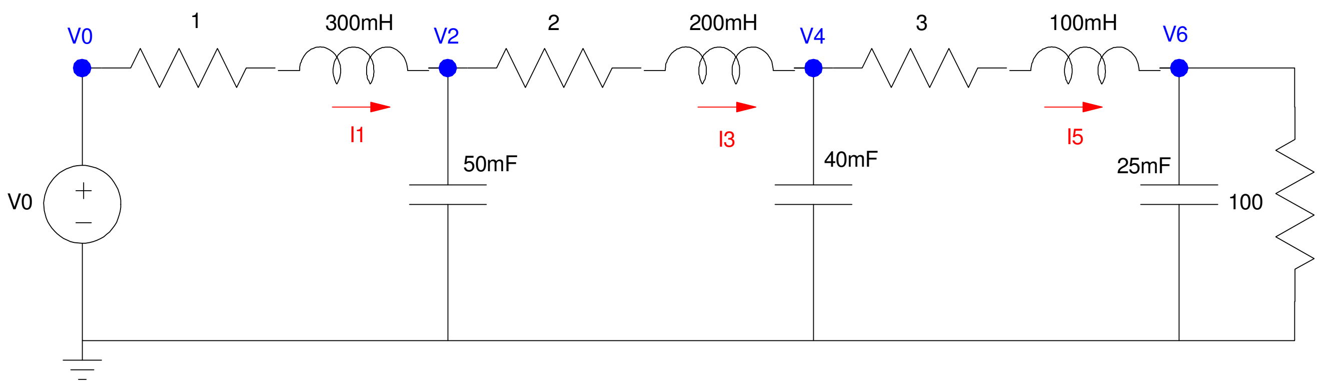
ECE 463/663 - Homework #2

State-Space, Eigenvalues, Eigenvectors. Due Monday, Jan 23rd

1) For the following RLC circuit

Specify the dynamics for the system (write N coupled differential equations)



Inductors: V = L dI/dt = LsI

$$V_1 = 0.3sI_1 = V_0 - I_1 - V_2$$

$$V_3 = 0.2sI_3 = V_2 - 2I_3 - V_4$$

$$V_5 = 0.1sI_5 = V_4 - 3I_5 - V_6$$

Capacitors: I = C dV/dt = CsV

$$I_2 = 0.05sV_2 = I_1 - I_3$$

$$I_4 = 0.04sV_4 = I_3 - I_5$$

$$I_6 = 0.025sV_6 = I_5 - \left(\frac{V_6}{100}\right)$$

Plugging in numbers:

$$sI_1 = 3.33V_0 - 3.33I_1 - 3.33V_2$$

$$sV_2 = 20I_1 - 20I_3$$

$$sI_3 = 5V_2 - 10I_3 - 5V_4$$

$$sV_4 = 25I_3 - 25I_5$$

$$sI_5 = 10V_4 - 30I_5 - 10V_6$$

$$sV_6 = 40I_5 - 0.4V_6$$

Express these dynamics in state-space form

$$\begin{bmatrix} sI_1 \\ sV_2 \\ sI_3 \\ sV_4 \\ sI_5 \\ sV_6 \end{bmatrix} = \begin{bmatrix} -3.33 & -3.33 & 0 & 0 & 0 & 0 \\ 20 & 0 & -20 & 0 & 0 & 0 \\ 0 & 5 & -10 & -5 & 0 & 0 \\ 0 & 0 & 25 & 0 & -25 & 0 \\ 0 & 0 & 0 & 10 & -35 & -10 \\ 0 & 0 & 0 & 0 & 50 & -0.4 \end{bmatrix} \begin{bmatrix} I_1 \\ V_2 \\ I_3 \\ V_4 \\ I_5 \\ V_6 \end{bmatrix} + \begin{bmatrix} 3.33 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} V_0$$

$$Y = V_6 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} X$$

Determine the transfer function from Vin to V5

```
>> a1 = [-3.3333,-3.3333,0,0,0,0];
>> a2 = [20,0,-20,0,0,0];
>> a3 = [0,5,-10,-5,0,0];
>> a4 = [0,0,25,0,-25,0];
>> a5 = [0,0,0,10,-35,-10];
>> a6 = [0,0,0,0,50,-0.4];
>> A = [a1;a2;a3;a4;a5;a6]

   -3.3333   -3.3333         0         0         0         0
   20.0000         0  -20.0000         0         0         0
         0    5.0000  -10.0000   -5.0000         0         0
         0         0   25.0000         0  -25.0000         0
         0         0         0   10.0000  -35.0000  -10.0000
         0         0         0         0   50.0000   -0.4000

>> B = [3.3333;0;0;0;0;0]

    3.3333
         0
         0
         0
         0
         0

>> C = [0,0,0,0,0,1];
>> D = 0;
>> G6 = ss(A,B,C,D);
```

```
>> zpk(G6)

                          4166625
-----------------------------------------------------------------
(s^2 + 5.573s + 27.64) (s^2 + 10.6s + 217) (s^2 + 32.56s + 730.7)

>> P = eig(A)

  -2.7863 + 4.4580i
  -2.7863 - 4.4580i
  -5.2981 +13.7465i
  -5.2981 -13.7465i
 -16.2822 +21.5785i
 -16.2822 -21.5785i
```

2) For the transfer function from V0 to V6

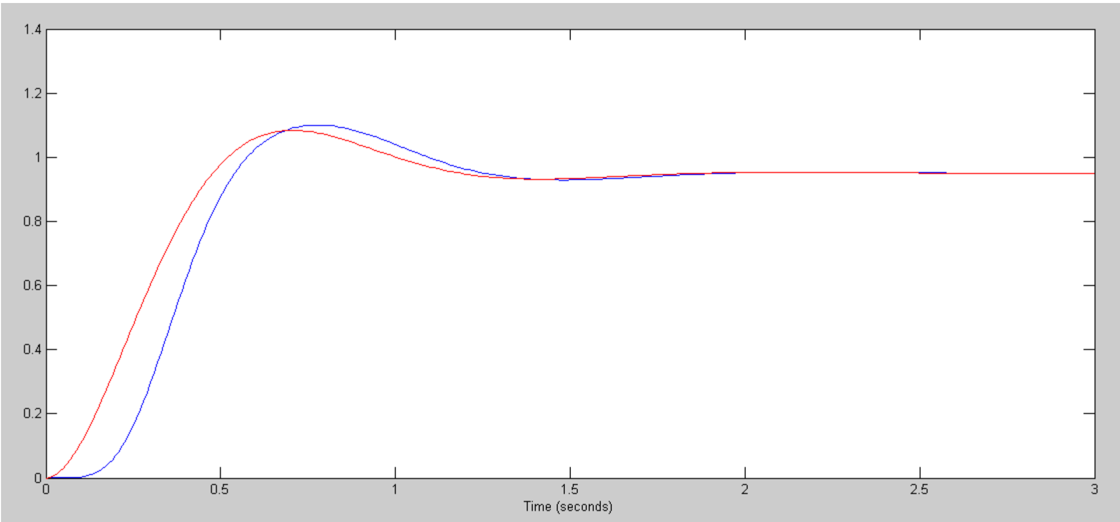
- Determine a 1st or 2nd-order approximation for this trasfer function
- Plot the step response of the actual 6th-order system and its approximation

Keep the dominant pole(s)

```
  -2.7863 + 4.4580i
  -2.7863 - 4.4580i
```

Match the DC gain

```
>> G2 = zpk([],P(1:2),1);
>> k = evalfr(G6,0) / evalfr(G2,0)

k =   26.2716

>> G2 = zpk([],P(1:2),k)


       26.2716
----------------------
(s^2 + 5.573s + 27.64)

>> t = [0:0.01:3]';
>> y6 = step(G6,t);
>> y2 = step(G2,t);
>> plot(t,y6,'b',t,y2,'r');
```



6th Order System (blue) & 2nd-Order Approximation (red)

3) For this circuit...

- What initial condition will the energy in the system decay as slowly as possible?
- What initial condition will the energy in the system decay as fast as possible?

This is an eigvalue / eigenvector problem. The eigenvectors or A are:

```
>> [M,V] = eig(A)

      slow                                                                                            fast
 -0.2154 + 0.1523i  -0.2154 - 0.1523i   0.0211 + 0.1479i   0.0211 - 0.1479i   0.0097 - 0.0039i   0.0097 + 0.0039i
  0.2390 + 0.2630i   0.2390 - 0.2630i   0.6224             0.6224             0.0125 - 0.0775i   0.0125 + 0.0775i
 -0.1234 + 0.1356i  -0.1234 - 0.1356i   0.1860 - 0.2799i   0.1860 + 0.2799i  -0.0638 - 0.0805i  -0.0638 + 0.0805i
  0.5380 + 0.1774i   0.5380 - 0.1774i  -0.3220 - 0.2482i  -0.3220 + 0.2482i  -0.4150 + 0.0966i  -0.4150 - 0.0966i
 -0.0318 + 0.0595i  -0.0318 - 0.0595i  -0.0187 - 0.1554i  -0.0187 + 0.1554i  -0.2507 + 0.3406i  -0.2507 - 0.3406i
  0.6671             0.6671            -0.4801 + 0.2391i  -0.4801 - 0.2391i   0.7892             0.7892


 -2.7863 - 4.4580i  -2.7863 + 4.4580i  -5.2981 -13.7465i  -5.2981 +13.7465i -16.2822 -21.5785i -16.2822 +21.5785i

>>
```

The slow mode (blue) has initial conditions of

$$x(t) = x_0 e^{-2.7863t} \cos(4.4580t)$$

```
>> real(M(:,1))

I1   -0.2154
V2    0.2390
I3   -0.1234
V4    0.5380
I5   -0.0318
V6    0.6671
```

$$x(t) = x_0 e^{-2.7863t} \sin(4.4580t)$$

```
>> imag(M(:,1))

I1    0.1523
V2    0.2630
I3    0.1356
V4    0.1774
I5    0.0595
V6         0
```

The fast mode (red) has initial conidtions of

$$x(t) = x_0 e^{-16.28t} \cos(21.58t)$$

```
>> real(M(:,6))

I1    0.0097
V2    0.0125
I3   -0.0638
V4   -0.4150
I5   -0.2507
V6    0.7892
```

$$x(t) = x_0 e^{-16.28t} \sin(21.58t)$$

```
>> imag(M(:,6))

I1    0.0039
V2    0.0775
I3    0.0805
V4   -0.0966
I5   -0.3406
V6         0
```

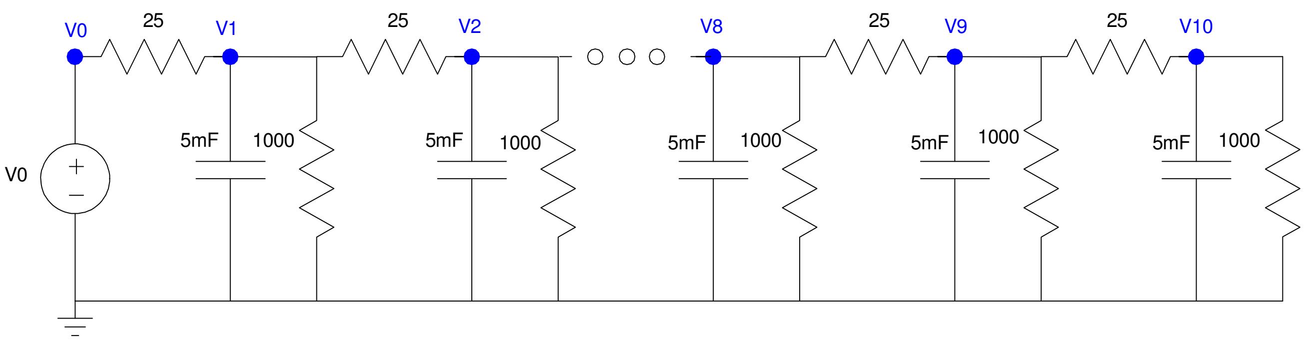
Problem 4-7: 10-Stage RC Filter.

note: You can turn in the Matlab code along with screen shots of the plots if you like.

4) For the following 10-stage RC circuit

Specify the dynamics for the system (write N coupled differential equations)

- note: Nodes 1..9 have the same form. Just write the node equation for node 1 and node 10.



$$0.005sV_1 = \left(\frac{V_0 - V_1}{25}\right) + \left(\frac{V_2 - V_1}{25}\right) - \left(\frac{V_1}{1000}\right)$$

$$sV_1 = 8V_0 - 16.2V_1 + 8V_2$$

the same pattern holds for nodes 2..9

$$sV_2 = 8V_1 - 16.2V_2 + 8V_3$$

$$sV_3 = 8V_2 - 16.2V_3 + 8V_4$$

$$\vdots$$

$$sV_9 = 8V_8 - 16.2V_9 + 8V_{10}$$

Node #10 is the oddball since it only has one 25 ohm resistor conencted:

$$0.005sV_{10} = \left(\frac{V_9 - V_{10}}{25}\right) - \left(\frac{V_{10}}{1000}\right)$$

$$sV_{10} = 8V_9 - 8.2V_{10}$$

Express these dynamics in state-space form

$$
\begin{bmatrix} sV_1 \\ sV_2 \\ sV_3 \\ sV_4 \\ sV_5 \\ sV_6 \\ sV_7 \\ sV_8 \\ sV_9 \\ sV_{10} \end{bmatrix}
=
\begin{bmatrix}
-16.2 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
8 & -16.2 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 8 & -16.2 & 8 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 8 & -16.2 & 8 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 8 & -16.28 & 8 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 8 & -16.2 & 8 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 8 & -16.2 & 8 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 8 & -16.2 & 8 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 8 & -16.2 & 8 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 8 & -8.2
\end{bmatrix}
\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \\ V_7 \\ V_8 \\ V_9 \\ V_{10} \end{bmatrix}
+
\begin{bmatrix} 8 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} V_0
$$

Determine the transfer function from V0 to V10

```
>> A = zeros(10,10);
>> for i=1:9
A(i,i) = -16.2;
A(i+1,i) = 8;
A(i,i+1) = 8;
end
>> A(10,10) = -8.2;

  -16.2000    8.0000         0         0         0         0         0         0         0         0
    8.0000  -16.2000    8.0000         0         0         0         0         0         0         0
         0    8.0000  -16.2000    8.0000         0         0         0         0         0         0
         0         0    8.0000  -16.2000    8.0000         0         0         0         0         0
         0         0         0    8.0000  -16.2000    8.0000         0         0         0         0
         0         0         0         0    8.0000  -16.2000    8.0000         0         0         0
         0         0         0         0         0    8.0000  -16.2000    8.0000         0         0
         0         0         0         0         0         0    8.0000  -16.2000    8.0000         0
         0         0         0         0         0         0         0    8.0000  -16.2000    8.0000
         0         0         0         0         0         0         0         0    8.0000   -8.2000

>> B = [8;0;0;0;0;0;0;0;0;0]
>> C = [0,0,0,0,0,0,0,0,0,1];
>> D = 0;
>> G10 = ss(A,B,C,D);
>> zpk(G10)

                                        1073741824
----------------------------------------------------------------------------------------------
(s+31.49) (s+29.42) (s+26.18) (s+22.05) (s+17.4) (s+12.64) (s+8.2) (s+4.471) (s+1.784) (s+0.3787)
```

5) For the transfer function for problem #4

- Determine a 2nd-order approximation for this trasfer function
- Plot the step response of the actual 10th-order system and its 2nd-order approximation

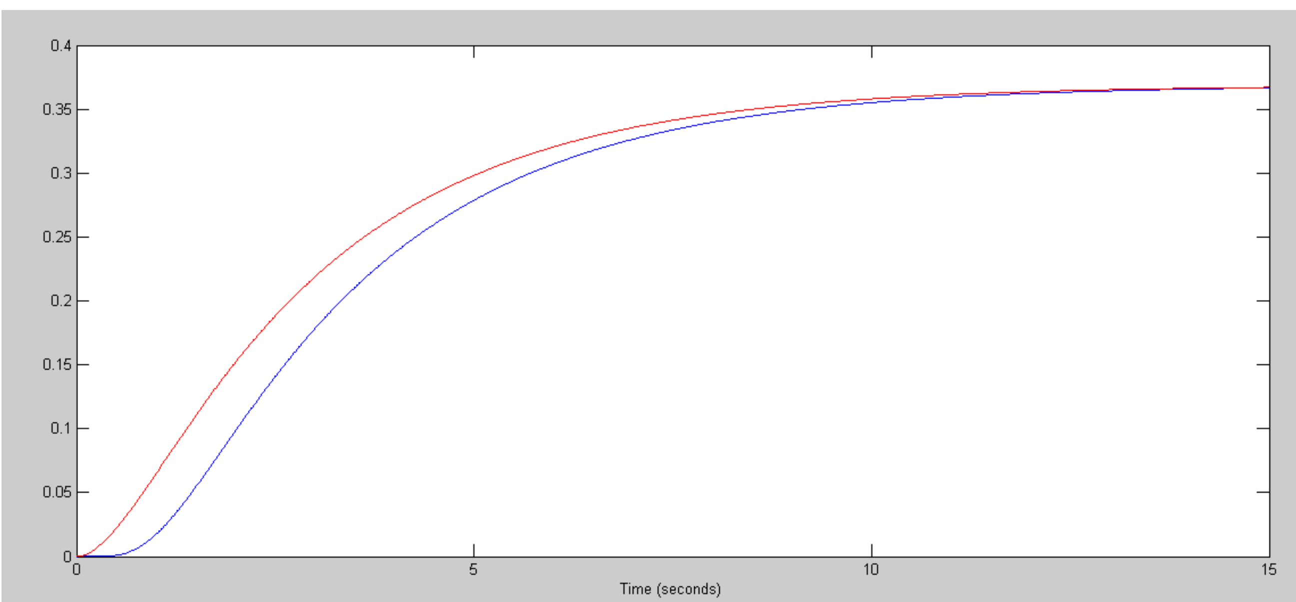
Keep the two dominant poles (in blue)

Match the DC gain

```
>> P = eig(G10)'

  -31.4892  -29.4198  -26.1758  -22.0455  -17.3957  -12.6397   -8.2000   -4.4712   -1.7845   -0.3787

>> G2 = zpk([],P(9:10),1);
>> k = evalfr(G10,0) / evalfr(G2,0)

k =    0.2492

>> G2 = zpk([],P(9:10),k);
>> zpk(G2)


      0.24915
--------------------
(s+1.784) (s+0.3787)
```

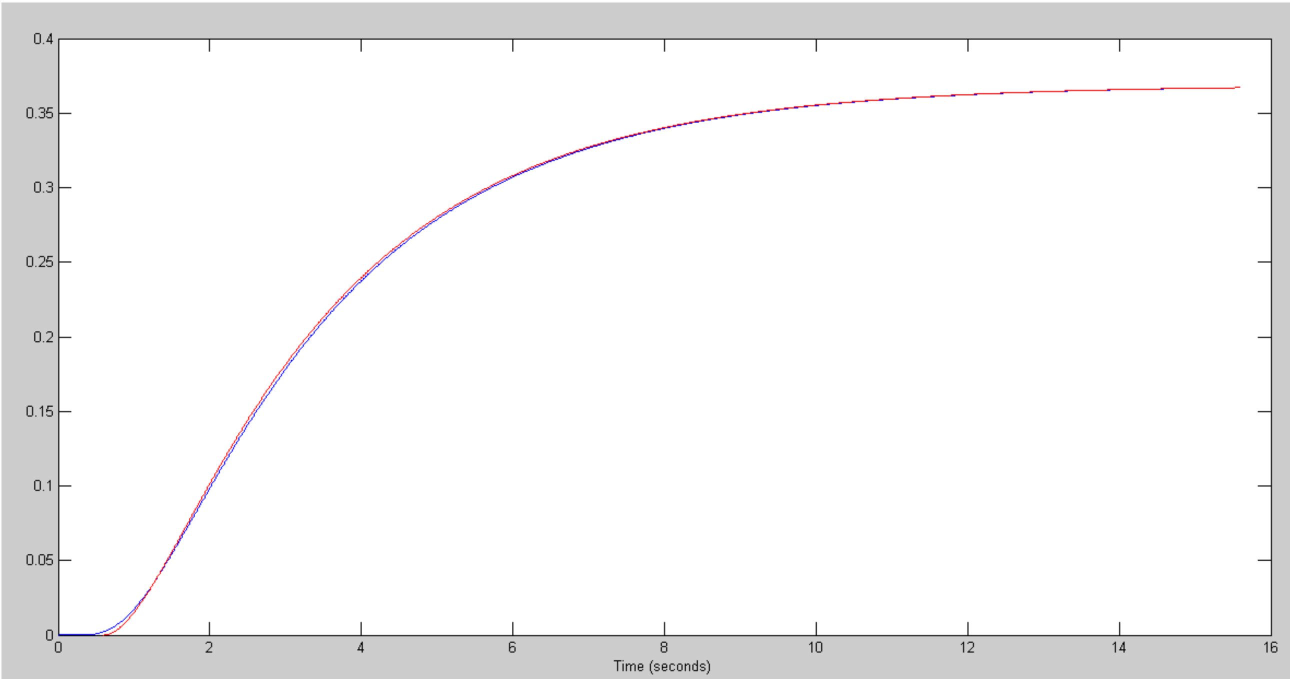
Plot the step responses

```
>> t = [0:0.01:15]';
>> y10 = step(G10,t);
>> y2 = step(G2,t);
>> plot(t,y10,'b',t,y2,'r');
>> xlabel('Time (seconds)');
```



Note: The fast poles that are ignored look like a 0.6 second delay

```
>> plot(t,y10,'b',t+0.6,y2,'r');
>> xlabel('Time (seconds)');
```



Step response of 10th-Order System (blue) & 2nd-Order Approximation with a 0.6s delay (red)

The is one way to get a delay in your model (ignored fast poles)

6) For the circuit for problem #4

- What initial condition will decay as slowly as possible?
- What initial condition will decay as fast as possible?

This is an eigenvector probem

```
>> [M,V] = eig(A);
>> M

M =

   -0.1286   -0.2459    0.3412    0.4063    0.4352    0.4255    0.3780    0.2969   -0.1894    0.0650
    0.2459    0.4063   -0.4255   -0.2969   -0.0650    0.1894    0.3780    0.4352   -0.3412    0.1286
   -0.3412   -0.4255    0.1894   -0.1894   -0.4255   -0.3412   -0.0000    0.3412   -0.4255    0.1894
    0.4063    0.2969    0.1894    0.4352    0.1286   -0.3412   -0.3780    0.0650   -0.4255    0.2459
   -0.4352   -0.0650   -0.4255   -0.1286    0.4063    0.1894   -0.3780   -0.2459   -0.3412    0.2969
    0.4255   -0.1894    0.3412   -0.3412   -0.1894    0.4255    0.0000   -0.4255   -0.1894    0.3412
   -0.3780    0.3780   -0.0000    0.3780   -0.3780   -0.0000    0.3780   -0.3780   -0.0000    0.3780
    0.2969   -0.4352   -0.3412    0.0650    0.2459   -0.4255    0.3780   -0.1286    0.1894    0.4063
   -0.1894    0.3412    0.4255   -0.4255    0.3412   -0.1894   -0.0000    0.1894    0.3412    0.4255
    0.0650   -0.1286   -0.1894    0.2459   -0.2969    0.3412   -0.3780    0.4063    0.4255    0.4352

>> eig(A)'

ans =

  -31.4892  -29.4198  -26.1758  -22.0455  -17.3957  -12.6397   -8.2000   -4.4712   -1.7845   -0.3787
```

The fast mode (red) decays quickly

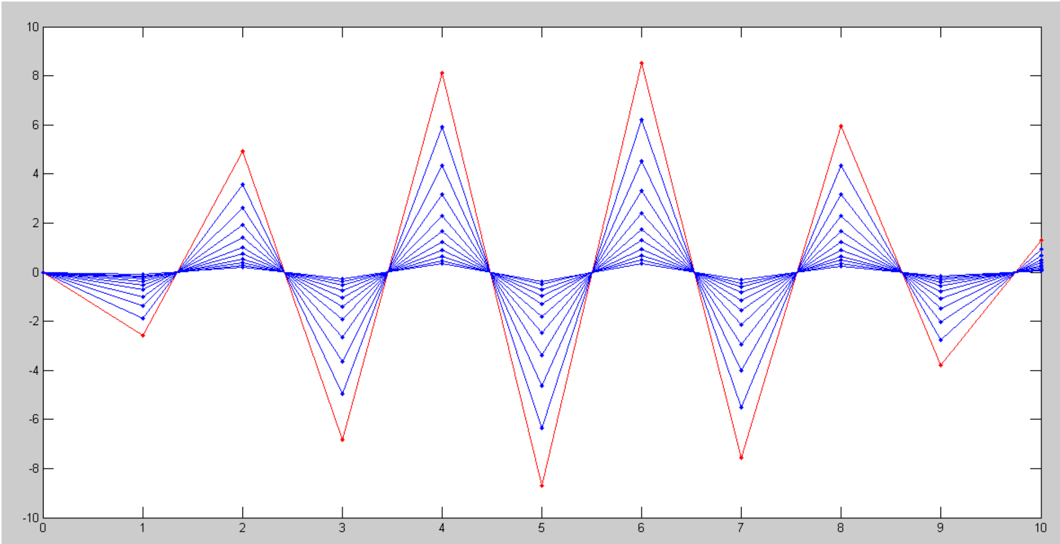
```
>> X0 = M(:,1) * 20

   -2.5728
    4.9171
   -6.8244
    8.1253
   -8.7043
    8.5099
   -7.5593
    5.9370
   -3.7872
    1.3009
```

The slow mode (blue) decays slowly

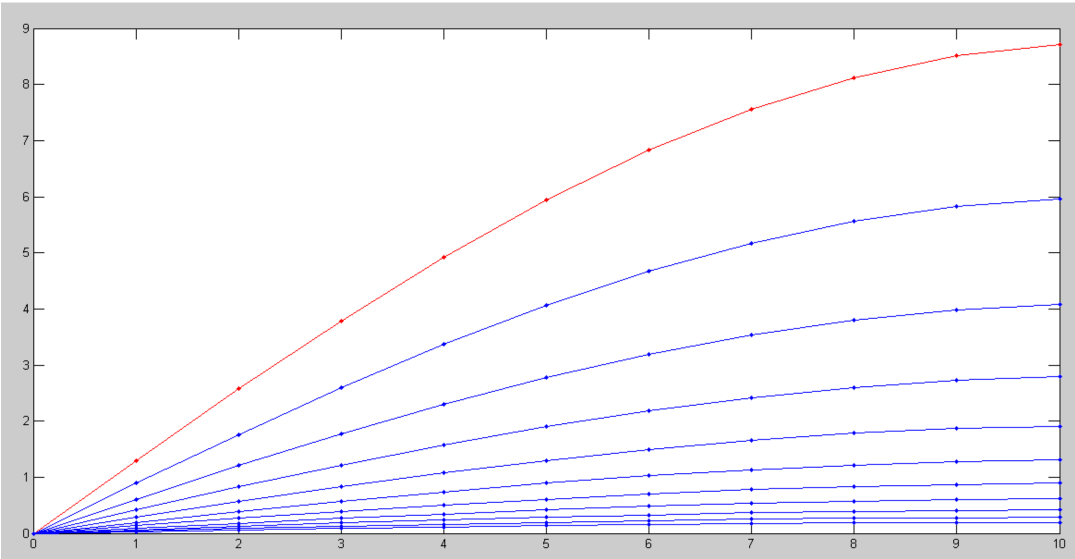
```
>> X0 = M(:,10) * 20

    1.3009
    2.5728
    3.7872
    4.9171
    5.9370
    6.8244
    7.5593
    8.1253
    8.5099
    8.7043
```

7) Modify the program *heat.m* to match the dynamics you calculated for this problem.

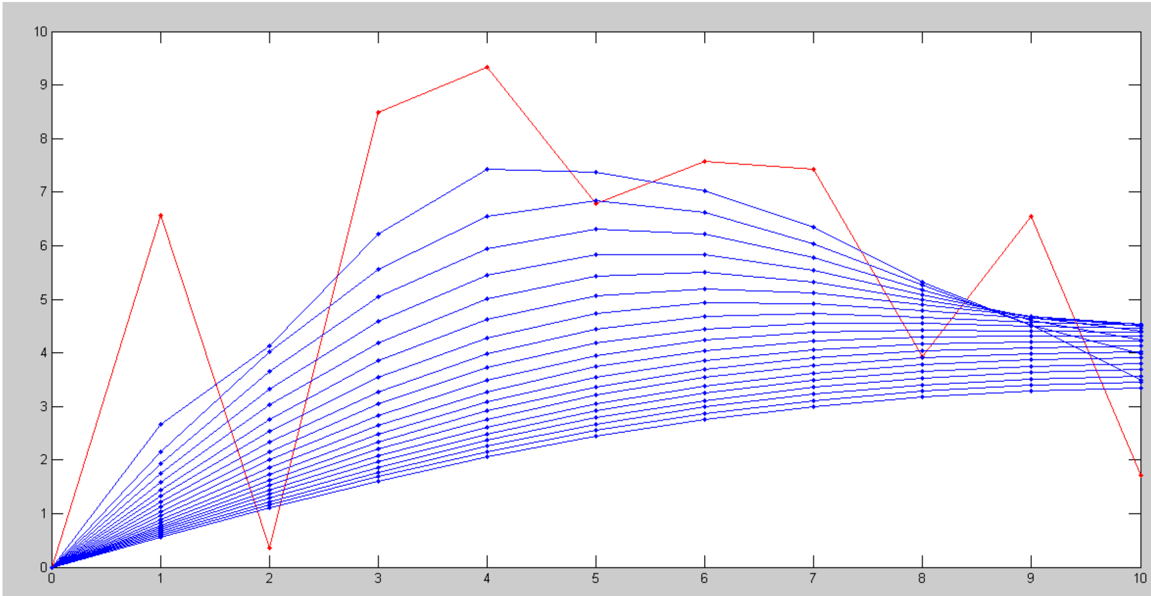
- Give the program listing
- Give the response for Vin = 0 and the initial conditions being
  - The slowest eigenvector
  - The fastest eigenvector
  - A random set of voltages



Fast mode: Voltages plotted every 10ms
Note: The shape stays the same (the eigenvector). The amplitide decays quickly (the eigenvalue)



Slow Mode: Votlages Plotted every 1.00 second
Note: The shape stays the same (the eigenvector). The amplitide decays slowly (the eigenvalue)



Random Initial Conditions.
Fast modes decay quickly, leaving the slow (dominant) mode

Code:

```
% 10-stage RC Filter

% V = M(:,10) * 20;
% V = M(:,1) * 20;
V = 10*rand(10,1)

dV = zeros(10,1);
V0 = 0;
dt = 0.001;
t = 0;
i = 0;

y = [];

hold off
plot([0,10],[0,100],'w.');
plot([0:10], [V0;V], 'r.-');
pause(0.01);

hold on

while(t < 2)
   dV(1)  = 8*V0   - 16.2*V(1) + 8*V(2);
   dV(2)  = 8*V(1) - 16.2*V(2) + 8*V(3);
   dV(3)  = 8*V(2) - 16.2*V(3) + 8*V(4);
   dV(4)  = 8*V(3) - 16.2*V(4) + 8*V(5);
   dV(5)  = 8*V(4) - 16.2*V(5) + 8*V(6);
   dV(6)  = 8*V(5) - 16.2*V(6) + 8*V(7);
   dV(7)  = 8*V(6) - 16.2*V(7) + 8*V(8);
   dV(8)  = 8*V(7) - 16.2*V(8) + 8*V(9);
   dV(9)  = 8*V(8) - 16.2*V(9) + 8*V(10);
   dV(10) = 8*V(9) - 8.2*V(10);

   V = V +  dV*dt;
   t = t + dt;

   y = [y ; V'];
   i = mod(i + 1, 100);
   if(i == 0)
      plot([0:10], [V0;V], '.-');
      pause(0.01);
   end

   end
```