Table of Contents

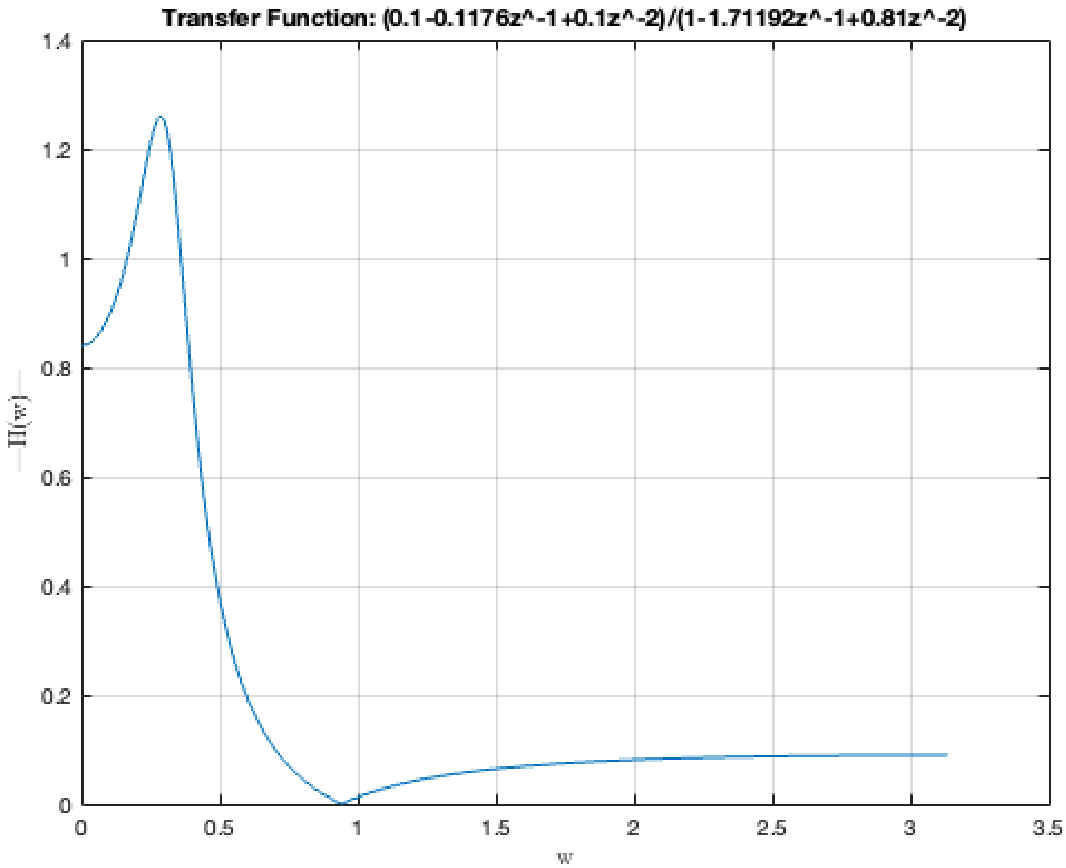

```
clc; close all; clear;

% Setting up required functions
u = @(n)double(n>=0);
del = @(n)double(n==0);
```

Question 1

```
% Transfer Function: (0.1-0.1176z^-1+0.1z^-2)/(1-1.71192z^-1+0.81z^-2)
```

Question 2

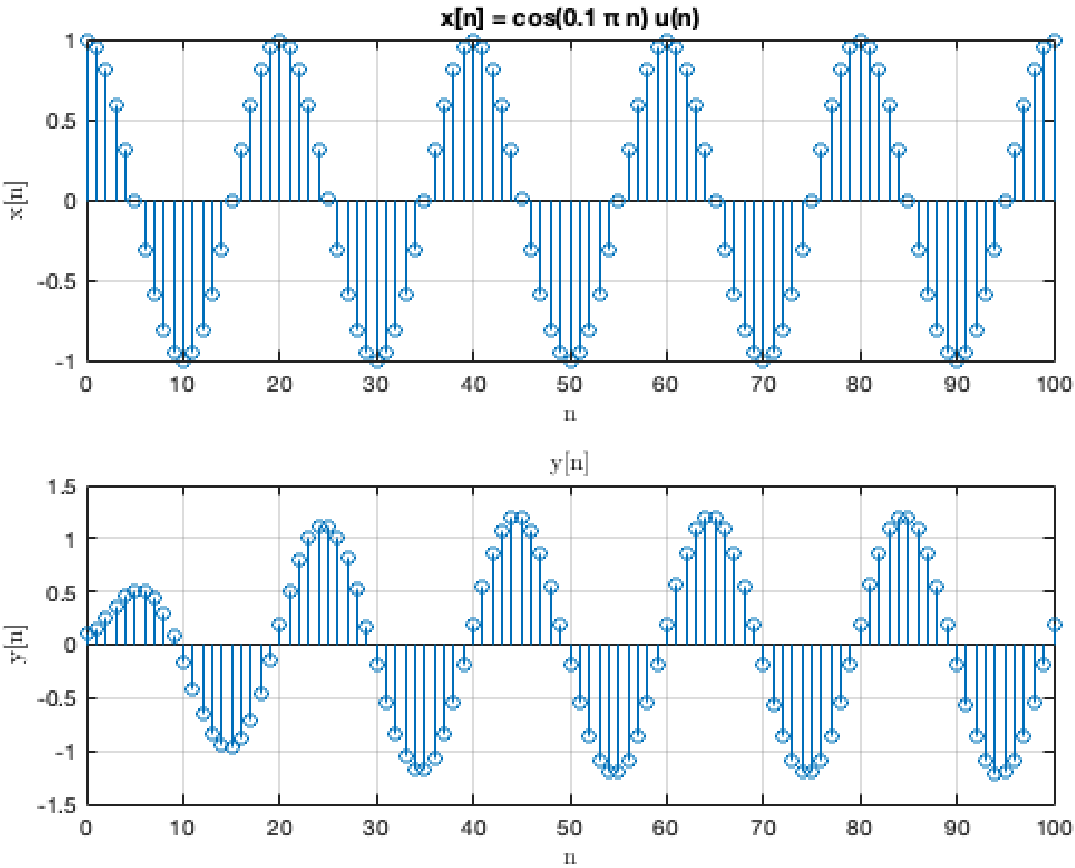
```
figure;
b = [0.1, -0.1176, 0.1];
a = [1, -1.71192, 0.81];
[H,w] = freqz(b,a);
plot(w,abs(H));
title('Transfer Function: (0.1-0.1176z^-1+0.1z^-2)/(1-1.71192z^-1+0.81z^-2)')
grid on;
xlabel('w');
ylabel('|H(w)|')
```



Question 3

```
figure;
n = 0:100;
x = cos(0.1*pi*n).*u(n);

subplot(2,1,1);
stem(n,x);
title('x[n] = cos(0.1 # n) u(n)')
grid on;
xlabel('n');
ylabel('x[n]');

y = filter(b,a,x);

subplot(2,1,2);
stem(n,y);
title('y[n]');
grid on;
xlabel('n');
ylabel('y[n]');
```

```
% Comment on your observations. How could y(n) be predicted from the frequency
 response of the system?
% Having the frequency response H(w), y(n) can be written as
% |H(0.1pi)|*cos(0.1pin+phase(H(0.1pi))). So it can be thought of as a
% being multiplied with a magnitude and phase shifted.
```



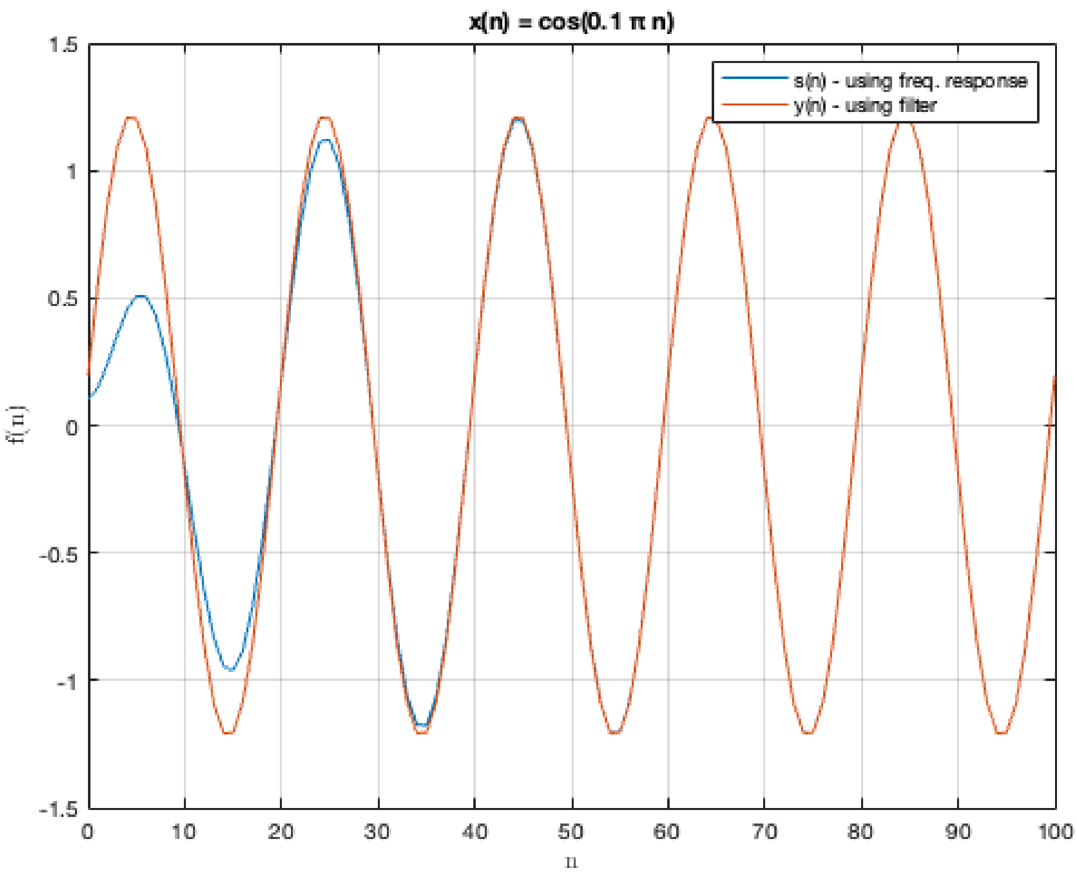
Quesiton 4

```
z = exp(1i*0.1*pi);
b_z = polyval(b,z);
a_z = polyval(a,z);
H = b_z/a_z;
```

Quesiton 5

```
s = abs(H) .* cos(0.1*pi*n + angle(H));
figure;
plot(n,y);
hold on;
plot(n,s);
legend('s(n) - using freq. response','y(n) - using filter');
grid on;
```

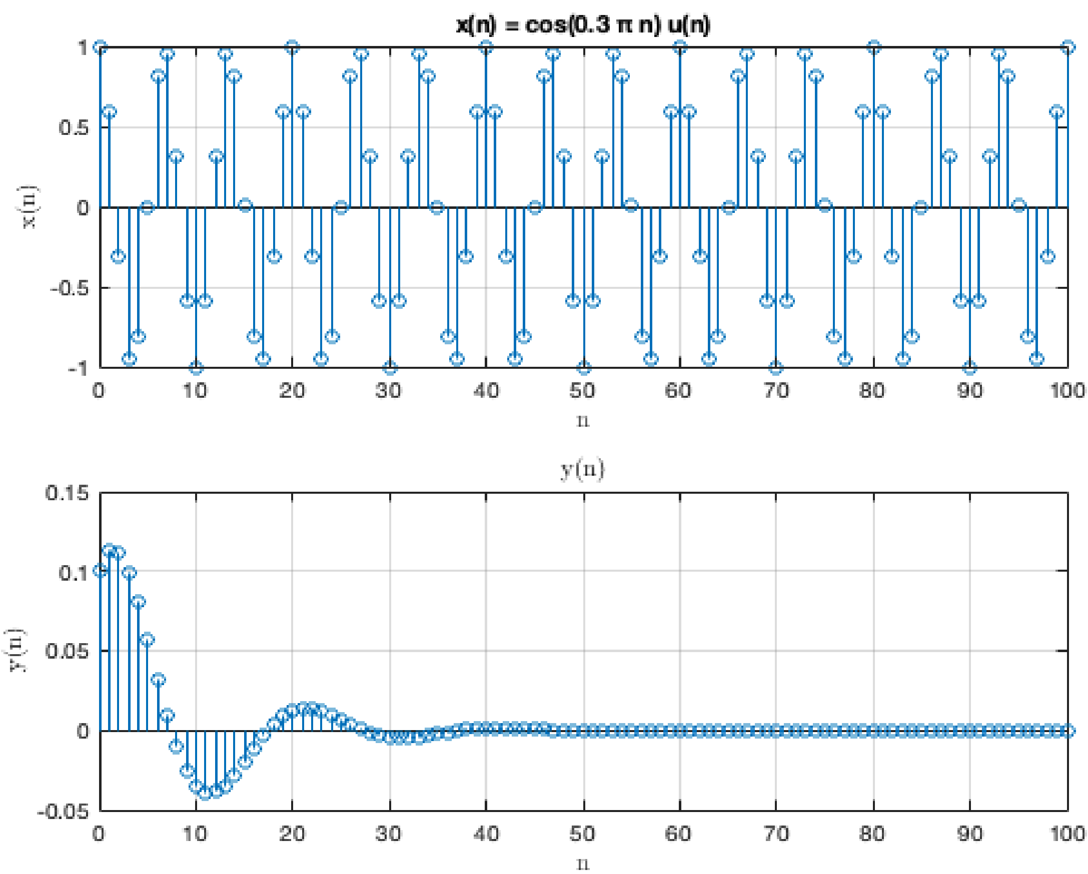
```
title('x(n) = cos(0.1 # n)');
grid on;
xlabel('n');
ylabel('f(n)');

% (After the transient response has died out.) Is this what you find?
% Yes, the are in agreement after the trasnient died out.
```



Quesiton 6

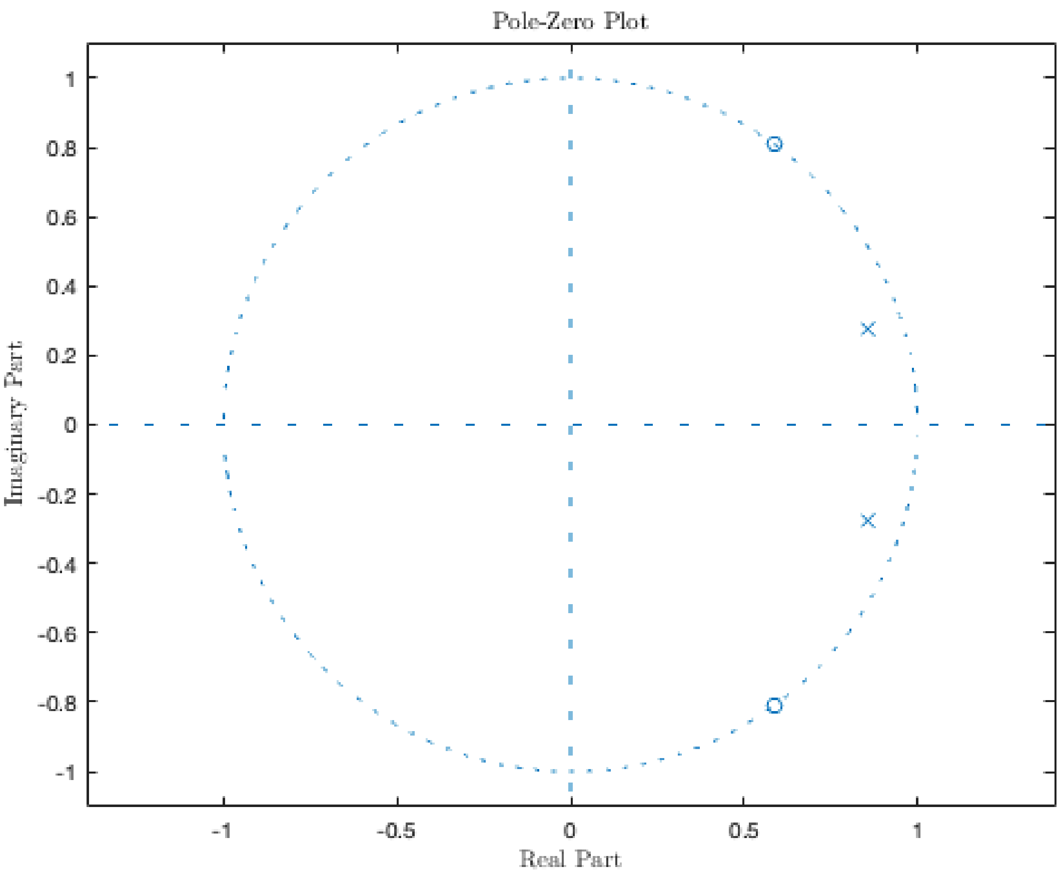
```
figure;
n = 0:100;
x = cos(0.3*pi*n).*u(n);

subplot(2,1,1);
stem(n,x);
title('x(n) = cos(0.3 # n) u(n)')
grid on;
xlabel('n');
ylabel('x(n)');

y = filter(b,a,x);
```

```
subplot(2,1,2);
stem(n,y);
title('y(n)');
grid on;
xlabel('n');
ylabel('y(n)');

% Comment on your observations.
% How could y(n) be predicted from the frequency response of the system?
% Having the frequency response H(w), y(n) can be written as
% |H(0.3pi)|*cos(0.1pin+phase(H(0.3pi))). So it can be thought of as being
% hte product of a magnitude and have a phase shift.

z = exp(1i*0.3*pi);
b_z = polyval(b,z);
a_z = polyval(a,z);
H = b_z/a_z;


% What is the steady-state signal? Relate your explanation to the concept
 describe in the previous question.
% It is a dampened cosine wave as it got multiplied by the magnitude of H that
 decreases to zero
```



Quesiton 7

```
figure;
zplane(b,a);

% Comment on how the frequency response magnitude |Hf (#)| can be predicted
 from the pole-zero diagram.

% When the complex exponential e^(j#) is close to a zero, the gap between
 e^(j#)
% % and the zero is minimal. Since this proximity is in the numerator of the
 transfer function, H(e^(j#)),
% it causes H(e^(j#)) to have a small magnitude. However, and if e^(j#)
 approaches a pole, the distance
% between e^(j#) and the pole also decreases. However, because this closeness
 is reflected in the denominator
% of the transfer function, it results in a significant increase in magnitude.
```



Question 8

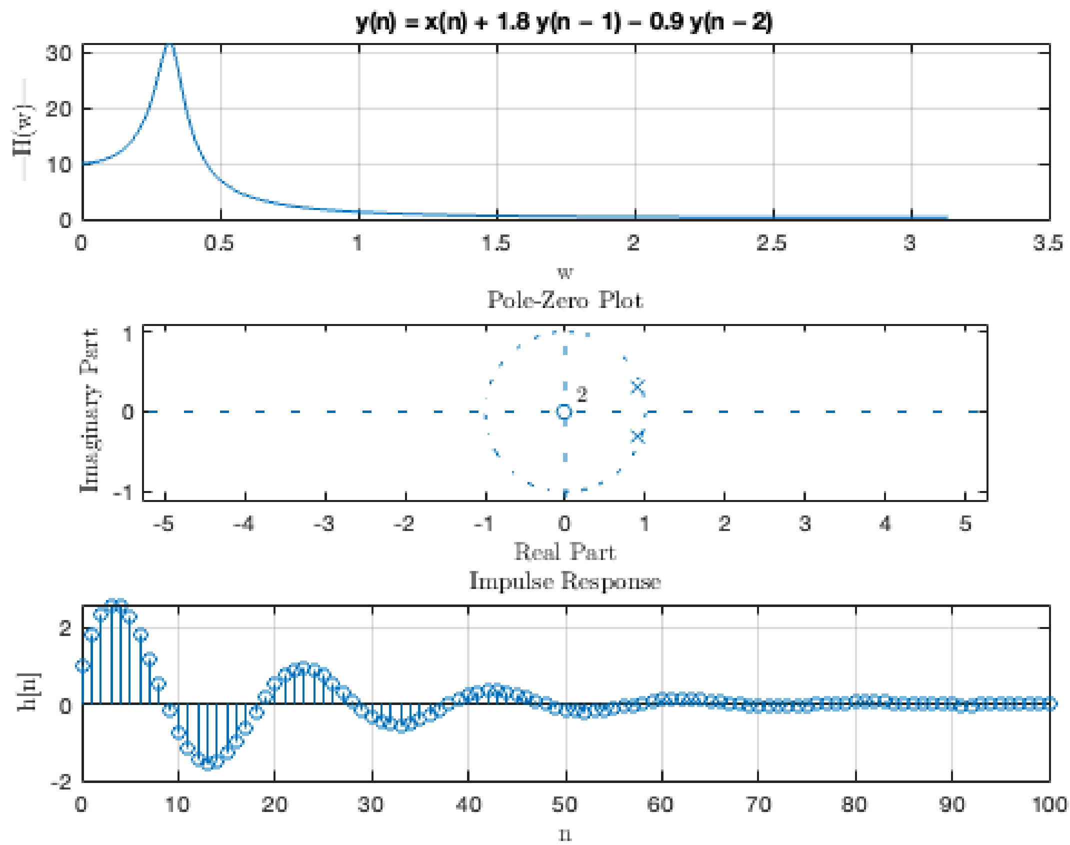
```
% (4) y(n) = x(n) + 1.8 y(n # 1) # 0.9 y(n # 2)
```

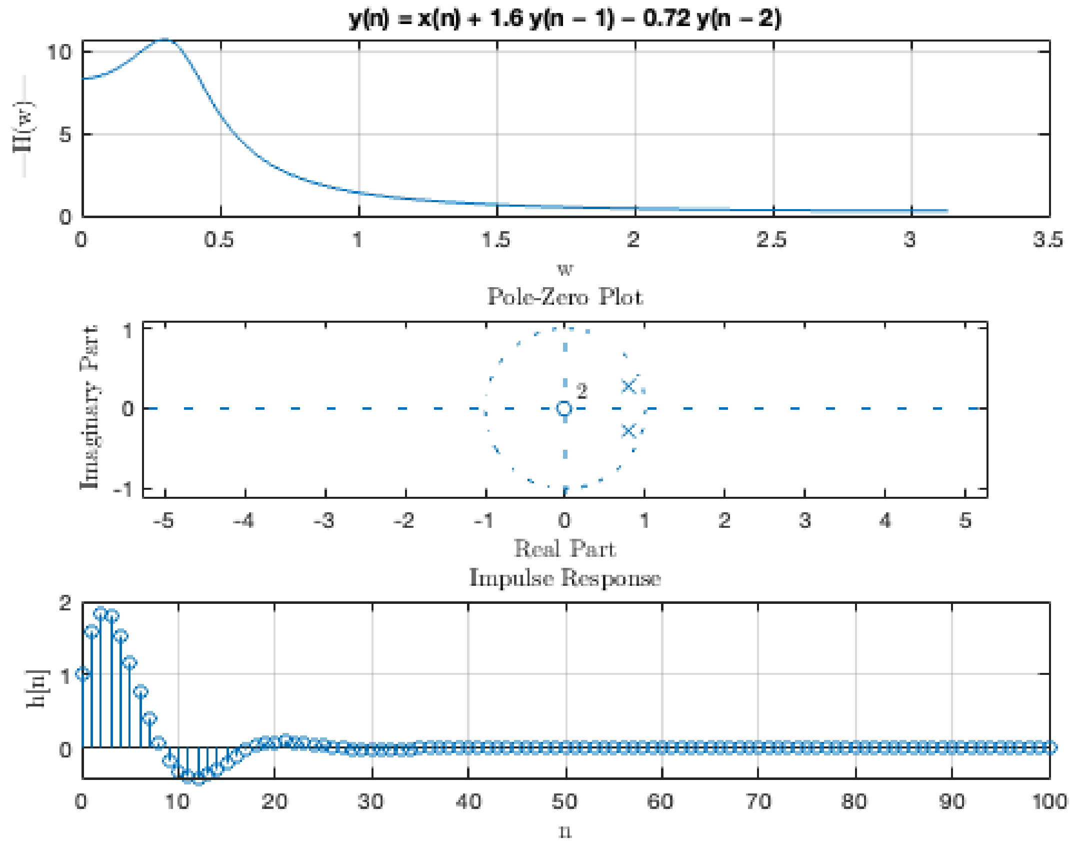
```
figure;
subplot(3,1,1);
b = [1];
a = [1 -1.8 0.9];
[H,w] = freqz(b,a);
plot(w,abs(H));
title('y(n) = x(n) + 1.8 y(n # 1) # 0.9 y(n # 2)')
grid on;
xlabel('w');
ylabel('|H(w)|')
subplot(3,1,2)
zplane(b,a);
subplot(3,1,3);
impulse = del(n);
impulse_response = filter(b,a,impulse);
stem(n,impulse_response);
title("Impulse Response");
xlabel("n")
ylabel("h[n]")
grid on;


% (5) y(n) = x(n) + 1.6 y(n # 1) # 0.72 y(n # 2)

figure;
subplot(3,1,1);
b = [1];
a = [1 -1.6 0.72];
[H,w] = freqz(b,a);
plot(w,abs(H));
title('y(n) = x(n) + 1.6 y(n # 1) # 0.72 y(n # 2)')
grid on;
xlabel('w');
ylabel('|H(w)|')
subplot(3,1,2)
zplane(b,a);
subplot(3,1,3);
impulse = del(n);
impulse_response = filter(b,a,impulse);
stem(n,impulse_response);
title("Impulse Response");
xlabel("n")
ylabel("h[n]")
grid on;


% (6) y(n) = x(n) + 1.53 y(n # 1) # 0.9 y(n # 2)

figure;
subplot(3,1,1);
b = [1];
a = [1 -1.53 0.90];
[H,w] = freqz(b,a);
plot(w,abs(H));
```

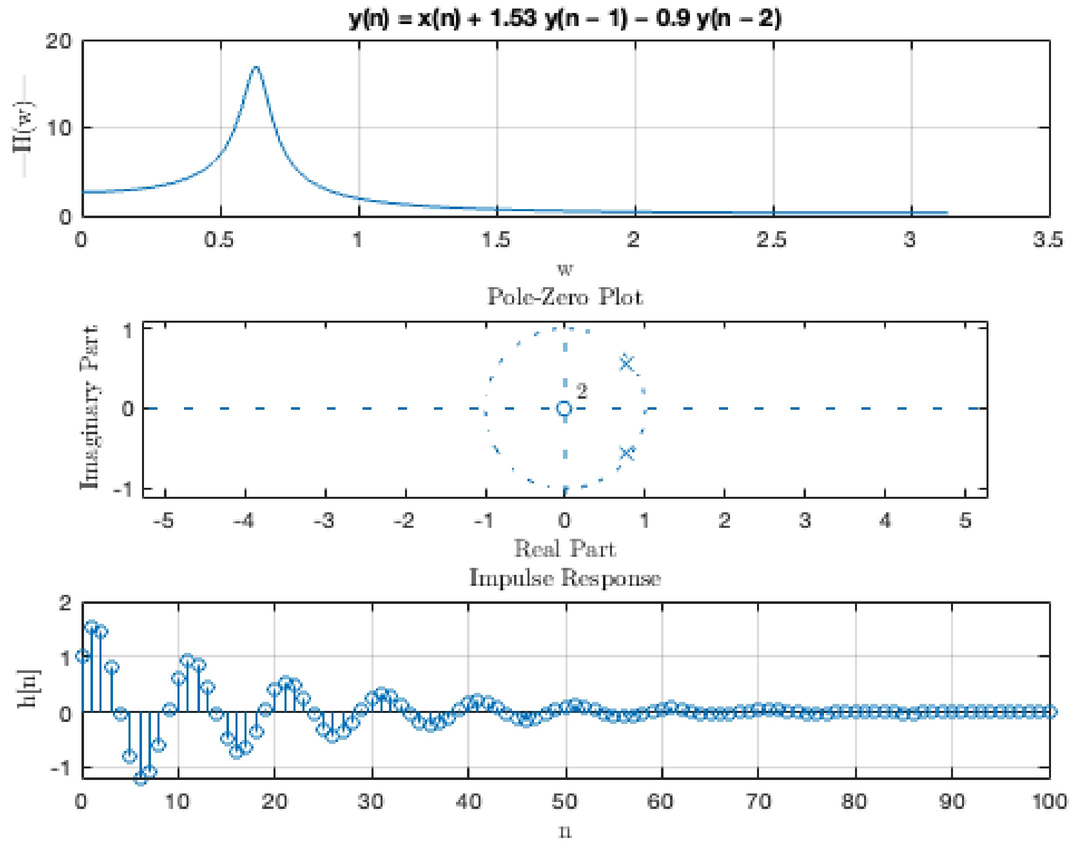
```
title('y(n) = x(n) + 1.53 y(n # 1) # 0.9 y(n # 2)')
grid on;
xlabel('w');
ylabel('|H(w)|')
subplot(3,1,2)
zplane(b,a);
subplot(3,1,3);
impulse = del(n);
impulse_response = filter(b,a,impulse);
stem(n,impulse_response);
title("Impulse Response");
xlabel("n")
ylabel("h[n]")
grid on;

% (7) y(n) = x(n) + 1.4 y(n # 1) + 0.72 y(n # 2)

figure;
subplot(3,1,1);
b = [1];
a = [1 -1.4 -0.72];
[H,w] = freqz(b,a);
plot(w,abs(H));
title('y(n) = x(n) + 1.4 y(n # 1) + 0.72 y(n # 2)')
grid on;
xlabel('w');
ylabel('|H(w)|')
subplot(3,1,2)
zplane(b,a);
subplot(3,1,3);
impulse = del(n);
impulse_response = filter(b,a,impulse);
stem(n,impulse_response);
title("Impulse Response");
xlabel("n")
ylabel("h[n]")
grid on;


% (8) y(n) = x(n) # 0.85 y(n # 1)

figure;
subplot(3,1,1);
b = [1];
a = [1 0.85];
[H,w] = freqz(b,a);
plot(w,abs(H));
title('y(n) = x(n) # 0.85 y(n # 1)')
grid on;
xlabel('w');
ylabel('|H(w)|')
subplot(3,1,2)
zplane(b,a);
subplot(3,1,3);
```

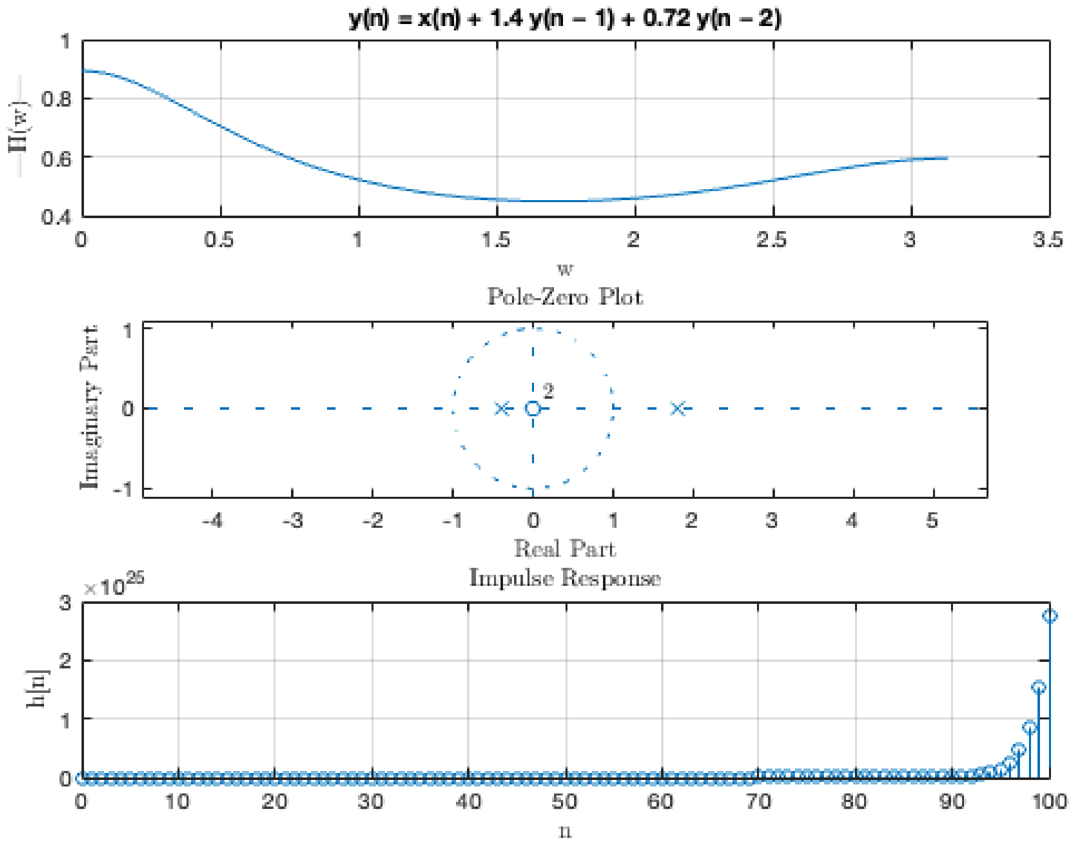
```
impulse = del(n);
impulse_response = filter(b,a,impulse);
stem(n,impulse_response);
title("Impulse Response");
xlabel("n")
ylabel("h[n]")
grid on;



% (9) y(n) = x(n) # 0.95 y(n # 1) (

figure;
subplot(3,1,1);
b = [1];
a = [1 0.95];
[H,w] = freqz(b,a);
plot(w,abs(H));
title('y(n) = x(n) # 0.95 y(n # 1)')
grid on;
xlabel('w');
ylabel('|H(w)|')
subplot(3,1,2)
zplane(b,a);
subplot(3,1,3);
impulse = del(n);
impulse_response = filter(b,a,impulse);
stem(n,impulse_response);
title("Impulse Response");
xlabel("n")
ylabel("h[n]")
grid on;

% Comment on your observations:
% How can you predict from the pole-zero diagram what the frequency response
 will look like and what the impulse response will look like?

% When the complex exponential e^(j#) is close to a zero, the gap between
 e^(j#)
% % and the zero is minimal. Since this proximity is in the numerator of the
 transfer function, H(e^(j#)),
% it causes H(e^(j#)) to have a small magnitude. However, and if e^(j#)
 approaches a pole, the distance
% between e^(j#) and the pole also decreases. However, because this closeness
 is reflected in the denominator
% of the transfer function, it results in a significant increase in
% magnitude. Additionally. we can use the rules I gave in the next
% question.

% Suppose you were given a diagram of each of the pole-zero diagrams,
 frequency responses,
% and impulse responses, but that they were out of order. Could you match each
 diagram to
% the others (without doing any computation)?
```

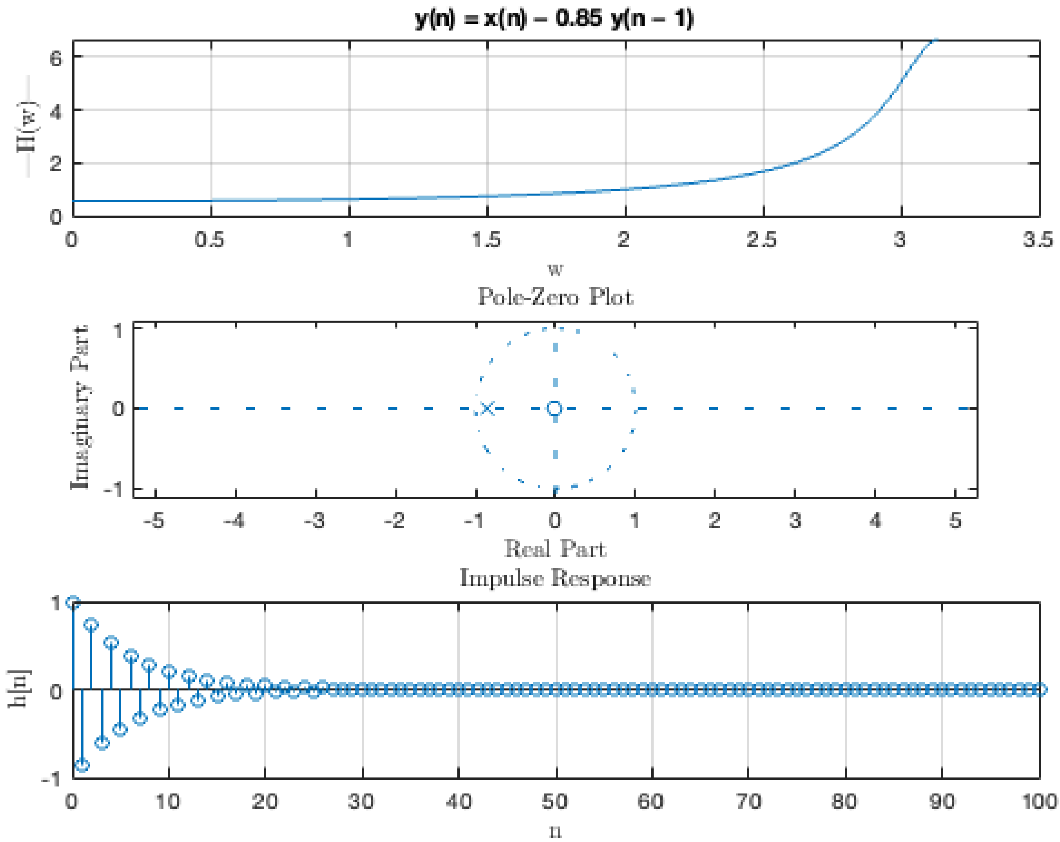
```
% Using using a varitey of properties. For example, and when the zeros are far
 from the unit circle, the frequency response is quite flat.
% Additionally, the poles must be strictly inside the unit circle for the
 system to be causal and stable.
% The closer the poles are to the unit circle, the sharper the peak is.
% Also, if the zeros are on the unit circle, so the frequency response has
% nulls at frequencies. Another rule that can be used is that complex
% conjugated poles cause oscillation.
```

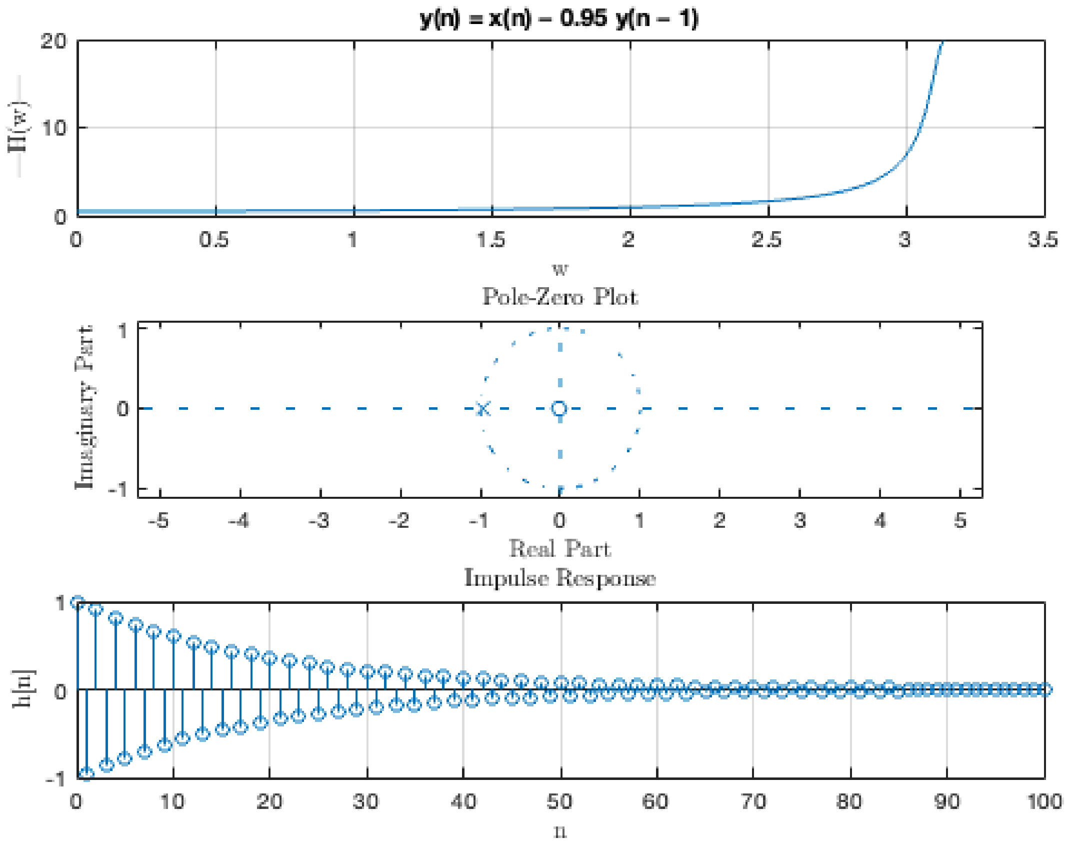



y(n) = x(n) + 1.6 y(n − 1) − 0.72 y(n − 2)
H(w)
w
Pole-Zero Plot
Imaginary Part
Real Part
Impulse Response
h[n]
n


y(n) = x(n) + 1.53 y(n − 1) − 0.9 y(n − 2)
H(w)
w
Pole-Zero Plot
Imaginary Part
Real Part
Impulse Response
h[n]
n


y(n) = x(n) + 1.4 y(n − 1) + 0.72 y(n − 2)
H(w)
w
Pole-Zero Plot
Imaginary Part
Real Part
2
Impulse Response
×10^25
h[n]
n


y(n) = x(n) − 0.85 y(n − 1)
H(w)
w
Pole-Zero Plot
Imaginary Part
Real Part
Impulse Response
h[n]
n



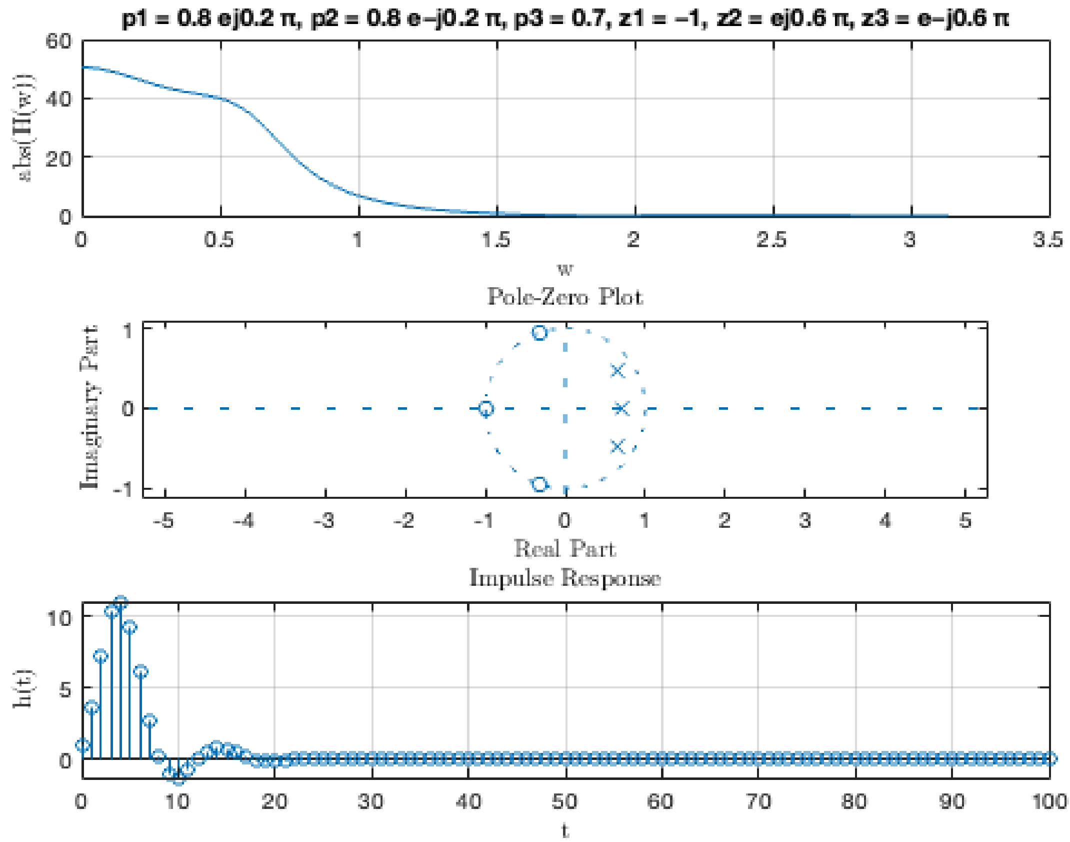
Question 9

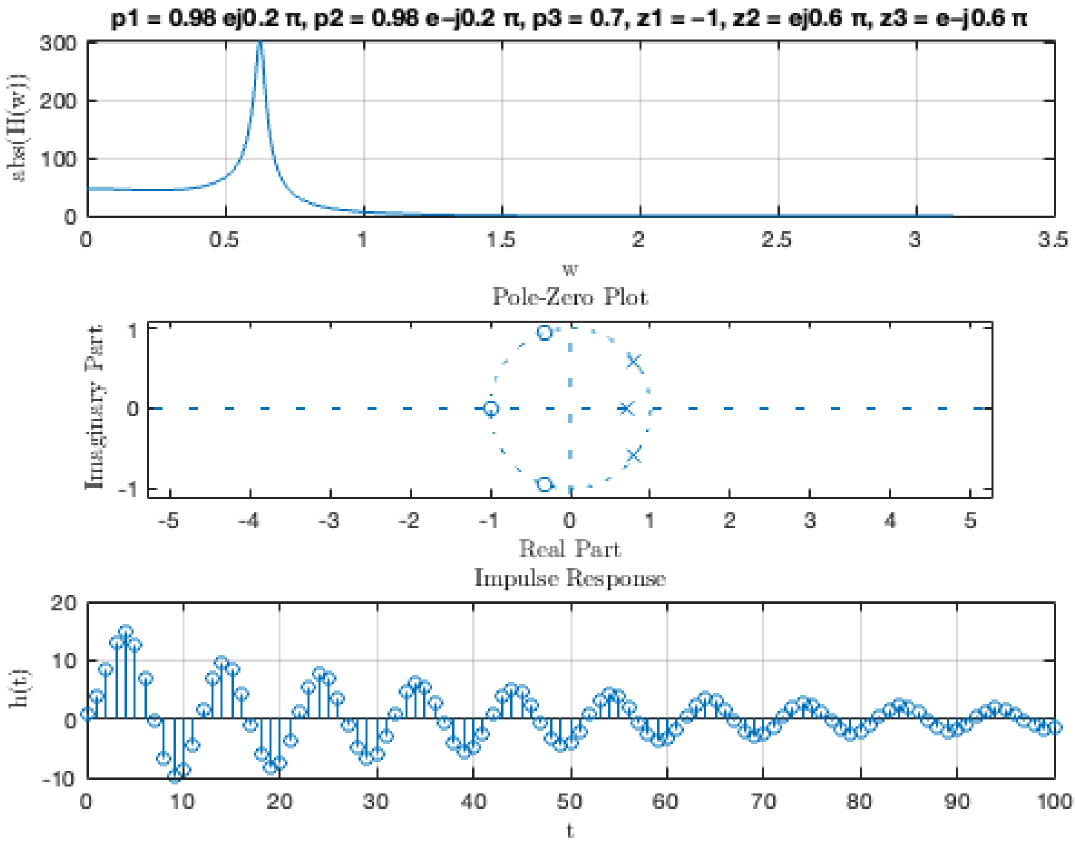
```
% p1 = 0.8 ej0.2 #, p2 = 0.8 e#j0.2 #, p3 = 0.7, z1 = #1, z2 = ej0.6 #, z3 = e
#j0.6 #

p = [0.8*exp(1i*0.2*pi), 0.8*exp(-1i*0.2*pi), 0.7];
z = [-1, exp(1i*0.6*pi), exp(-1i*0.6*pi)];

b = poly(z);
a = poly(p);

disp("b=");
disp(b);
disp("a=");
disp(a);

% Difference Equaion:
% y(n) = x(n) + 1.6180x(n-2) + x(n-3) + 1.994y(n-1) - 1.5461y(n-2) +
% 0.4480y(n-3)

figure;
subplot(3,1,1);
```

```
[H,w] = freqz(b,a);
plot(w,abs(H));
title('p1 = 0.8 ej0.2 #, p2 = 0.8 e#j0.2 #, p3 = 0.7, z1 = #1, z2 = ej0.6 #,
 z3 = e#j0.6 #')
grid on;
xlabel('w');
ylabel('abs(H(w))')
subplot(3,1,2)
zplane(b,a);
subplot(3,1,3);
impulse = del(n);
impulse_response = filter(b,a,impulse);
stem(n,impulse_response);
title("Impulse Response");
xlabel("t")
ylabel("h(t)")
grid on;

% p1 = 0.98 ej0.2 #, p2 = 0.98 e#j0.2 #, p3 = 0.7, z1 = #1, z2 = ej0.6 #, z3 =
 e#j0.6 #

p = [0.98*exp(1i*0.2*pi), 0.98*exp(-1i*0.2*pi), 0.7];
z = [-1, exp(1i*0.6*pi), exp(-1i*0.6*pi)];

b = poly(z);
a = poly(p);

disp("b=");
disp(b);
disp("a=");
disp(a);

% Difference Equaion:
% y(n) = x(n) + 1.6180x(n-2) + x(n-3) + 1.994y(n-1) - 1.5461y(n-2) +
% 0.4480y(n-3)


figure;
subplot(3,1,1);
[H,w] = freqz(b,a);
plot(w,abs(H));
title('p1 = 0.98 ej0.2 #, p2 = 0.98 e#j0.2 #, p3 = 0.7, z1 = #1, z2 = ej0.6 #,
 z3 = e#j0.6 #')
grid on;
xlabel('w');
ylabel('abs(H(w))')
subplot(3,1,2)
zplane(b,a);
subplot(3,1,3);
impulse = del(n);
impulse_response = filter(b,a,impulse);
stem(n,impulse_response);
title("Impulse Response");
xlabel("t")
```

```
ylabel("h(t)")
grid on;

% How does this change to the position of the poles affect the nature of the
 impulse response
% of the system and the frequency response?

% As the position of the poles was shifted, both the impulse and frequency
 responses were modified. The impulse response seem to now peak at around
 0.6 and the impulse response seem to have way less damping. That is because
 as the poles were moved to becomes closer to the boundaries of the unit
 circle, the resonance and sensitivity in the frequency response becomes more
 intentsive. In the impulse response, nearness to the unit circle influences
 decay rates and potential oscillations, leading to alterations in the
 system's behavior.
% The closer the poles are to the unit circle, the sharper the peak is.
```

```
b=
    1.0000    1.6180    1.6180    1.0000

a=
    1.0000   -1.9944    1.5461   -0.4480

b=
    1.0000    1.6180    1.6180    1.0000

a=
    1.0000   -2.2857    2.0704   -0.6723
```


p1 = 0.8 ej0.2 π, p2 = 0.8 e−j0.2 π, p3 = 0.7, z1 = −1, z2 = ej0.6 π, z3 = e−j0.6 π
abs(H(w))
w
Pole-Zero Plot
Imaginary Part
Real Part
Impulse Response
h(t)
t



Published with MATLAB® R2023a