

Here is my code:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
type iter.m

function x = iter(A,x0,b,S,T,q)
%function x = iter(A,x0,b,S,T,q)
%
%This is an algorithm designed by Alexander Winkles that iteratively solves
%systems of equations of the form  $Ax = b$  using various standard methods.
%
%A : the matrix
%x0 : the guess solution to start with
%b : the vector
%S : the number of iterations
%T : the tolerance of the result using an infinity norm
%q : indicates which method to use
%    q == 1 - Jacobi
%    q == 2 - Gauss-Seidel
%    q == 3 - steepest descent

n = size(A,1);
m = size(A,2);

if n ~= m
    fprintf('\nThis is not an n x n matrix!\n')
elseif det(A) < 1e-14
    fprintf('\nThis matrix is singular!\n\n')
else
    if q == 1

        % Jacobi %

        x = x0;
        u = zeros(n,1);
        k = 1;
        while k <= S
            for i=1:n
                sum = 0;
                for j=1:n
                    if j ~= i
                        sum = sum + A(i,j)*x(j);
                    end;
                end;
                u(i) = (b(i) - sum)/A(i,i);
            end;
            for i=1:n
                x(i) = u(i);
            end;
            err = norm(A*x-b,Inf);
            if err < T
                fprintf('\n The iteration was successful after %d iterations with ||Ax - b|| = %d!\n', k,err)
                break;
            end;
            k = k+1;
        end;
        if k == S + 1
            err = norm(A*x-b, Inf);
            fprintf('\nMethod failed after %d iterations with ||Ax-b|| = %d.\n\n',S,err)
        end;
    end;

    % Gauss-Seidel %

    if q == 2
        x = x0;
        k = 1;
        while k <= S
            for i=1:n
```

```

        sum = 0;
        for j=1:n
            if j ~= i
                sum = sum + A(i,j)*x(j);
            end;
        end;
        x(i) = (b(i)-sum)/A(i,i);
    end;
    err = norm(A*x-b, Inf);
    if err < T
        fprintf('\nThe iteration was successful after %d iterations with ||Ax - b|| = %d!\n', k,err)
        break;
    end;
    k = k+1;
end;
if k == S+1
    err = norm(A*x-b, Inf);
    fprintf('\nMethod failed after %d iterations with ||Ax-b|| = %d.\n\n',S,err)
end;
end;

% Steepest descent %

if q == 3
    k = 1;
    x = x0;
    while k <= S
        v = b - A*x;
        t = dot(v,v)/dot(v,A*v);
        x = x + t*v;
        err = norm(A*x-b, Inf);
        if err < T
            fprintf('\nThe iteration was successful after %d iterations with ||Ax - b|| = %d!\n',k,err);
            break;
        end;
        k = k+1;
    end;
    if k == S+1
        err = norm(A*x-b, Inf);
        fprintf('\nMethod failed after %d iterations with ||Ax-b|| = %d.\n\n',S,err)
    end;
end;

end;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A code to generate matrices to be solved:

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
type nonsingmat.m

function A = nonsingmat(n,q)
%function A = nonsingmat
%
%A quick script that generates nonsingular matrices to solve.
%
%n : the size of the n*n matrix to be generated
%q : the type of matrix desired
%    q == 1 - nonsingular
%    q == 2 - positive definite
%    q == 3 - diagonal dominant
%    q == 4 - diagonally dominant & positive definite

A = randi(10,n,n);
if q == 1
    while det(A) < 1e-14
        A = randi(10,n,n);
    end;

```

```

end;
if q == 2
    B = randi(10,n,n);
    A = B'*B + eye(n);
    while det(A) < 1e-14;
        B = randi(10,n,n);
        A = B'*B + eye(n);
    end;
end;
if q == 3
    A = randi(10,n,n);
    A = A + diag(sum(abs(A),2));
end;
if q == 4
    B = randi(10,n,n);
    A = B'*B + eye(n);
    A = A + diag(sum(abs(A),2));
end;

%%% Problem 1 %%%%%%%%%%%%%%

A = nonsingmat(10,2)

A =

    401    313    254    238    294    281    321    284    206    353
    313    367    316    320    337    293    272    211    204    341
    254    316    344    304    312    230    197    179    162    300
    238    320    304    344    285    208    216    163    152    253
    294    337    312    285    349    278    231    203    178    326
    281    293    230    208    278    308    240    206    206    330
    321    272    197    216    231    240    312    228    163    288
    284    211    179    163    203    206    228    224    159    277
    206    204    162    152    178    206    163    159    175    256
    353    341    300    253    326    330    288    277    256    450

b = randi(10,10,1)

b =

     2
     8
     7
     2
     3
    10
     3
     1
     6
     1

x = zeros(10,1)

x =

     0
     0
     0
     0
     0
     0
     0
     0
     0
     0

y = iter(A,x,b,100000,1e-10,2)

```

The iteration was successful after 3418 iterations with $||Ax - b|| = 9.941736e-11!$

y =

```
-0.5986
 1.3649
 0.2703
-0.7032
-0.4254
-0.0844
-0.1228
 0.9896
-0.0084
-0.5030
```

A*y - b

ans =

```
1.0e-10 *
 0.6060
-0.9942
-0.9223
-0.4339
-0.0885
 0.0150
 0.3327
-0.3230
-0.1825
-0.0002
```

%% Problem 2 %%

A = nonsingmat(10,3)

A =

```
67      8      7      6      7      8      8      5      2      2
 1    43      6      5      2      3      4      2      8    10
 1      1    51      9      8      8      3      8      3      2
 8      7      1    57      2    10      1      2      7      3
10      7      8      8    75      9      8      3    10      8
 6      6      4      1      7    42      3      3      5      5
 2      8      7      1      4      4    55      6      7      8
 9      8      8      1      7      4      6    57      8      4
 4      8      2      8      8      7      3      5    58      3
 3      3      2    10      6      6      7      2      7    48
```

b = randi(10,10,1)

b =

```
7
5
5
7
1
4
8
7
2
2
```

x = zeros(10,1)

x =

```

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

y = iter(A,x,b,1000,1e-10,1)

The iteration was successful after 198 iterations with ||Ax - b|| = 9.701562e-11!

y =

    0.0578
    0.0837
    0.0568
    0.0907
   -0.0385
    0.0622
    0.1143
    0.0828
   -0.0103
   -0.0100

A*y - b

ans =

    1.0e-10 *

   -0.7467
   -0.5627
   -0.6007
   -0.5790
   -0.9702
   -0.5587
   -0.6687
   -0.7608
   -0.6725
   -0.6219

%% Problem 3 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

A = nonsingmat(10,2)

A =

    391    257    283    379    349    174    179    275    302    329
    257    263    227    306    268    121    139    250    296    249
    283    227    366    342    279    221    227    332    290    327
    379    306    342    475    396    216    263    383    381    404
    349    268    279    396    438    237    257    344    352    413
    174    121    221    216    237    217    186    264    188    246
    179    139    227    263    257    186    235    272    217    299
    275    250    332    383    344    264    272    431    355    350
    302    296    290    381    352    188    217    355    405    337
    329    249    327    404    413    246    299    350    337    459

b = randi(10,10,1)

b =

     2
     5

```

```

9
6
4
3
5
10
2
5

x = zeros(10,1)

x =

    0
    0
    0
    0
    0
    0
    0
    0
    0
    0

y = iter(A,x,b,100000000,1e-10,3)

The iteration was successful after 7135 iterations with ||Ax - b|| = 9.980017e-11!

y =

-0.1126
-0.0937
 0.2063
-0.0648
 0.2830
-0.3826
-0.2188
 0.3235
-0.2052
 0.0494

%%% Problem 4 %%%%%%%%%%%%%%

A = nonsingmat(100,4);

b = randi(10,100,1);

x = zeros(100,1);

y = iter(A,x,b,100000000,1e-5,1);

The iteration was successful after 526 iterations with ||Ax - b|| = 9.982480e-06!

y = iter(A,x,b,100000000,1e-5,2);

The iteration was successful after 9 iterations with ||Ax - b|| = 6.813123e-06!

y = iter(A,x,b,100000000,1e-5,3);

The iteration was successful after 12 iterations with ||Ax - b|| = 5.626486e-06!

%%% Problem 5 %%%%%%%%%%%%%%

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

A =

```

```

1      1      1
1      2      3
1      3      6

a1 = A(:,1);
a2 = A(:,2);
a3 = A(:,3);
proj1 = (a1'*A*a1)^(1/2)

proj1 =

    4.3589

v1 = a1/proj1

v1 =

    0.2294
    0.2294
    0.2294

u1 = v1;
alpha1 = -v1'*A*a2

alpha1 =

   -10.3237

v2 = a2 + alpha1*u1

v2 =

   -1.3684
   -0.3684
    0.6316

proj2 = (v2'*A*v2)^(1/2)

proj2 =

    1.5560

u2 = v2/proj2

u2 =

   -0.8795
   -0.2368
    0.4059

alpha2 = -u1'*A*a3;
alpha3 = -u2'*A*a3;
v3 = a3 + alpha2*u1+alpha3*u2

v3 =

    0.2174
   -0.3261
    0.1304

proj3 = (v3'*A*v3)^(1/2)

proj3 =

    0.1474

u3 = v3/proj3

```

```
u3 =
```

```
    1.4744  
   -2.2116  
    0.8847
```

```
u1'*A*u1
```

```
ans =
```

```
    1.0000
```

```
u1'*A*u2
```

```
ans =
```

```
    1.8874e-15
```

```
u1'*A*u3
```

```
ans =
```

```
   -3.1530e-14
```

```
diary off
```