Table of Contents

QUESTION 2

DO NOT REMOVE THE LINE BELOW MAKE SURE 'eel3135_lab07_comment.m' IS IN THE SAME DIRECTORY AS THIS FILE

```
clear; close all; clc;
type('eel3135_lab07_comment.m')


%% QUESTION #1 COMMENTING
clear
close all
clc

%% DEFINE FILTER AND INPUT
N = 100;
n = 0:(N-1);

% FILTER
b = (1/4)*[1 1 1];
a = [1 -1 0.5 0.25];
% <-- Answer: What is the numerator of this filter's transfer
 function?
% b is the numerator = 1/4(z^-1 + z^-2 + z^-3)
```

```
% <-- Answer: What is the denominator of this filter's transfer
 function?
% a is the denominator = z^-1 - z^-2 + 0.5z^-3 + 0.25z^-4
% Hint: look a later when we implement the filter with the "filter"
 command
%  (use "help filter" to see what it does)
%

% INPUT 1
x1 = zeros(N,1);
x1(1) = 1;
% <-- Answer: (True or False) x1 is an impulse excitation? If false,
% describe the excitation.
%

% INPUT 2
x2 = zeros(N,1);
x2(1:12)  = cos(3*pi/2*   n(1:12));
x2(13:24) = cos(pi/4*   n(13:24));
x2(25:36) = cos(2*pi*   n(25:36));
x2(37:48) = cos(3*pi/4* n(37:48));
x2(49:60) = cos(pi*     n(49:60));
x2(61:72) = cos(pi/8* n(61:72));
x2(73:84) = cos(pi/2*   n(73:84));
% <-- Answer: (True or False) x2 is an single frequency excitation?
 If
% false, describe the excitation.
%

%% DEFINE AND PLOT OUTPUT

% OUTPUT 1
y1 = filter(b,a,x1);

% OUTPUT 2
y2 = filter(b,a,x2);


% PLOT THE IMPULSE RESPONSE AND DTFT
figure(1)
subplot(311)
stem(n,x1)
xlabel('Time (Samples)')
ylabel('x_1[n]')
subplot(312)
stem(n,y1)
xlabel('Time (Samples)')
ylabel('y_1[n]')
subplot(313)
pzplot(b,a)
axis equal

% PLOT THE IMPULSE RESPONSE AND DTFT
figure(2)
```

```
subplot(311)
stem(n,x2)
xlabel('Time (Samples)')
ylabel('x_2[n]')
subplot(312)
stem(n,y2)
xlabel('Time (Samples)')
ylabel('y_2[n]')
subplot(313)
pzplot(b,a)
axis equal



% ====================================================================
% NOTE: YOU DO NOT NEED TO ADD COMMENTS IN THE CODE BELOW. WE JUST
% NEEDED POLE-ZERO PLOTTING CODE AND THUS WROTE IT.
% ====================================================================
function pzplot(b,a)
% PZPLOT(B,A)  plots the pole-zero plot for the filter described by
% vectors A and B.  The filter is a "Direct Form II Transposed"
% implementation of the standard difference equation:
%
%    a(1)*y(n) = b(1)*x(n) + b(2)*x(n-1) + ... + b(nb+1)*x(n-nb)
%                          - a(2)*y(n-1) - ... - a(na+1)*y(n-na)
%

    % MODIFY THE POLYNOMIALS TO FIND THE ROOTS
    b1 = zeros(max(length(a),length(b)),1); % Need to add zeros to get
 the right roots
    a1 = zeros(max(length(a),length(b)),1); % Need to add zeros to get
 the right roots
    b1(1:length(b)) = b;    % New a with all values
    a1(1:length(a)) = a;    % New a with all values

    % FIND THE ROOTS OF EACH POLYNOMIAL AND PLOT THE LOCATIONS OF THE
 ROOTS
    h1 = plot(real(roots(a1)), imag(roots(a1)));
    hold on;
    h2 = plot(real(roots(b1)), imag(roots(b1)));
    hold off;

    % DRAW THE UNIT CIRCLE
    circle(0,0,1)

    % MAKE THE POLES AND ZEROS X's AND O's
    set(h1, 'LineStyle', 'none', 'Marker', 'x',
'MarkerFaceColor','none', 'linewidth', 1.5, 'markersize', 8);
    set(h2, 'LineStyle', 'none', 'Marker', 'o',
'MarkerFaceColor','none', 'linewidth', 1.5, 'markersize', 8);
    axis equal;

    % DRAW VERTICAL AND HORIZONTAL LINES
    xminmax = xlim();
```

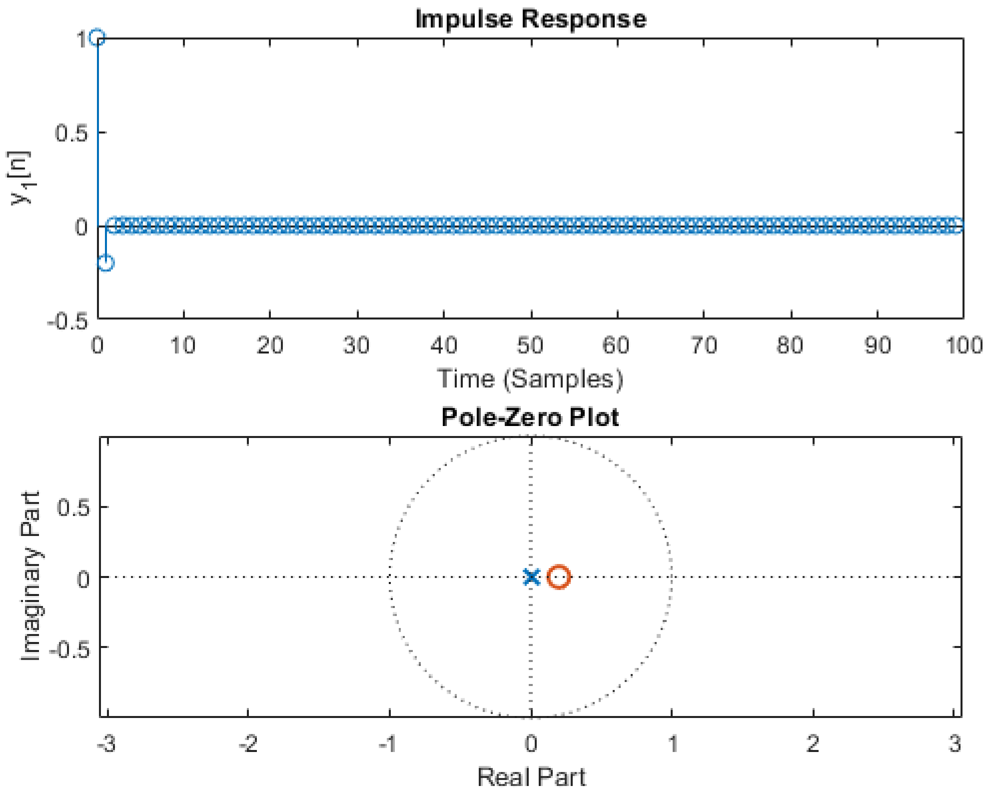
```
    yminmax = ylim();
    line([xminmax(1) xminmax(2)],[0 0], 'linestyle', ':', 'linewidth',
 0.5, 'color', [1 1 1]*.1)
    line([0 0],[yminmax(1) yminmax(2)], 'linestyle', ':', 'linewidth',
 0.5, 'color', [1 1 1]*.1)

    % ADD LABELS AND TITLE
    xlabel('Real Part')
    ylabel('Imaginary Part')
    title('Pole-Zero Plot')

end


function circle(x,y,r)
% CIRCLE(X,Y,R)  draws a circle with horizontal center X, vertical
 center
% Y, and radius R.
%

    % ANGLES TO DRAW
    ang=0:0.01:2*pi;

    % DEFINE LOCATIONS OF CIRCLE
    xp=r*cos(ang);
    yp=r*sin(ang);

    % PLOT CIRCLE
    hold on;
    plot(x+xp,y+yp, ':', 'linewidth', 0.5, 'color', [1 1 1]*.1);
    hold off;

end
```

QUESTION 2: Z-TRANSFORM

2 (a) PLOT IMPULSE RESPONSE AND POLE-ZERO PLOT

```
N = 100;
n = 0:(N-1);

% FILTER
b = [1 -0.2];
a = 1;

% INPUT 1
x1 = zeros(N,1);
x1(1) = 1;

% OUTPUT 1
```

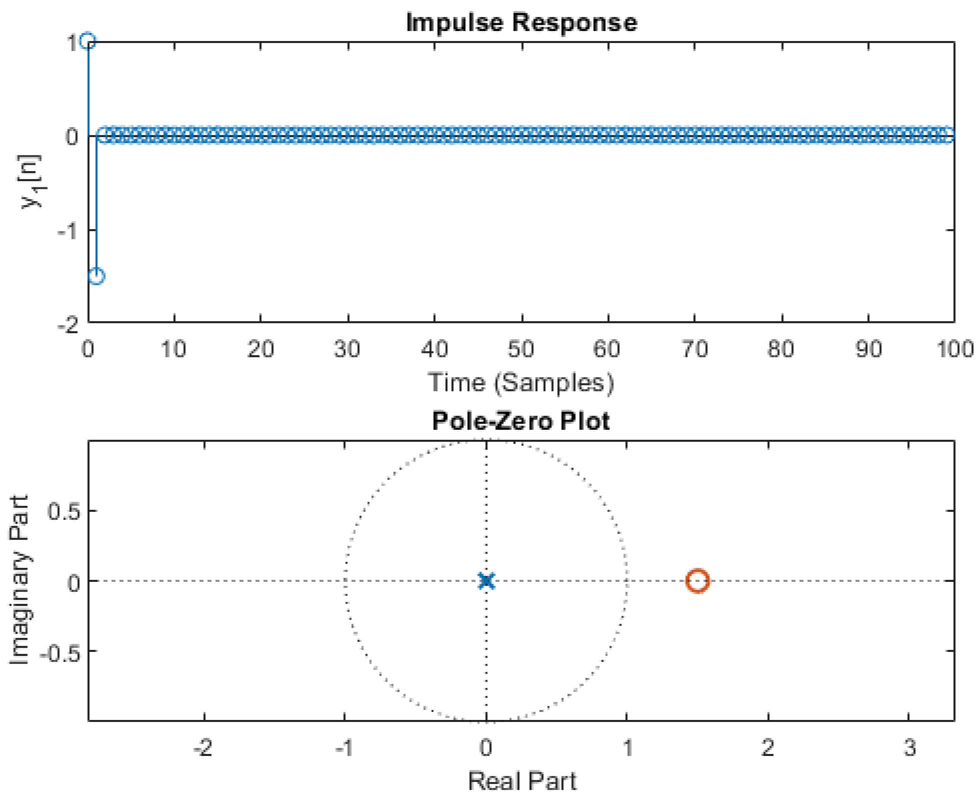
```
y1 = filter(b,a,x1);

subplot(211)
stem(n,y1)
xlabel('Time (Samples)')
ylabel('y_1[n]')
title('Impulse Response')
subplot(212)
pzplot(b,a)
axis equal
```



2 (b) PLOT IMPULSE RESPONSE AND POLE-ZERO PLOT

```
N = 100;
n = 0:(N-1);

% FILTER
b = [1 -1.5];
a = 1;

% INPUT 1
x1 = zeros(N,1);
x1(1) = 1;
```

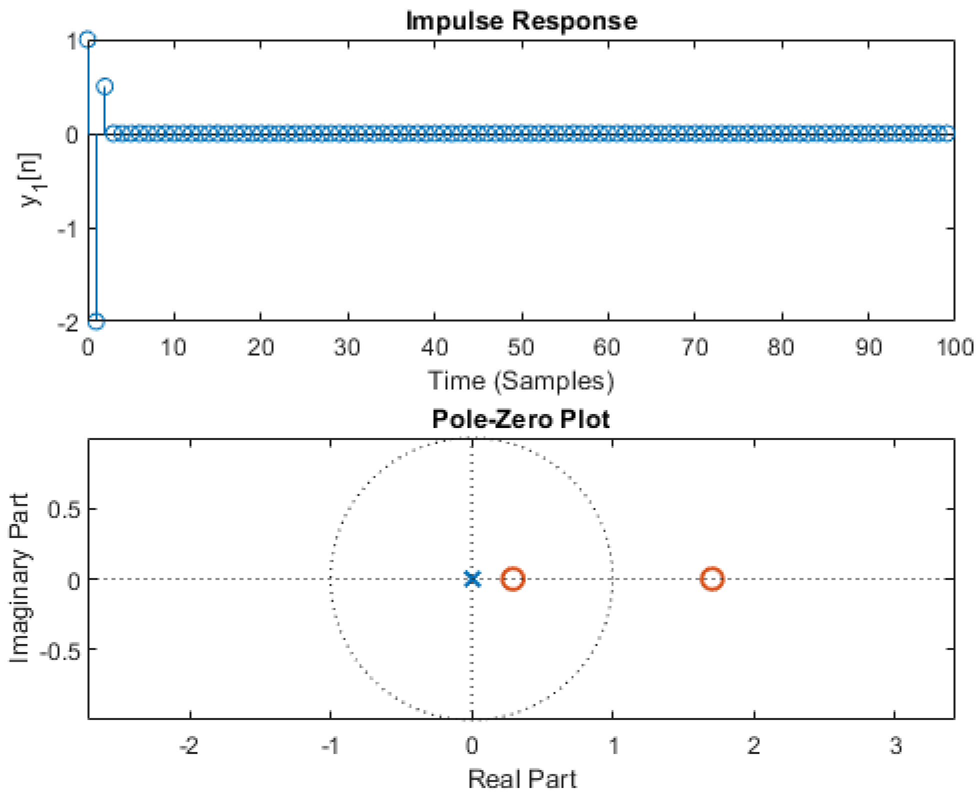
```
% OUTPUT 1
y1 = filter(b,a,x1);

subplot(211)
stem(n,y1)
xlabel('Time (Samples)')
ylabel('y_1[n]')
title('Impulse Response')
subplot(212)
pzplot(b,a)
axis equal
```



2 (c) PLOT IMPULSE RESPONSE AND POLE-ZERO PLOT

```
N = 100;
n = 0:(N-1);

% FILTER
b = [1 -2 0.5];
a = 1;

% INPUT 1
x1 = zeros(N,1);
x1(1) = 1;
```

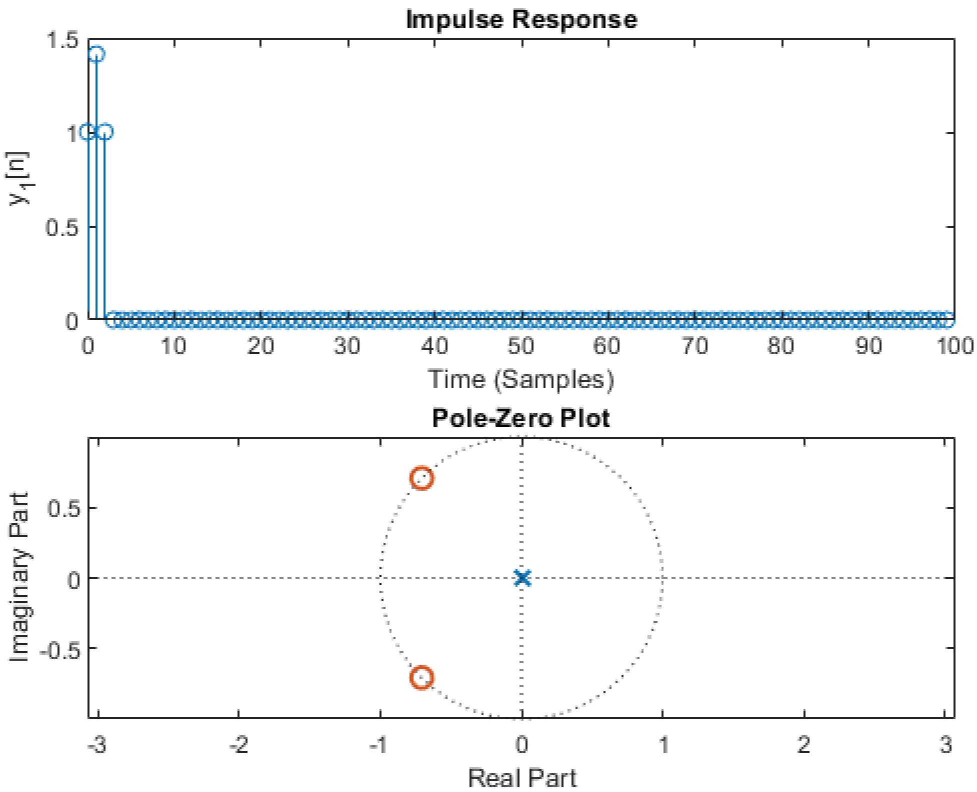
```
% OUTPUT 1
y1 = filter(b,a,x1);

subplot(211)
stem(n,y1)
xlabel('Time (Samples)')
ylabel('y_1[n]')
title('Impulse Response')
subplot(212)
pzplot(b,a)
axis equal
```



2 (d) PLOT IMPULSE RESPONSE AND POLE-ZERO PLOT

```
N = 100;
n = 0:(N-1);

% FILTER
b = [1 sqrt(2) 1];
a = 1;

% INPUT 1
x1 = zeros(N,1);
```

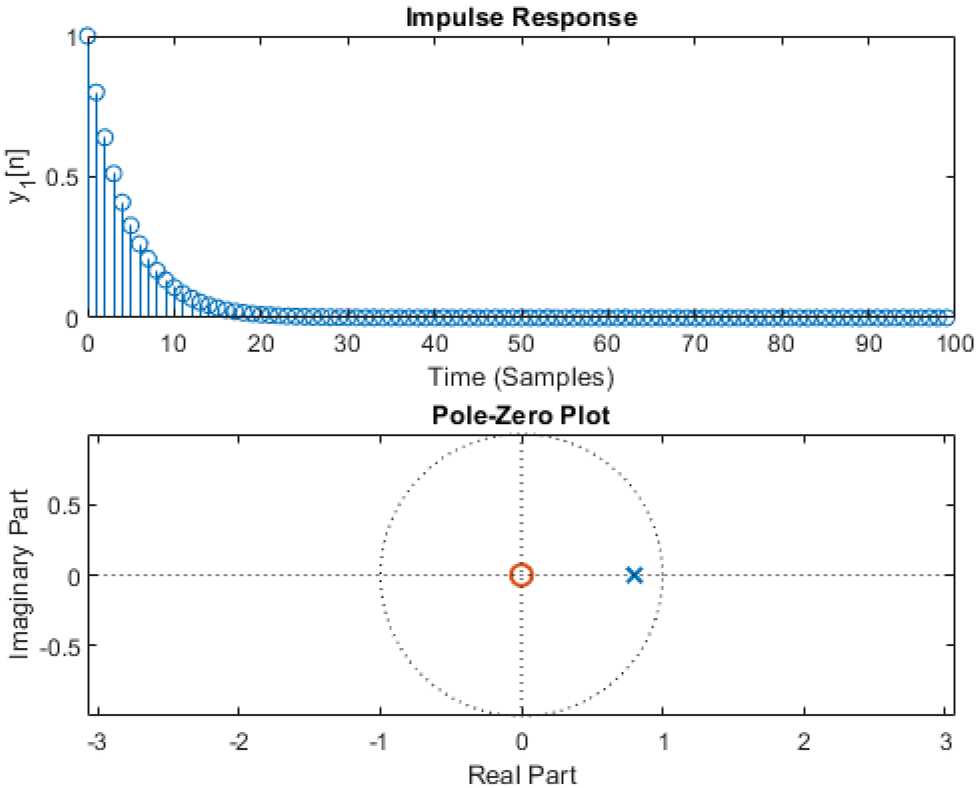
```
x1(1) = 1;

% OUTPUT 1
y1 = filter(b,a,x1);

subplot(211)
stem(n,y1)
xlabel('Time (Samples)')
ylabel('y_1[n]')
title('Impulse Response')
subplot(212)
pzplot(b,a)
axis equal
```



2 (e) PLOT IMPULSE RESPONSE AND POLE-ZERO PLOT

```
N = 100;
n = 0:(N-1);

% FILTER
b = 1;
a = [1 -0.8];

% INPUT 1
```

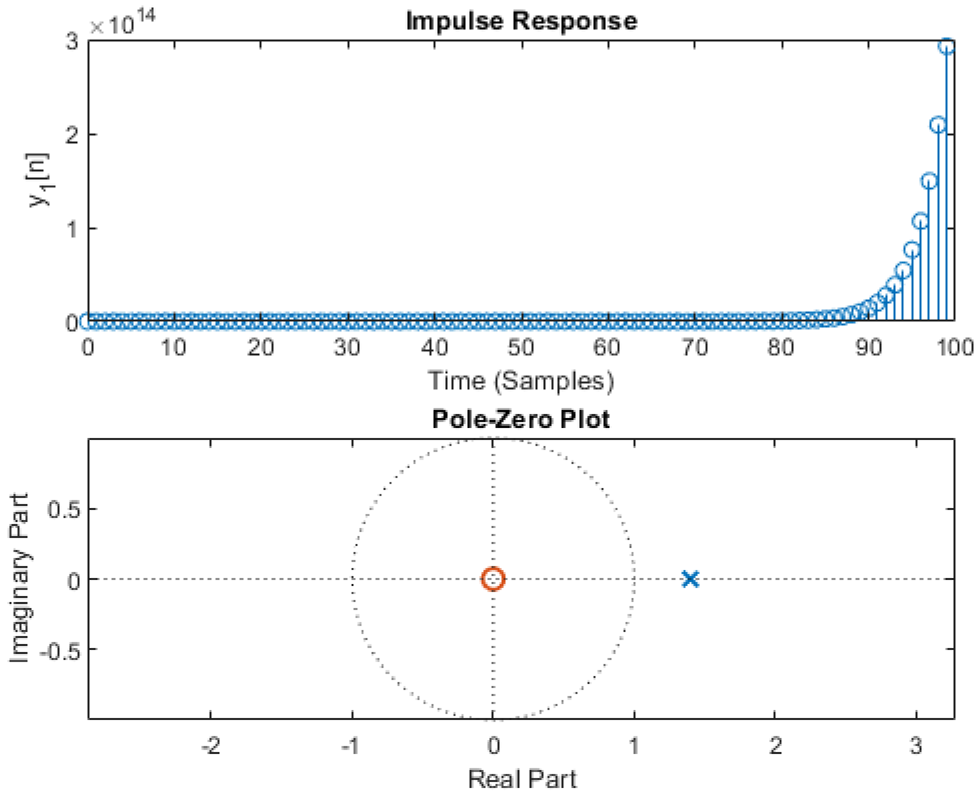
```
x1 = zeros(N,1);
x1(1) = 1;

% OUTPUT 1
y1 = filter(b,a,x1);

subplot(211)
stem(n,y1)
xlabel('Time (Samples)')
ylabel('y_1[n]')
title('Impulse Response')
subplot(212)
pzplot(b,a)
axis equal
```



2 (f) PLOT IMPULSE RESPONSE AND POLE-ZERO PLOT

```
N = 100;
n = 0:(N-1);

% FILTER
b = 1;
a = [1 -1.4];
```

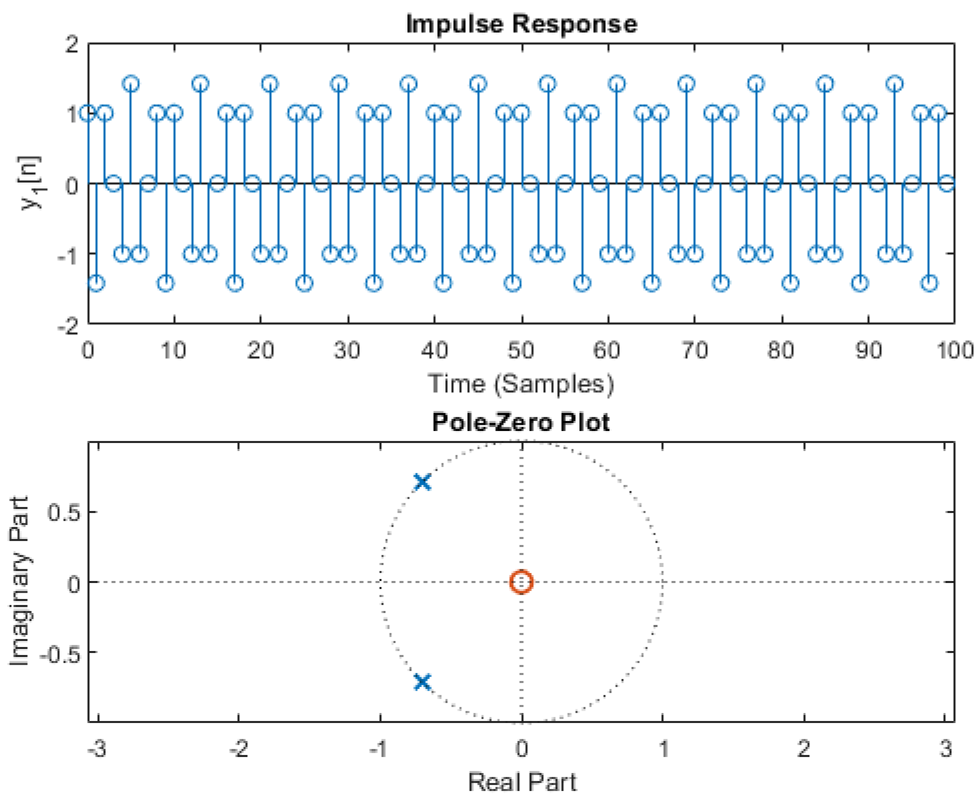
```
% INPUT 1
x1 = zeros(N,1);
x1(1) = 1;

% OUTPUT 1
y1 = filter(b,a,x1);

subplot(211)
stem(n,y1)
xlabel('Time (Samples)')
ylabel('y_1[n]')
title('Impulse Response')
subplot(212)
pzplot(b,a)
axis equal
```



2 (g) PLOT IMPULSE RESPONSE AND POLE-ZERO PLOT

```
N = 100;
n = 0:(N-1);

% FILTER
b = 1;
a = [1 sqrt(2) 1];
```

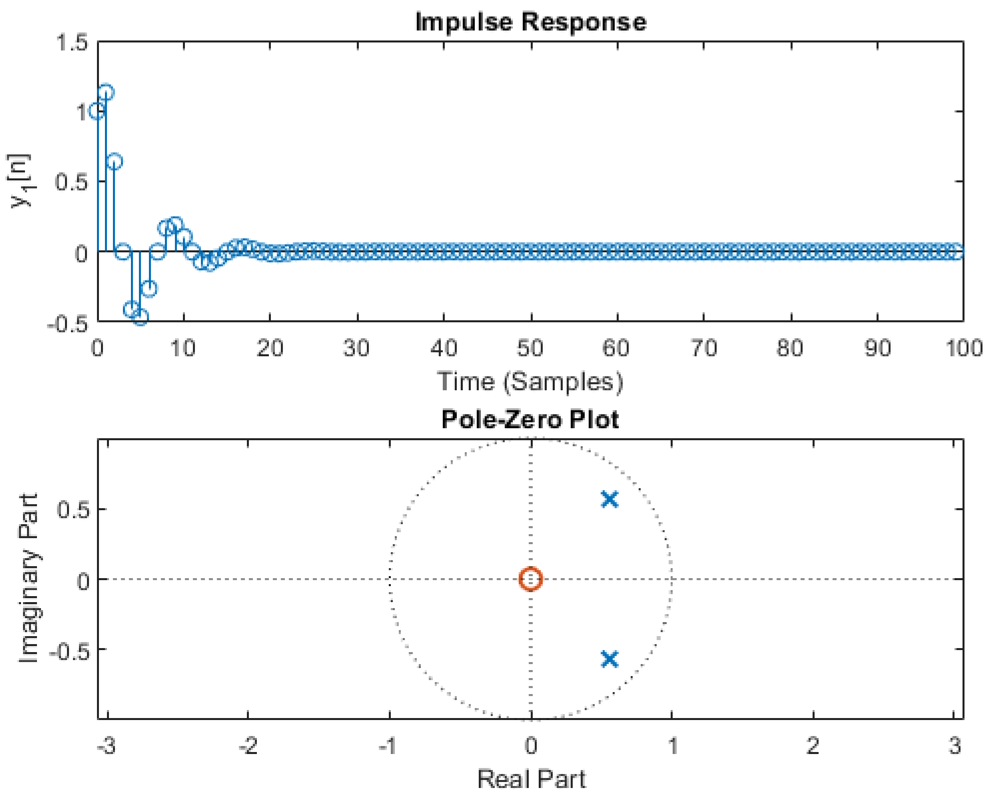
```
% INPUT 1
x1 = zeros(N,1);
x1(1) = 1;

% OUTPUT 1
y1 = filter(b,a,x1);

subplot(211)
stem(n,y1)
xlabel('Time (Samples)')
ylabel('y_1[n]')
title('Impulse Response')
subplot(212)
pzplot(b,a)
axis equal
```



2 (h) PLOT IMPULSE RESPONSE AND POLE-ZERO PLOT

```
N = 100;
n = 0:(N-1);

% FILTER
b = 1;
```

```
a = [1 -1.13137 0.64];

% INPUT 1
x1 = zeros(N,1);
x1(1) = 1;

% OUTPUT 1
y1 = filter(b,a,x1);

subplot(211)
stem(n,y1)
xlabel('Time (Samples)')
ylabel('y_1[n]')
title('Impulse Response')
subplot(212)
pzplot(b,a)
axis equal
```



QUESTION 3: MORE Z-TRANSFORM

3 (a) ANSWER QUESTION

The pole-zero conditions which the impulse response is unstable is when the poles are outside the unit cirlce

3 (b) ANSWER QUESTION

The pole-zero conditions which the impulse response is stable is when the poles are inside the unit cirlce

3 (c) ANSWER QUESTION

A system is critically stable if oscillations of the output continue forever

3 (d) ANSWER QUESTION

A system is finite if their is only a numerator

3 (e) ANSWER QUESTION

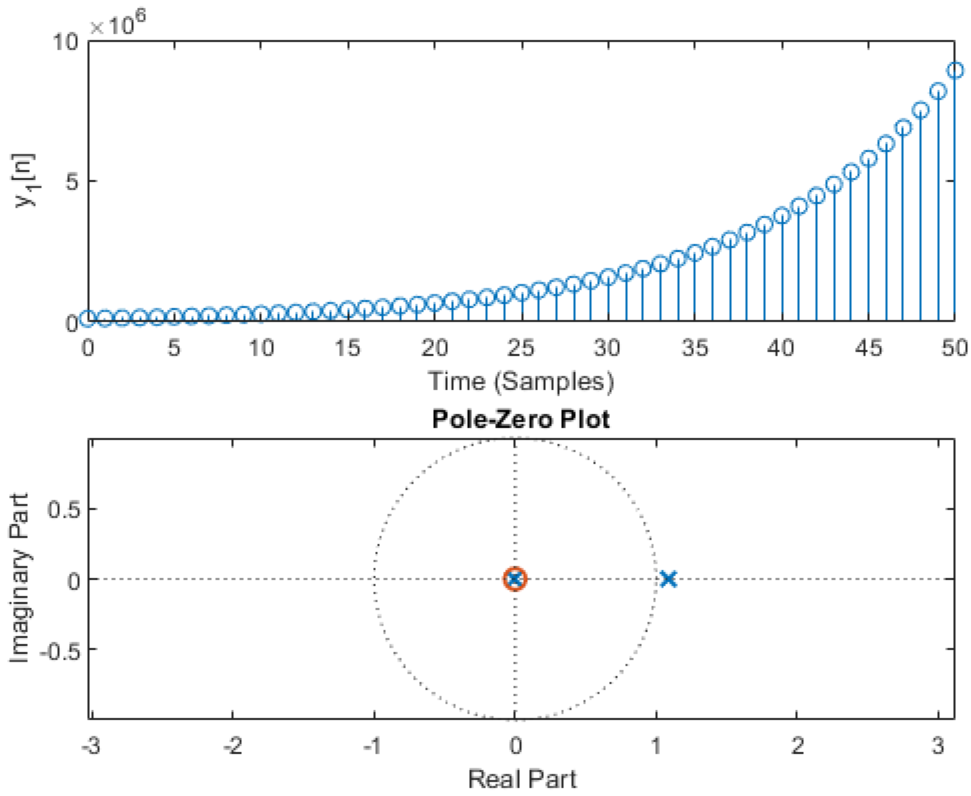
A system is infinite if their is is a non-trivial denominator (Not 1)

3 (f) ANSWER QUESTION

QUESTION 4: LOAN DIFFERENCE EQUATION

4 (a) PLOT OUTPUT AND POLE-ZERO PLOT

```
N = 51;
n = 0:(N-1);

% FILTER
b = 1;
a = [1 -1.09 0 0];

% INPUT 1
x1 = zeros(N,1);
x1(1) = 120000;

% OUTPUT 1
y1 = filter(b,a,x1);

subplot(211)
stem(n,y1)
xlabel('Time (Samples)')
ylabel('y_1[n]')
subplot(212)
pzplot(b,a)
axis equal
```



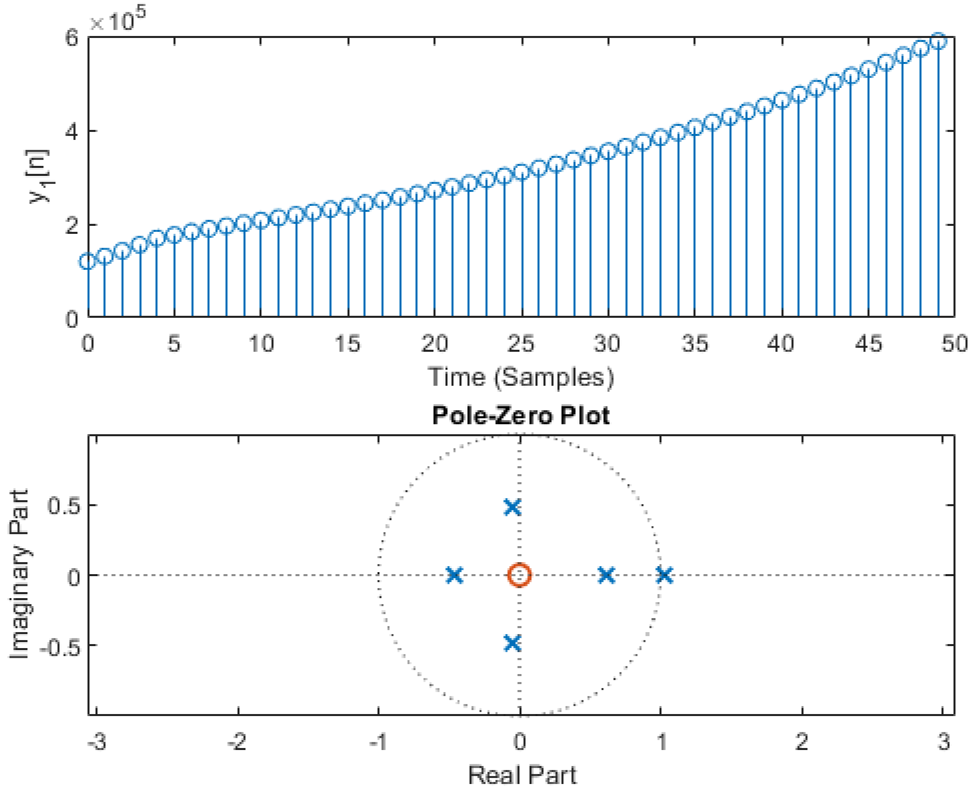
4 (b) ANSWER QUESTION

It would take roughly 25 years to owe 1 million dollars

4 (c) PLOT OUTPUT AND POLE-ZERO PLOT

```
N = 50;
n = 0:(N-1);

% FILTER
b = 1;
a = [1 -1.09 0 0 0 0.07];

% INPUT 1
x1 = zeros(N,1);
x1(1) = 120000;

% OUTPUT 1
y1 = filter(b,a,x1);

subplot(211)
stem(n,y1)
xlabel('Time (Samples)')
ylabel('y_1[n]')
subplot(212)
```

```
pzplot(b,a)
axis equal
```



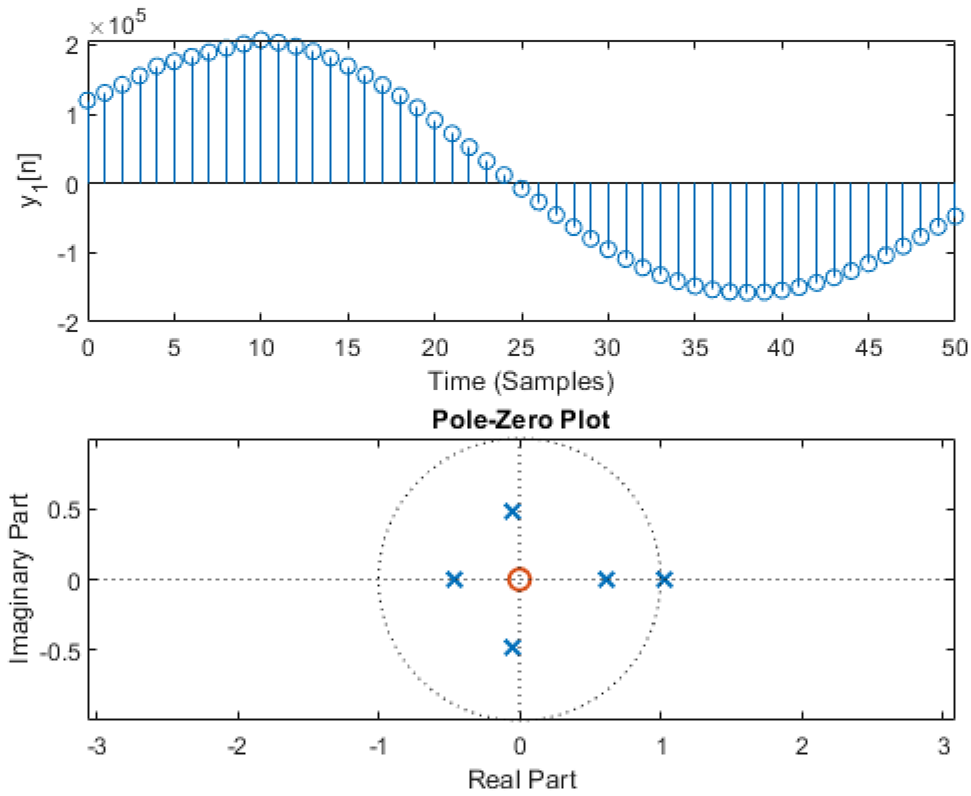
4 (d) ANSWER QUESTION

The loan amount goes towards infinity. It would take 69 years to owe 1 million dollars

4 (e) PLOT OUTPUT AND POLE-ZERO PLOT

```
N = 51;
n = 0:(N-1);

% FILTER
b = 1;
a = [1 -1.09 0 0 0 0.07 0 0 0 0 0 0.08];

% INPUT 1
x1 = zeros(N,1);
x1(1) = 120000;

% OUTPUT 1
y1 = filter(b,a,x1);

subplot(211)
stem(n,y1)
```

```
xlabel('Time (Samples)')
ylabel('y_1[n]')
subplot(212)
pzplot(b,a)
axis equal
```



4 (f) ANSWER QUESTION

The loan amount goes towards 0. The loan is paid off at 25 years.

ALL FUNCTIONS SUPPORTING THIS CODE % %

== NOTE: YOU DO NOT NEED TO ADD COMMENTS IN THE CODE BELOW. WE JUST NEEDED POLE-ZERO PLOTTING CODE AND THUS WROTE IT. ==

```
function pzplot(b,a)
% PZPLOT(B,A)  plots the pole-zero plot for the filter described by
% vectors A and B.  The filter is a "Direct Form II Transposed"
% implementation of the standard difference equation:
%
%    a(1)*y(n) = b(1)*x(n) + b(2)*x(n-1) + ... + b(nb+1)*x(n-nb)
```

```
%                           - a(2)*y(n-1) - ... - a(na+1)*y(n-na)
%

    % MODIFY THE POLYNOMIALS TO FIND THE ROOTS
    b1 = zeros(max(length(a),length(b)),1); % Need to add zeros to get
 the right roots
    a1 = zeros(max(length(a),length(b)),1); % Need to add zeros to get
 the right roots
    b1(1:length(b)) = b;    % New a with all values
    a1(1:length(a)) = a;    % New a with all values

    % FIND THE ROOTS OF EACH POLYNOMIAL AND PLOT THE LOCATIONS OF THE
 ROOTS
    h1 = plot(real(roots(a1)), imag(roots(a1)));
    hold on;
    h2 = plot(real(roots(b1)), imag(roots(b1)));
    hold off;

    % DRAW THE UNIT CIRCLE
    circle(0,0,1)

    % MAKE THE POLES AND ZEROS X's AND O's

 set(h1, 'LineStyle', 'none', 'Marker', 'x', 'MarkerFaceColor','none', 'linewidth',
 1.5, 'markersize', 8);

 set(h2, 'LineStyle', 'none', 'Marker', 'o', 'MarkerFaceColor','none', 'linewidth',
 1.5, 'markersize', 8);
    axis equal;

    % DRAW VERTICAL AND HORIZONTAL LINES
    xminmax = xlim();
    yminmax = ylim();
    line([xminmax(1) xminmax(2)],[0 0], 'linestyle', ':', 'linewidth',
 0.5, 'color', [1 1 1]*.1)
    line([0 0],[yminmax(1) yminmax(2)], 'linestyle', ':', 'linewidth',
 0.5, 'color', [1 1 1]*.1)

    % ADD LABELS AND TITLE
    xlabel('Real Part')
    ylabel('Imaginary Part')
    title('Pole-Zero Plot')

end


function circle(x,y,r)
% CIRCLE(X,Y,R)  draws a circle with horizontal center X, vertical
 center
% Y, and radius R.
%

    % ANGLES TO DRAW
    ang=0:0.01:2*pi;
```

```
    % DEFINE LOCATIONS OF CIRCLE
    xp=r*cos(ang);
    yp=r*sin(ang);

    % PLOT CIRCLE
    hold on;
    plot(x+xp,y+yp, ':', 'linewidth', 0.5, 'color', [1 1 1]*.1);
    hold off;

end
```

Published with MATLAB® R2020a