Matlab Code

```
% Hunter Phillips
% Homework 9
% MAE 488
% 04/17/19

clc
clear
format compact

%% Header
d_bullets = repmat('*', 50, 1); % concise way to make a lot of chars
fprintf('%c',d_bullets)
fprintf('\nMAE 488, Homework #9, Spring 2019, Hunter Phillips\n')
fprintf('%c',d_bullets)
fprintf('\n\n')

clear

%% Problem 11.2M

su_bullets = repmat('*', 25, 1); % setting up cmd line output
un_bullets = repmat('-', 25, 1);
fprintf('%c',su_bullets)
fprintf('\nProblem 11.2M\n')
fprintf('%c',su_bullets)
fprintf('\n\n')

f1 = figure(1);
hold on

% Root Locus Plot
num = [1 0];
den = [3 0 12];
h = tf(num,den)
rlocus(h)

title({'MAE 488, Homework 9, Problem 11.2M'},'interpreter','latex','FontSize',16)


% print(f1,'..\results\problem_10_29_1.png','-dpng','-r1200');

%% Problem 11.6M

su_bullets = repmat('*', 25, 1); % setting up cmd line output
un_bullets = repmat('-', 25, 1);
fprintf('%c',su_bullets)
fprintf('\nProblem 11.6M\n')
fprintf('%c',su_bullets)
fprintf('\n\n')

% Multiply Gp_s and Gc_s as given to obtain:
Gc_Gp_s = tf([1 12 30 100],[1 3 2 0])
poles_a = pole(Gc_Gp_s)
zeros_a = zero(Gc_Gp_s)

fprintf('K = (TD)(KP) = 0.5KP\n\n')

% Finding closed-loop transfer function

modifi = series(Gc_Gp_s,10);    % Kp
modifi_2 = series(modifi,0.5);  % 0.5
Ys_Rs = feedback(modifi_2,1)    % closed loop feedback

poles_c = pole(Ys_Rs)
zeros_c = zero(Ys_Rs)


%% Problem 11.9M

su_bullets = repmat('*', 25, 1); % setting up cmd line output
un_bullets = repmat('-', 25, 1);
fprintf('%c',su_bullets)
fprintf('\nProblem 11.9M\n')
fprintf('%c',su_bullets)
```

```matlab
fprintf('\n\n')

f2 = figure(2);
hold on

% Rearranging results in:
% 1 + (5p/4)*((s^2+6s+8)/(s(s^2+(25/4)s+4)))=0
t_fun = tf([1 6 8],[1 (25/4) 4 0])
poles_9_b = pole(t_fun)
zeros_9_b = zero(t_fun)
rlocus(t_fun)
ylim([-3 3])
title({'MAE 488, Homework 9, Problem 11.9M'},'interpreter','latex','FontSize',16)

%% Problem 11.45M

su_bullets = repmat('*', 25, 1); % setting up cmd line output
un_bullets = repmat('-', 25, 1);
fprintf('%c',su_bullets)
fprintf('\nProblem 11.45M\n')
fprintf('%c',su_bullets)
fprintf('\n\n')

f3 = figure(3);
hold on

% Solving mass spring damper system for equations of motion
% 2x'' + 8x' + (k2+26)*x = (k2)*y
% CE = 2s^2 + 8s + 26 + k2 = 0
% Standard form = 1 + ((k2)/2)(1/(s^2+4s+13) = 0
tfunn = tf([1],[1 4 13])
rlocus(tfunn)
title({'MAE 488, Homework 9, Problem 11.45M'},'interpreter','latex','FontSize',16)

largest_damp_ratio = cos(atan(3/2)) % info from pole on plot for 2 and 3
```

<u>Matlab Output</u>

```
************************************************
MAE 488, Homework #9, Spring 2019, Hunter Phillips
************************************************


************************
Problem 11.2M
************************
h =


     s
  ----------
  3 s^2 + 12


Continuous-time transfer function.


************************
Problem 11.6M
************************


Gc_Gp_s =


  s^3 + 12 s^2 + 30 s + 100
  -------------------------
      s^3 + 3 s^2 + 2 s


Continuous-time transfer function.
```

poles_a =

0

-2

-1

zeros_a =

-10.0000 + 0.0000i

-1.0000 + 3.0000i

-1.0000 - 3.0000i

K = (TD)(KP) = 0.5KP

Ys_Rs =

```
  5 s^3 + 60 s^2 + 150 s + 500
  ----------------------------
  6 s^3 + 63 s^2 + 152 s + 500
```

Continuous-time transfer function.

poles_c =

-8.6881 + 0.0000i

-0.9059 + 2.9616i

-0.9059 - 2.9616i

```
zeros_c =
 -10.0000 + 0.0000i
  -1.0000 + 3.0000i
  -1.0000 - 3.0000i
```

****

Problem 11.9M

****

```
t_fun =

     s^2 + 6 s + 8
  -------------------
  s^3 + 6.25 s^2 + 4 s

Continuous-time transfer function.

poles_9_b =
         0
   -5.5262
   -0.7238

zeros_9_b =
    -4
    -2
```

****

Problem 11.45M

****

```
tfunn =

        1
  --------------
  s^2 + 4 s + 13

Continuous-time transfer function.

largest_damp_ratio =
    0.5547
```

Matlab Figures

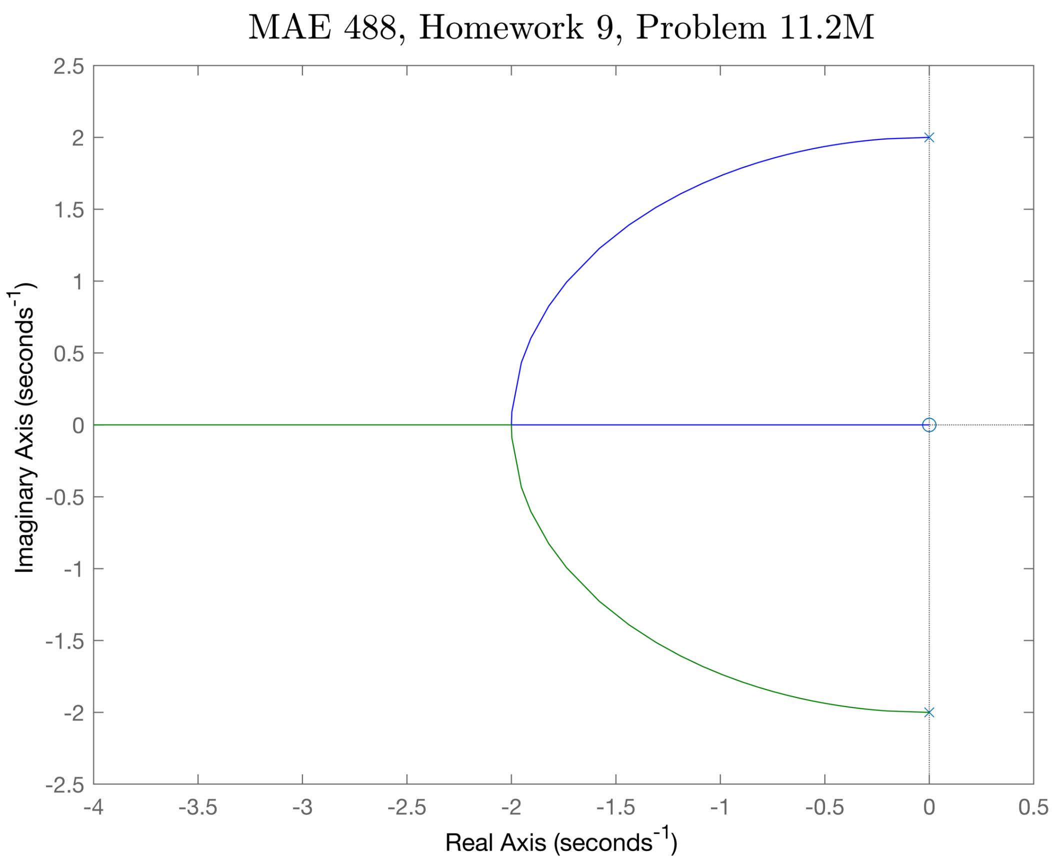

Figure 1: Problem 11.2

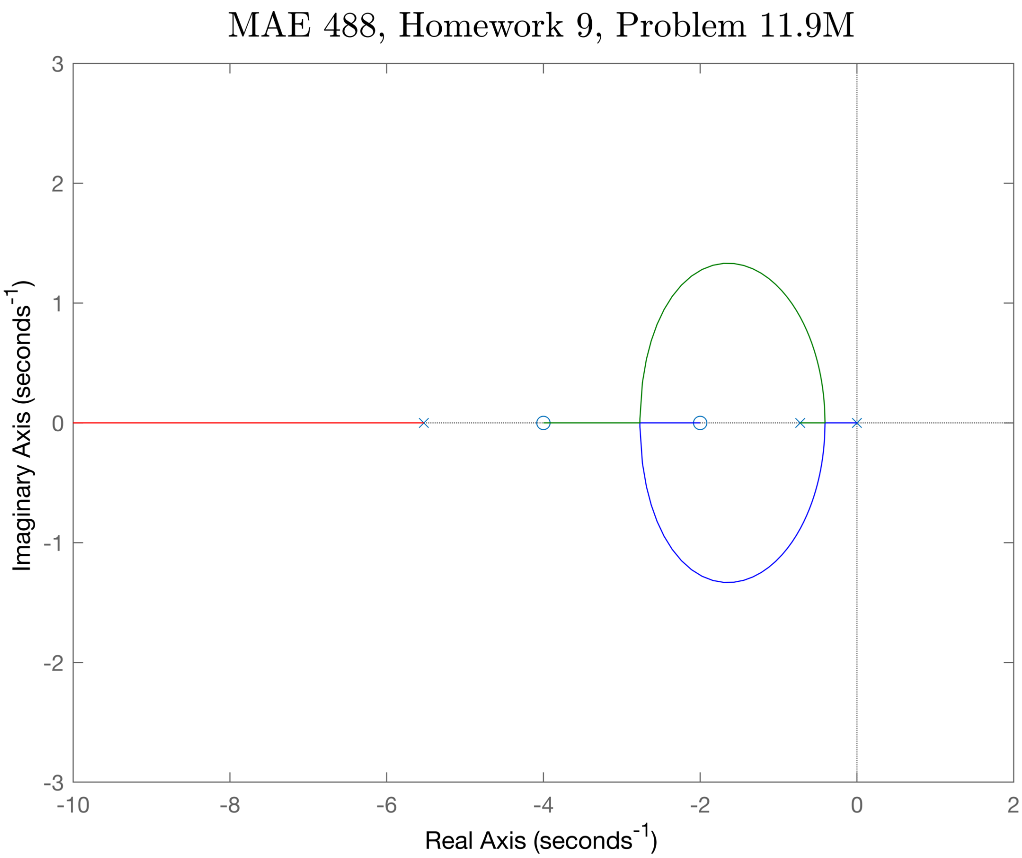

Figure 2: Problem 11.9

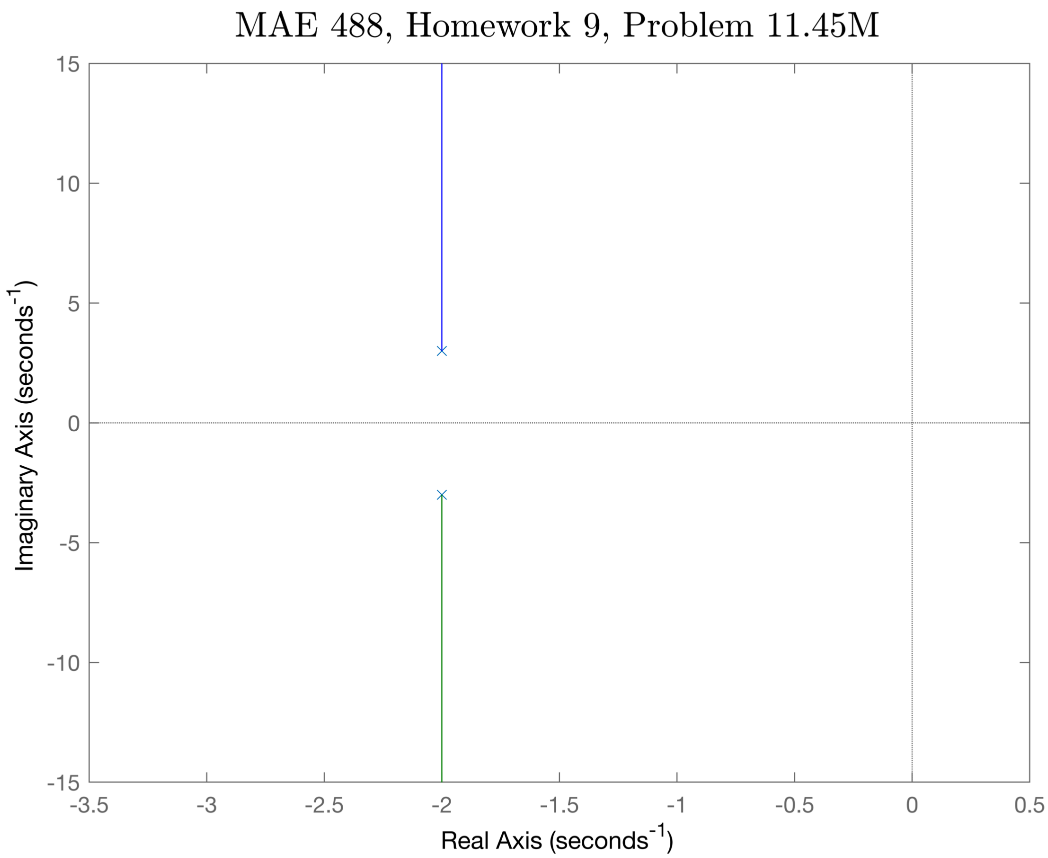

Figure 3: Problem 11.45