Link

`https://github.com/candymate/num_analysis/tree/master/7`

Problem 2

Code

```
clear;

A = [2 -1 -1; 2 2 2; -1 -1 2];
b = [-1; 4; -5];

T_j = [0 1/2 -1/2; -1 0 -1; 1/2 1/2 0];
disp("eig T_j");
disp(eig(T_j));
% sqrt(5)/2 % just to check if 1.1180 is sqrt(5)/2

x = zeros(3, 1);
C = diag([1/2; 1/2; 1/2]) * b;
for i = 1:25
    x = T_j*x + C;
end
disp("After iteration");
disp(x);

temp1 = [2 0 0; 2 2 0; -1 -1 2];
temp2 = [0 1 -1; 0 0 -2; 0 0 0];
T_g = inv(temp1)*temp2; %#ok<MINV>
disp("eig T_g");
disp(eig(T_g));

x = zeros(3, 1);
C = inv(temp1) * b; %#ok<MINV>
i = 0;
while max(abs(x - [1; 2; -1])) >= 10^(-5)
    i = i + 1;
    x = T_g*x + C;
end
disp("i");
disp(i);
disp("After iteration");
disp(x);
disp("error");
disp(max(abs(x - [1; 2; -1])));
```

Result

```
eig T_j
   0.0000 + 1.1180i
   0.0000 - 1.1180i
   0.0000 + 0.0000i

After iteration
  -20.8279
    2.0000
  -22.8279

eig T_g
         0
   -0.5000
   -0.5000

i
    21

After iteration
    1.0000
    2.0000
   -1.0000

error
   9.0599e-06
```

Problem 3

Code

```
clear;

A = [1 2 -2; 1 1 1; 2 2 1];
b = [7; 2; 5];

T_j = [0 -2 2; -1 0 -1; -2 -2 0];
disp("eig T_j");
disp(eig(T_j));

x = zeros(3, 1);
C = diag([1; 1; 1]) * b;
i = 0;
while max(abs(x - [1; 2; -1])) >= 10^(-5)
    i = i + 1;
    x = T_j*x + C;
end
disp("i");
```

```
disp(i);
disp("After iteration");
disp(x);
disp("error");
disp(max(abs(x - [1; 2; -1])));

temp1 = [1 0 0; 1 1 0; 2 2 1];
temp2 = [0 -2 2; 0 0 -1; 0 0 0];
T_g = inv(temp1)*temp2; %#ok<MINV>
disp("eig T_g");
disp(eig(T_g));

x = zeros(3, 1);
C = inv(temp1) * b; %#ok<MINV>
for i = 1:25
    x = T_g*x + C;
end
disp("After iteration");
disp(x);
```

Result

```
eig T_j
   1.0e-04 *

  -0.0617 + 0.1068i
  -0.0617 - 0.1068i
   0.1233 + 0.0000i

i
     3

After iteration
     1
     2
    -1

error
     0

eig T_g
     0
     2
     2

After iteration
   1.0e+09 *

    1.3086
   -1.3254
    0.0336
```

Problem 4

Code

```
clear;

A = [0.2 0.1 1 1 0; 0.1 4 -1 1 -1; 1 -1 60 0 -2; 1 1 0 8 4; 0 -1 -2 4 700];
b = [1; 2; 3; 4; 5];
exact_sol = [7.859713071; ...
             0.4229264082; ...
             -0.07359223906; ...
             -0.5406430164; ...
             0.01062616286];
D = diag(diag(A));
U = -triu(A, 1);
L = -tril(A, -1);

disp("a - jacobi method");
x = zeros(5, 1);
T = inv(D) * (L + U); %#ok<MINV>
C = inv(D) * b; %#ok<MINV>
i = 0;
while max(abs(x - exact_sol)) >= 10^(-2)
    i = i + 1;
    x = T*x + C;
end
disp("i");
disp(i);
disp("After iteration");
disp(x);
disp("error");
disp(max(abs(x - exact_sol)));

disp("b - gauss siedel method");
x = zeros(5, 1);
T = inv(D - L) * U; %#ok<MINV>
C = inv(D - L) * b; %#ok<MINV>
i = 0;
while max(abs(x - exact_sol)) >= 10^(-2)
    i = i + 1;
    x = T*x + C;
end
disp("i");
disp(i);
disp("After iteration");
disp(x);
disp("error");
disp(max(abs(x - exact_sol)));

disp("c - sor method");
x = zeros(5, 1);
```

```
w = 1.25;
T = inv(D - w*L) * (w*U + (1-w)*D); %#ok<MINV>
C = inv(D - w*L) * w*b; %#ok<MINV>
i = 0;
while max(abs(x - exact_sol)) >= 10^(-2)
    i = i + 1;
    x = T*x + C;
end
disp("i");
disp(i);
disp("After iteration");
disp(x);
disp("error");
disp(max(abs(x - exact_sol)));

disp("d - cg method");
x = zeros(5, 1);
r = b - A*x;
v = r;
while max(abs(x - exact_sol)) >= 10^(-2)
    i = i + 1;
    t = (r'*r) / (v'*A*v);
    x = x + t*v;
    r_temp = r;
    r = r - t*A*v;
    s = -(r'*r) / (r_temp'*r_temp);
    v = r + s*v;
end
disp("i");
disp(i);
disp("After iteration");
disp(x);
disp("error");
disp(max(abs(x - exact_sol)));
```

Result

```
a - jacobi method
i
    37

After iteration
    7.8695
    0.4242
   -0.0730
   -0.5359
    0.0106

error
    0.0098

b - gauss siedel method
```

```
i
    18

After iteration
    7.8509
    0.4228
   -0.0734
   -0.5395
    0.0106

error
    0.0088

c - sor method
i
     7

After iteration
    7.8515
    0.4228
   -0.0735
   -0.5398
    0.0106

error
    0.0082

d - cg method
i
       57874

After iteration
    7.8696
    0.4306
   -0.0763
   -0.5429
    0.0102

error
    0.0099
```