832082262964e-14	-2.359112634485593e-13
-4.53974558776551e-15	-1.694092952244400e-14	-6.321832082262964e-14
-1.216538324152392e-15	-4.53974558776551e-15	-1.694092952244400e-14
-3.260018574147609e-16	-1.216538324152392e-15	-4.53974558776551e-15
-8.736034774071704e-17	-3.260018574147609e-16	-1.216538324152392e-15
-2.341038918581727e-17	-8.736034774071704e-17	-3.260018574147609e-16
-6.273399042070083e-18	-2.341038918581727e-17	-8.736034774071704e-17
-1.681114108298834e-18	-6.273399042070083e-18	-2.341038918581727e-17
-4.504965531713658e-19	-1.681114108298834e-18	-6.273399042070083e-18
-1.207218138360967e-19	-4.504965531713658e-19	-1.681114108298834e-18
-3.235042806272655e-20	-1.207218138360969e-19	-4.504965531713715e-19
-8.669105960109864e-21	-3.235042806272655e-20	-1.207218138360969e-19

Columns 25 through 27

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
2.679750000000000e-01	0	0
-8.939937500006268e-05	2.679750000000000e-01	0
-2.395679751564036e-05	-8.939937500006268e-05	2.679750000000000e-01
-6.419822814254244e-06	-2.395679751564036e-05	-8.939937500006268e-05
-1.720352018649796e-06	-6.419822814254244e-06	-2.395679751564036e-05
-4.610113321976784e-07	-1.720352018649796e-06	-6.419822814254244e-06
-1.235395117456799e-07	-4.610113321976784e-07	-1.720352018649796e-06
-3.310550066004786e-08	-1.235395117456799e-07	-4.610113321976784e-07
-8.871446539376254e-09	-3.310550066004786e-08	-1.235395117456799e-07
-2.377325886389375e-09	-8.871446539376254e-09	-3.310550066004786e-08
-6.370639044052446e-10	-2.377325886389375e-09	-8.871446539376254e-09
-1.707171997829940e-10	-6.370639044052446e-10	-2.377325886389375e-09
-4.574794161184361e-11	-1.707171997829940e-10	-6.370639044052446e-10
-1.225930465343307e-11	-4.574794161184361e-11	-1.707171997829940e-10
-3.285187164503634e-12	-1.225930465343307e-11	-4.574794161184361e-11
-8.803480304078765e-13	-3.285187164503634e-12	-1.225930465343307e-11
-2.359112634485593e-13	-8.803480304078765e-13	-3.285187164503634e-12
-6.321832082262964e-14	-2.359112634485593e-13	-8.803480304078765e-13
-1.694092952244400e-14	-6.321832082262964e-14	-2.359112634485593e-13
-4.539745588776551e-15	-1.694092952244400e-14	-6.321832082262964e-14
-1.216538324152392e-15	-4.539745588776551e-15	-1.694092952244400e-14
-3.260018574147609e-16	-1.216538324152392e-15	-4.539745588776551e-15
-8.736034774071704e-17	-3.260018574147609e-16	-1.216538324152392e-15
-2.341038918581727e-17	-8.736034774071704e-17	-3.260018574147609e-16
-6.273399042070083e-18	-2.341038918581727e-17	-8.736034774071704e-17
-1.681114108298904e-18	-6.273399042069815e-18	-2.341038918581592e-17
-4.504965531713715e-19	-1.681114108298904e-18	-6.273399042069815e-18

Columns 28 through 30

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Columns 31 through 33

Columns 34 through 36

Columns 37 through 39

0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
2.679750000000000e-01	0	0
-8.939937500006268e-05	2.679750000000000e-01	2.679750000000000e-01
-2.395679751564036e-05	-8.939937500006268e-05	2.679750000000000e-01
-6.419822814254244e-06	-2.395679751564036e-05	-8.939937500006268e-05
-1.720352018649796e-06	-6.419822814254244e-06	-2.395679751564036e-05
-4.610113321976784e-07	-1.720352018649796e-06	-6.419822814254244e-06
-1.235395117456799e-07	-4.610113321976784e-07	-1.720352018649796e-06
-3.310550066004786e-08	-1.235395117456799e-07	-4.610113321976784e-07
-8.871446539376254e-09	-3.310550066004786e-08	-1.235395117456799e-07
-2.377325886389375e-09	-8.871446539376254e-09	-3.310550066004786e-08
-6.370639044052446e-10	-2.377325886389375e-09	-8.871446539376254e-09
-1.707171997829940e-10	-6.370639044052446e-10	-2.377325886389375e-09
-4.574794161184361e-11	-1.707171997829940e-10	-6.370639044052446e-10
-1.225930465343389e-11	-4.574794161184672e-11	-1.707171997830052e-10
-3.285187164503621e-12	-1.225930465343389e-11	-4.574794161184672e-11

Columns 40 through 42

0	0	0
2.67975000000000e-01	0	0
-8.93993750006268e-05	2.67975000000000e-01	0
-2.395679751564036e-05	-8.93993750006268e-05	2.67975000000000e-01
-6.419822814254244e-06	-2.395679751564036e-05	-8.93993750006268e-05
-1.720352018649796e-06	-6.419822814254244e-06	-2.395679751564036e-05
-4.610113321976784e-07	-1.720352018649796e-06	-6.419822814254244e-06
-1.235395117456799e-07	-4.610113321976784e-07	-1.720352018649796e-06
-3.310550066004786e-08	-1.235395117456799e-07	-4.610113321976784e-07
-8.871446539376254e-09	-3.310550066004786e-08	-1.235395117456799e-07
-2.377325886389375e-09	-8.871446539376254e-09	-3.310550066004786e-08
-6.370639044052660e-10	-2.377325886389469e-09	-8.871446539376185e-09
-1.707171997830052e-10	-6.370639044052660e-10	-2.377325886389469e-09

Columns 43 through 45

-1.720352018649796e-06	-6.419822814254244e-06	-2.395679751564036e-05
-4.610113321976784e-07	-1.720352018649796e-06	-6.419822814254244e-06
-1.235395117456799e-07	-4.610113321976784e-07	-1.720352018649796e-06
-3.310550066004851e-08	-1.235395117456749e-07	-4.610113321976530e-07
-8.871446539376185e-09	-3.310550066004851e-08	-1.235395117456749e-07

Columns 46 through 48

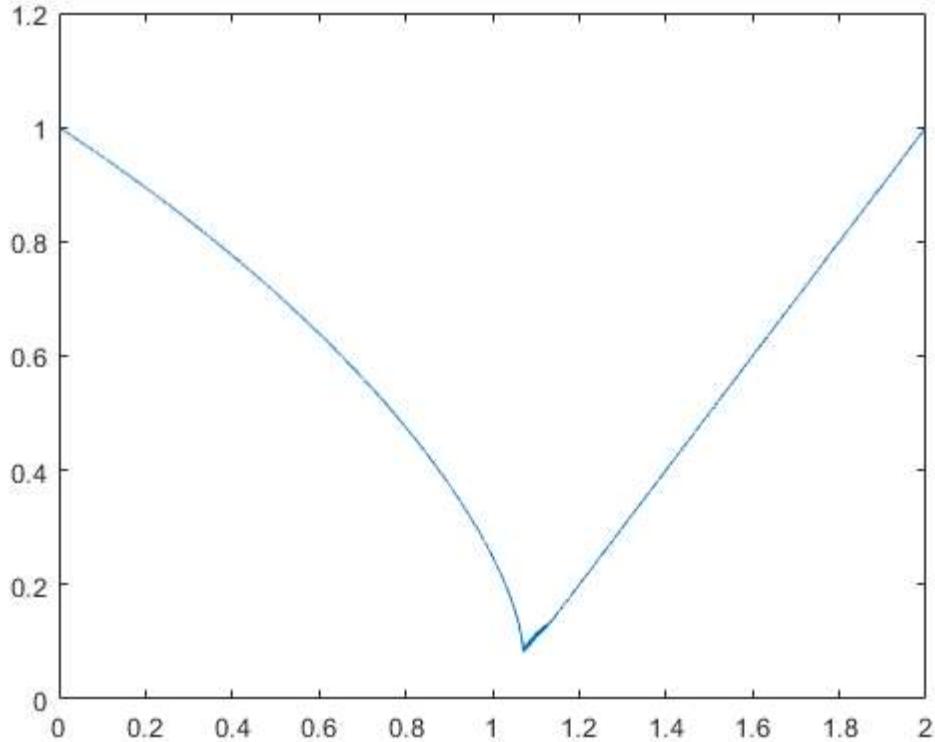
Columns 49 through 50

ans =

20×1 table

Results for all Iterations		
Iteration	Residual	Error
1.0000000000000e+00	2.0000000000000e+00	1.38838897897155e+01

2.00000000000000e+00	4.09802068918411e+04	1.90651809875899e+05
3.00000000000000e+00	3.82983414877158e+04	2.41061409608893e+05
4.00000000000000e+00	3.75787748706414e+04	2.54321234834212e+05
5.00000000000000e+00	3.73857096168345e+04	2.57801819283184e+05
6.00000000000000e+00	3.73339088342542e+04	2.58714307773756e+05
7.00000000000000e+00	3.73200103444969e+04	2.58953335791693e+05
8.00000000000000e+00	3.73162812963189e+04	2.59015909782619e+05
9.00000000000000e+00	3.73152807724070e+04	2.59032281193095e+05
1.00000000000000e+01	3.73150123269563e+04	2.59036562007445e+05
1.10000000000000e+01	3.73149403018804e+04	2.59037680689532e+05
1.20000000000000e+01	3.73149209772808e+04	2.59037972845809e+05
1.30000000000000e+01	3.73149157924285e+04	2.59038049095545e+05
1.40000000000000e+01	3.73149144013187e+04	2.59038068982164e+05
1.50000000000000e+01	3.73149140280811e+04	2.59038074164991e+05
1.60000000000000e+01	3.73149139279408e+04	2.59038075514692e+05
1.70000000000000e+01	3.73149139010731e+04	2.59038075865892e+05
1.80000000000000e+01	3.73149138938645e+04	2.59038075957197e+05
1.90000000000000e+01	3.73149138919304e+04	2.59038075980913e+05
2.00000000000000e+01	3.73149138914115e+04	2.59038075987067e+05



```

function [Results,T,rho] = myjacobi(A,b,x0,tol,Niter)
[A,b,x0] = Test(50);
tol = 10^-5;
Niter = 10;
[n,~] = size(A);
X = zeros(n,Niter);
X(:,1) = x0;
D = zeros(n);
U = -triu(A);
L = -tril(A);
for i = 1:n
    D(i,i) = A(i,i);
    U(i,i) = 0;
    L(i,i) = 0;
end
Dinv = D\eye(n);
T = Dinv*(L+U);
rho = max(abs(eig(T)));
for k = 2:Niter
    X(:,k) = Dinv*b + T*X(:,k-1);
    if norm(X(:,k)-X(:,k-1),inf)/norm(X(:,k),inf) < tol
        t = X(:,1:k);
        n = size(t,2);
        Iteration = zeros(n,1);
        Residual = zeros(n,1);
        Error = zeros(n,1);
        for i = 1:n
            Iteration(i) = i;
            Residual(i) = norm(A*X(:,i)-b,inf);
            Error(i) = norm(X(:,i)-A\b);
        end
        R = table(Iteration,Residual>Error);
        Results = table(R,'VariableNames',{'Results for all Iterations'});
        return
    end
end
Results = 'N/A';
disp('Jacobi failed to converge after maximum iterations')

```

Jacobi failed to converge after maximum iterations

ans =

'N/A'

```

function [Results,T,rho] = mygauseidel(A,b,x0,tol,Niter)
[A,b,x0] = Test(50);
tol = 10^-5;
Niter = 10;
[n,~] = size(A);
X = zeros(n,Niter);
X(:,1) = x0;
D = zeros(n);
U = -triu(A);
L = -tril(A);
for i = 1:n
    D(i,i) = A(i,i);
    U(i,i) = 0;
    L(i,i) = 0;
end
Dinv = (D-L)\eye(n);
T = Dinv*U;
rho = max(abs(eig(T)));
for k = 2:Niter
    X(:,k) = Dinv*b + T*X(:,k-1);
    if norm(X(:,k)-X(:,k-1),inf)/norm(X(:,k),inf) < tol
        t = X(:,1:k);
        n = size(t,2);
        Iteration = zeros(n,1);
        Residual = zeros(n,1);
        Error = zeros(n,1);
        for i = 1:n
            Iteration(i) = i;
            Residual(i) = norm(A*X(:,i)-b,inf);
            Error(i) = norm(X(:,i)-A\b);
        end
        R = table(Iteration,Residual>Error);
        Results = table(R,'VariableNames',{'Results for all Iterations'});
        return
    end
end
Results = 'N/A';
disp('Gauss-Seidel failed to converge after maximum iterations')

```

Gauss-Seidel failed to converge after maximum iterations

ans =

'N/A'

```

function [Results,T,w_opt,rho] = completeSOR(A,b,x0,tol,Niter)
[A,b,x0] = Test(50);
tol = 10^-5;
Niter = 10;
[n,~] = size(A);
X = zeros(n,Niter);
X(:,1) = x0;
D = zeros(n);
U = -triu(A);
L = -tril(A);
for i = 1:n
    D(i,i) = A(i,i);
    U(i,i) = 0;
    L(i,i) = 0;
end
W = zeros(20001,1);
for w = 0:20000
    Dinv = (D-w/10000*L)\eye(n);
    T = Dinv*((1-w/10000)*D+w/10000*U);
    W(w+1) = max(abs(eig(T)));
end
[rho,I] = min(W);
w_opt = (I-1)/10000;
x = 0:0.0001:2;
Dinv = (D-w_opt*L)\eye(n);
T = Dinv*((1-w_opt)*D+w_opt*U);
for k = 2:Niter
    X(:,k) = Dinv*w*b + T*X(:,k-1);
    if norm(X(:,k)-X(:,k-1),inf)/norm(X(:,k),inf) < tol
        t = X(:,1:k);
        n = size(t,2);
        Iteration = zeros(n,1);
        Residual = zeros(n,1);
        Error = zeros(n,1);
        for i = 1:n
            Iteration(i) = i;
            Residual(i) = norm(A*X(:,i)-b,inf);
            Error(i) = norm(X(:,i)-A\b);
        end
        R = table(Iteration,Residual>Error);
        Results = table(R,'VariableNames',{'Results for all Iterations'});
        return
    end
end
Results = 'N/A';
disp('SOR failed to converge after maximum iterations')

```

SOR failed to converge after maximum iterations

ans =

'N/A'