Question:

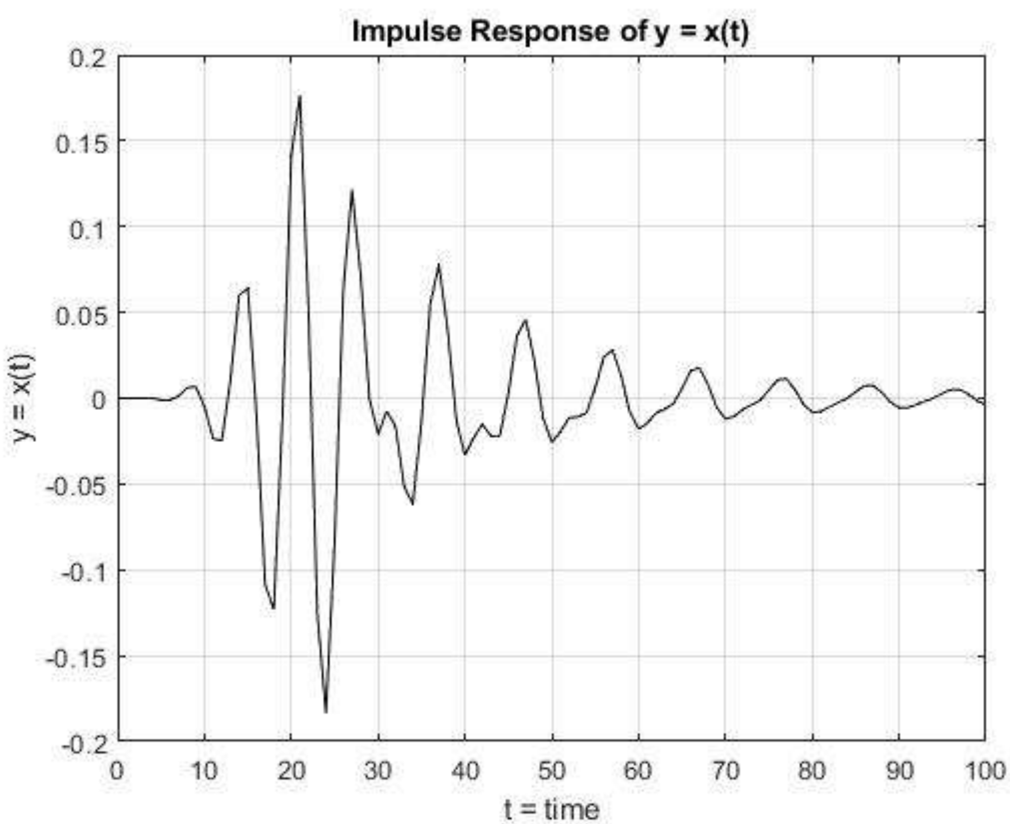
Design a 10th order band pass (BP) *Butterworth filter* with a pass band from 100 to 200 Hz and plot both impulse response and frequency response.

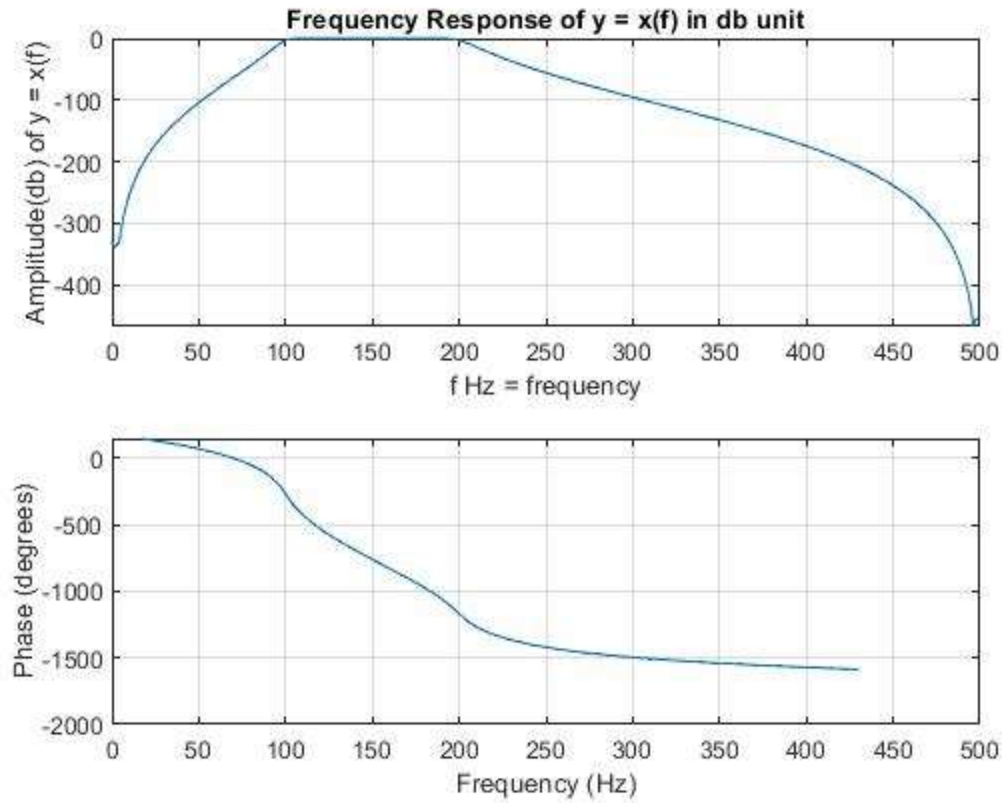
Solution:

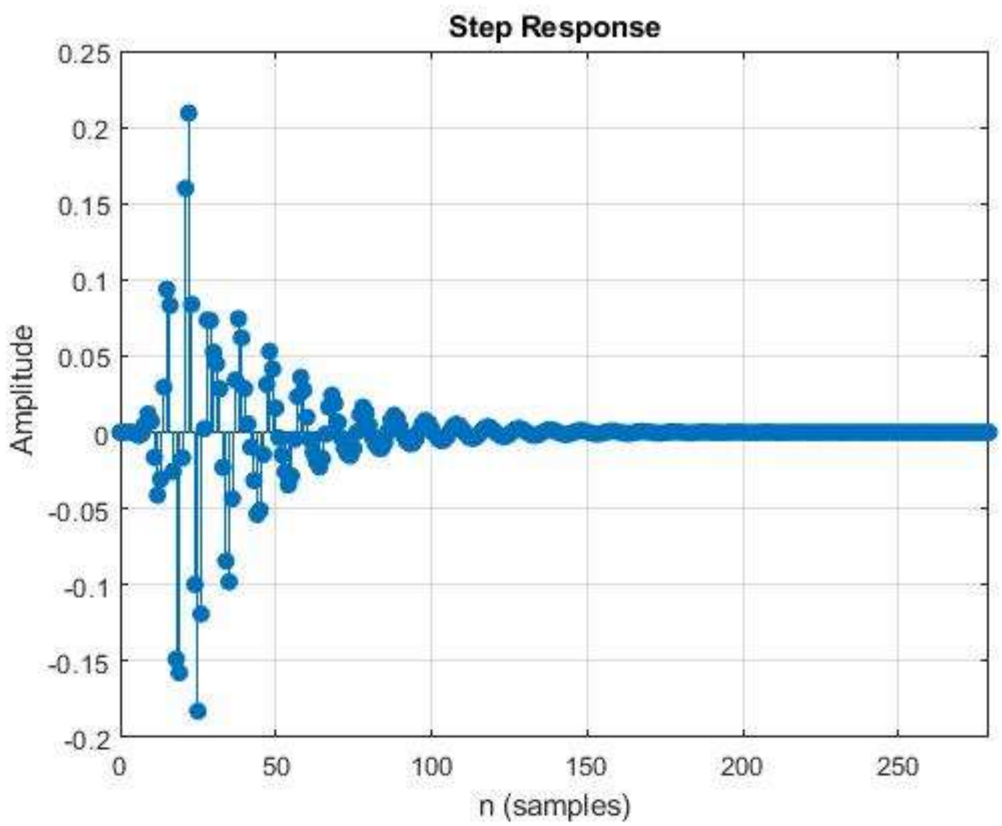
```
%First part- design an impulse response(unit impulse)
design
%Second part- frequency response design

%First Part:
Fs = 1000;  %let sampling frequency ,Fs = 1000 Hz
n = 10;     % Given order = 10

wn = [100 200]/500;      % wn = BP range, that is given
for BP passband frequency from 100 to 200Hz
                         % 500 means , Fs/2 = 1000/2 =
500Hz

[b,a] = butter(n,wn);    % b,a coefficients of z
transform X(z)=b0+b1y1+b2y2/a0+a1x1+a2x2
                         % b = zeroes, a = poles
figure(1);               % for first figure
[y,t] = impz( b, a, 101); % impz = impluse
response(time domain)
                          % where, y = coefficients b,
t = coefficients a
                          % let, 101 = no of sampling
[0-100]

plot(t,y,'k');              % plot in x-axis = t,  y-
axis = y in time domain, k = black

grid on;

title("Impulse Response of y = x(t)");
xlabel("t = time");
ylabel("y = x(t)");

% second part
figure(2);
```

```
freqz( b, a, 501, Fs);        % for frequency response in
f-domain
                              % 501 = no of frequency
samples
                              % Fs = sampling frequency
used for
                              % normalization

grid on;
title("Frequency Response of y = x(f) in db unit");
xlabel("f Hz = frequency");
ylabel("Amplitude(db) of y = x(f)");

% for finite visualization we need to convert infinite
impulse response
% to step response for snapshot for better
understanding
figure(3);
stepz(b,a); % stepz() for step response
grid on ;   % always we have to write grid on statement
after write stepz()
```

Output:




Frequency Response of y = x(f) in db unit
Amplitude(db) of y = x(f)
f Hz = frequency
Phase (degrees)
Frequency (Hz)


Step Response
Amplitude
n (samples)

Observation:

So far, Butterworth filters have a more linear phase response in the pass-band than other types of filters such as Chebyshev or Elliptic Filters. Because it does not have any ripples in the band-pass area as we can see on frequency response graph. As a result it provides a flat or smooth cutoff frequency. But it has a slow roll-off bandwidth transition that is not much expected for optimum usages in minimum cost. In brief, the Butterworth filter is able to provide better flat response at the order of 10 but slower roll-off transition but its impulse frequency is much stable compared to the Chebyshev filter.

Question:
Design a 10th order BP Chebyshev filter of type-I and type-II with a passband from 100 to 200 Hz and plot the impulse response and frequency response. Consider 25dB ripple in passband.

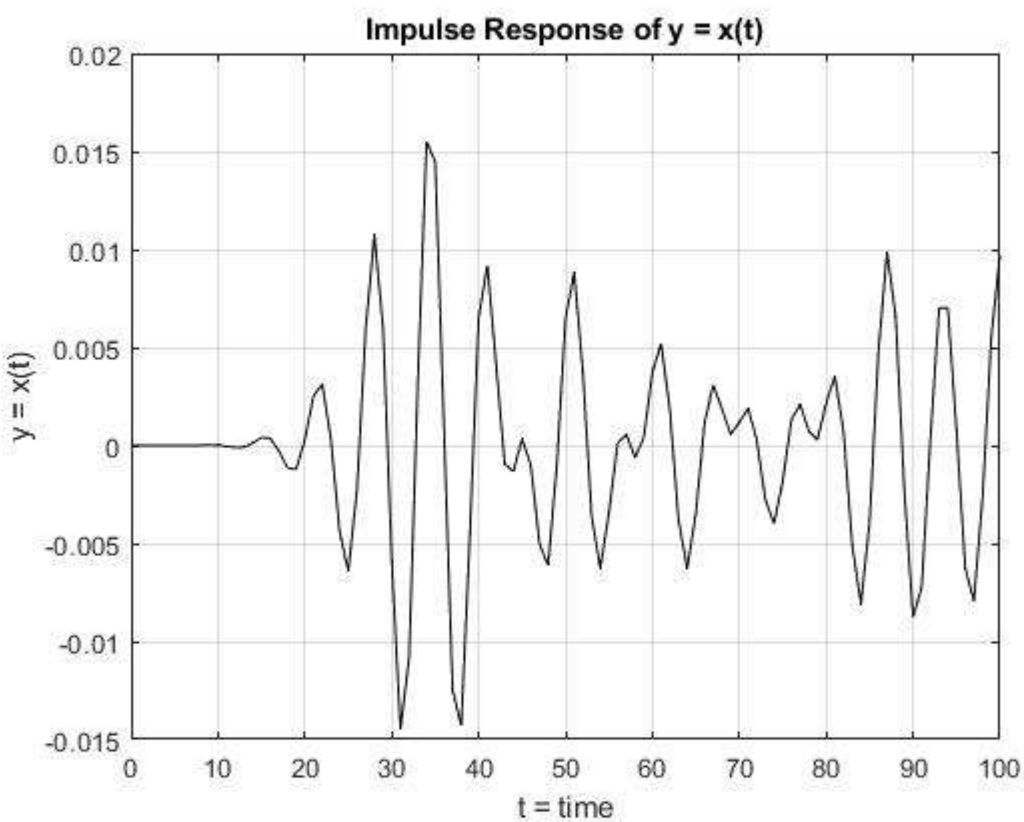
Solution:

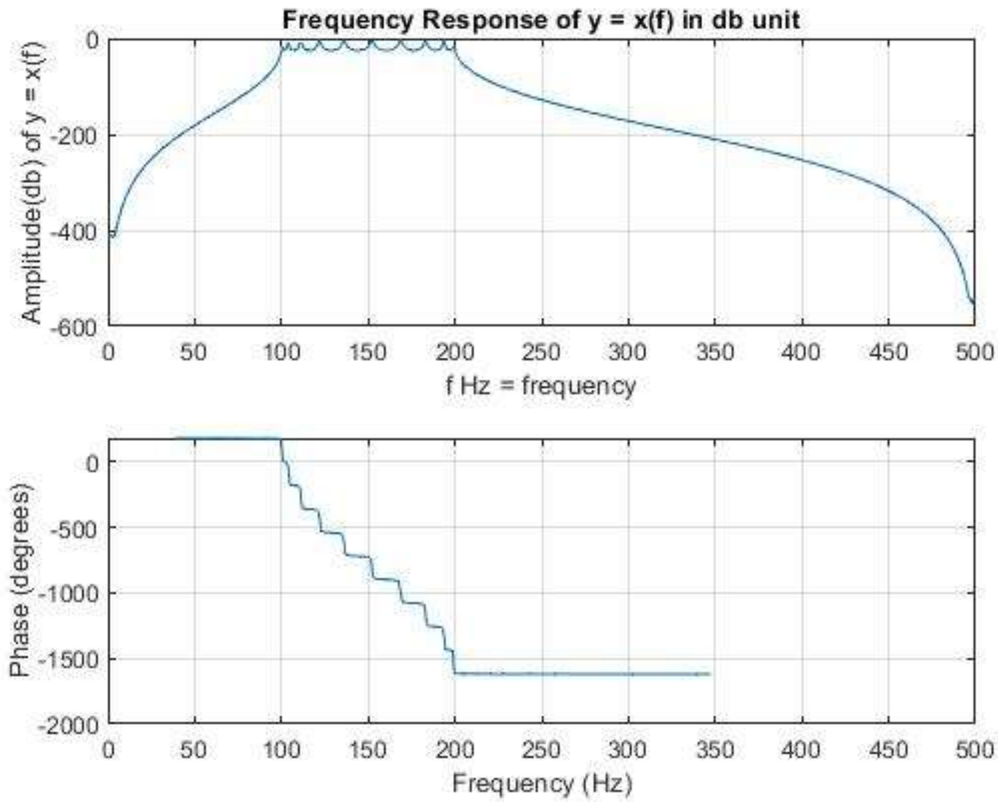
```
% First Part: Chebyshev Type-I

Fs=1000;        % Let sampling frequency ,Fs = 1000 Hz
n=10;           % Given order = 10
wn = [100 200]/500;     % wn = BP range, that is given
for BP passband frequency from 100 to 200Hz
                        % 500 means , Fs/2 = 1000/2 =
500Hz
Rp=25;                  % Given 25dB ripple in passband

[b,a]=cheby1(n,Rp,wn);  % cheby1() = For chebyshev
filter type-I (by default for passband)
                        % b,a coefficients of z
transform X(z)=b0+b1y1+b2y2/a0+a1x1+a2x2
                        % b = zeroes, a = poles
[y,t]=impz(b,a,101);    % impz = impluse response(in
time domain)
                        % where, y = coefficients b, t
= coefficients a
```

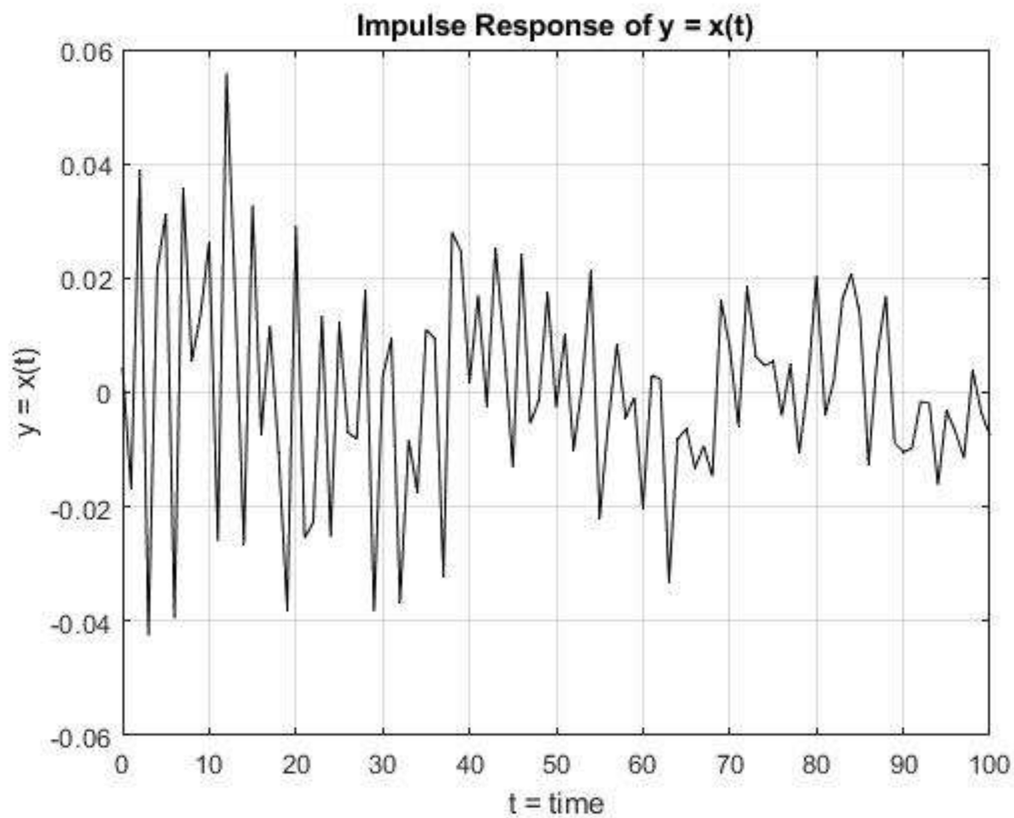
```
                             % let, 101 = no of sampling [0-
100]
figure(1);                   % for first figure
plot(t,y,'k');               % plot in x-axis = t,  y-axis =
y in time domain, k = black
grid on;
title("Impulse Response of y = x(t)");
xlabel("t = time");
ylabel("y = x(t)");

figure(2);                   % for figure-2
freqz(b,a,501,Fs);           % for frequency response in f-
domain
                             % 501 = no of frequency samples
                             % Fs = sampling frequency used
for
                             % normalization
title("Frequency Response of y = x(f) in db unit");
xlabel("f Hz = frequency");
ylabel("Amplitude(db) of y = x(f)");

% Second Part: Chebyshev Type-II
Fs=1000;            % Let sampling frequency ,Fs = 1000 Hz
n=10;               % Given order = 10
wn = [100 200]/500;          % wn = BP range, that is given
for BP passband frequency from 100 to 200Hz
                             % 500 means , Fs/2 = 1000/2 =
500Hz
Rp=25;                       % Given 25dB ripple in passband
[b,a]=cheby1(n,Rp,wn,"Stop");% cheby1() = For chebyshev
filter type-I (for stopband)
                                   % b,a coefficients of z
transform X(z)=b0+b1y1+b2y2/a0+a1x1+a2x2
                                   % b = zeroes, a = poles
[y,t]=impz(b,a,101);         % impz = impluse response(in
time domain)
                             % where, y = coefficients b, t
= coefficients a
                             % let, 101 = no of sampling [0-
100]
figure(3);                   % for 3rd figure
```

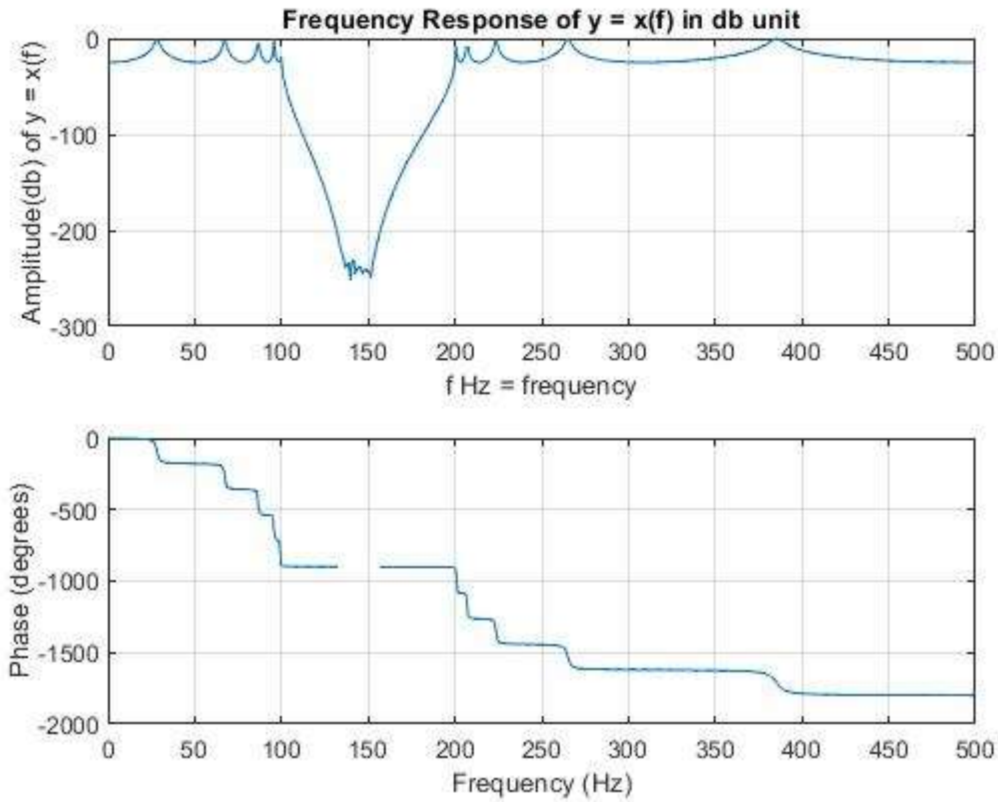
```
plot(t,y,'k');           % plot in x-axis = t,  y-axis =
y in time domain, k = black
grid on;
title("Impulse Response of y = x(t)");
xlabel("t = time");
ylabel("y = x(t)");

figure(4);               % for 4th figure
freqz(b,a,501,Fs);% for frequency response in f-domain
                  % 501 = no of frequency samples
                  % Fs = sampling frequency used for
                  % normalization
title("Frequency Response of y = x(f) in db unit");
xlabel("f Hz = frequency");
ylabel("Amplitude(db) of y = x(f)");
```

Output:




Frequency Response of y = x(f) in db unit
Amplitude(db) of y = x(f)
f Hz = frequency
Phase (degrees)
Frequency (Hz)


Impulse Response of y = x(t)
y = x(t)
t = time



Observation:

In brief, Chebyshev Type-I filter has pass-band ripples and maximally flat in stop-band area. In Chebyshev Type-II filter has stop-band ripples and maximally flat in pass-band area. It has faster roll-off and better gain than Butterworth filter. Impulse response of Chebyshev filter Type-I is good compared to the Type-II. Chebyshev filter Type-I achieves a faster roll-off by allowing ripple in pass-band. When the ripple is set 0% then it is called a maximally flat or Butterworth filter. Consider using a ripple of 0.5% in Chebyshev filter design then pass-band unflattens is so small that it cannot be seen in this graph but the roll-off is much faster than Butterworth. Chebyshev filter Type-I with 0.5% ripples is mostly used than Type-II because of achieving average flatness and faster roll-off. Greater than using 0.5% ripples in Type-I, Chebyshev provide best roll-off that is same as the Ideal impulse response but in pass-band it shows unflattens that is unexpected because it provides noisy signal. Chebyshev filter is better than Butterworth if we consider roll-off transition, because Butterworth has slow roll-of transition but Chebyshev filter has faster roll-off transition that we want. It may good compromise between Elliptic and Butterworth filter. Chebyshev filter is able to provide better bandwidth response and good performance at the order of 10.

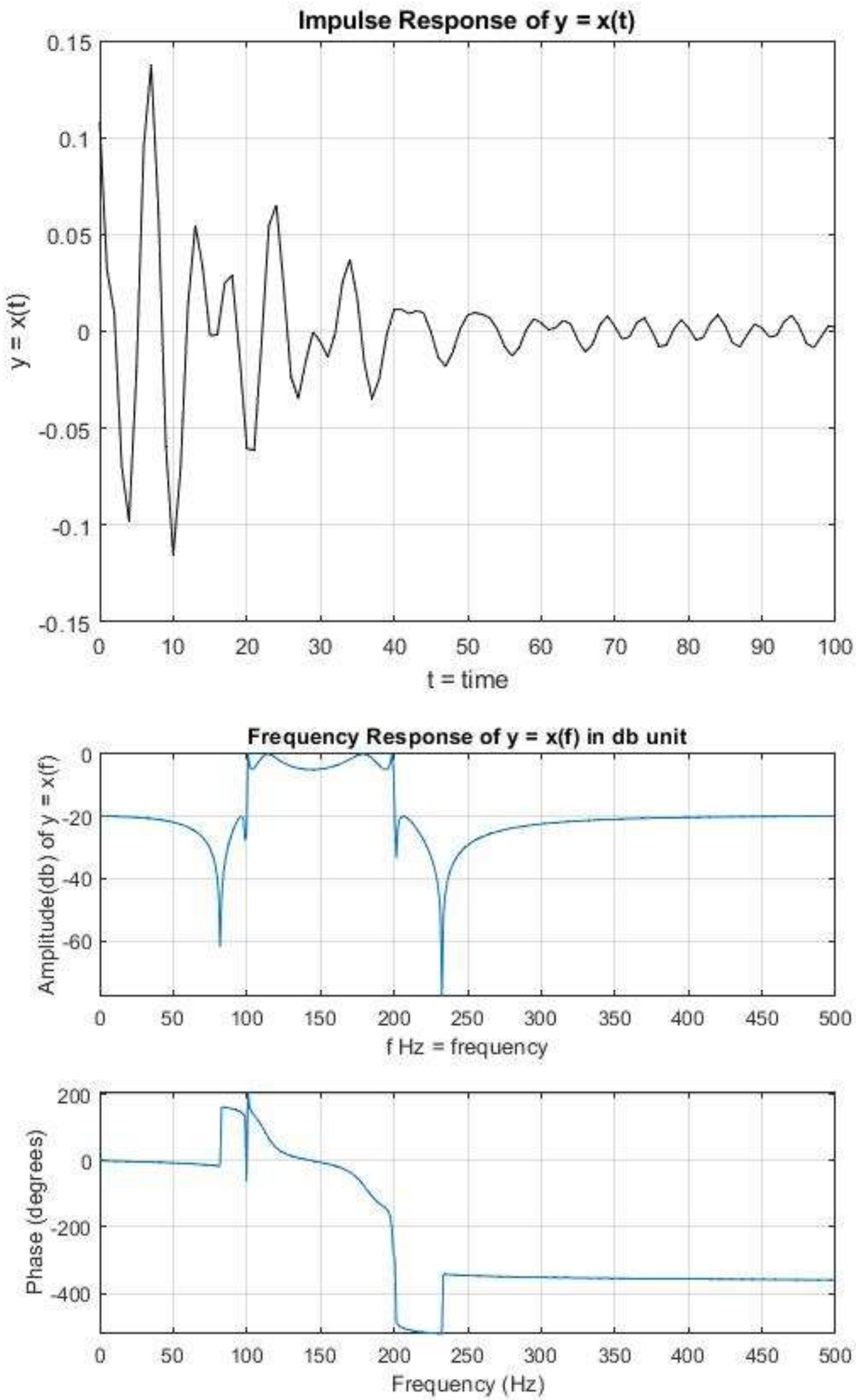
Question:
Design a 10th order BP elliptic filter with passband in the range of 100 to 200 Hz. Plot the frequency response and impulse response of the filter. Consider 5dB ripple in passband and 20dB attenuation in stopband.

Solution:

```
Fs=1000;          % Let sampling frequency ,Fs = 1000 Hz

n=10;             % Given order = 10
wn = [100 200]/500;     % wn = BP range, that is given
for BP passband frequency from 100 to 200Hz
% 500 means , Fs/2 = 1000/2 = 500Hz
Rp=5;                   % Given 5dB ripple for passband

Rs=20;                  % Given 20dB ripple for
stopband

[b,a]=ellip(n,Rp,Rs,wn);% ellip() = For Elliptic filter
(in time domain)
                        % b,a coefficients of z
transform X(z)=b0+b1y1+b2y2/a0+a1x1+a2x2
                        % b = zeroes, a = poles
[y,t]=impz(b,a,101);    % impz = impluse response(in
time domain)

figure(1);              % for first figure
plot(t,y,'k');          % plot in x-axis = t,  y-axis =
y in time domain, k = black

grid on;
title("Impulse Response of y = x(t)");

xlabel("t = time");
```

```
ylabel("y = x(t)");
figure(2);                    % for 2nd figure

freqz(b,a,501,Fs);            % for frequency response in f-domain
                              % 501 = no of frequency samples
                              % Fs = sampling frequency used
                              % normalization
title("Frequency Response of y = x(f) in db unit");

xlabel("f Hz = frequency");
ylabel("Amplitude(db) of y = x(f)");
```

Output:


Impulse Response of y = x(t)
y = x(t)
t = time
Frequency Response of y = x(f) in db unit
Amplitude(db) of y = x(f)
f Hz = frequency
Phase (degrees)
Frequency (Hz)

Observation:

The observations of the elliptic filter are best description in terms of the following parameters that specify the frequency response graph. Elliptical Filter has ripples in both band-pass and stop-band area. Elliptical Filter provides good impulse response compared to the Butterworth and Chebyshev Filter because its impulse response is very stable due to increasing time. Elliptical Filter has faster roll-off so it has better gain compared to the Butterworth and Chebyshev filter because its transition is so close to the ideal impulse response. That's why this filter is mostly used in the present world although it has ripples in both pass-band and stop-band areas. For better gain Elliptical is mostly used in Real-time application. And the elliptic filter is able to provide better bandwidth response and good performance at the order of 10.

Question:

A bandpass IIR filer has the transfer function of, $H(z) = \dfrac{z^2 - 1}{z^2 + 0.877}$; determine poles and zeros of the filter and frequency response taking sampling frequency $Fs = 500\text{Hz}$.

Solution:

The transfer function of the filter, $H(z) = \dfrac{z^2 - 1}{z^2 + 0.877} = \dfrac{1 - z^{-2}}{1 + 0.877z^{-2}}$.

Poles: 0 + 0.9365i, 0 - 0.9365i

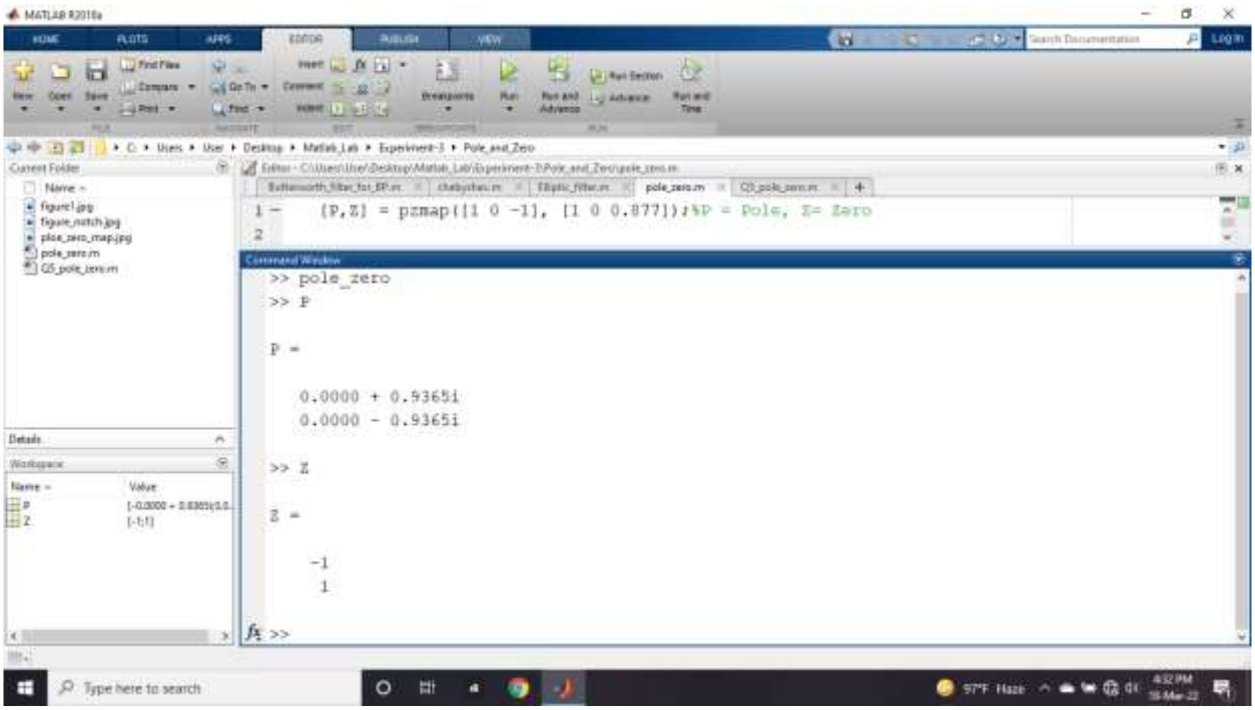
Zeroes: -1, 1

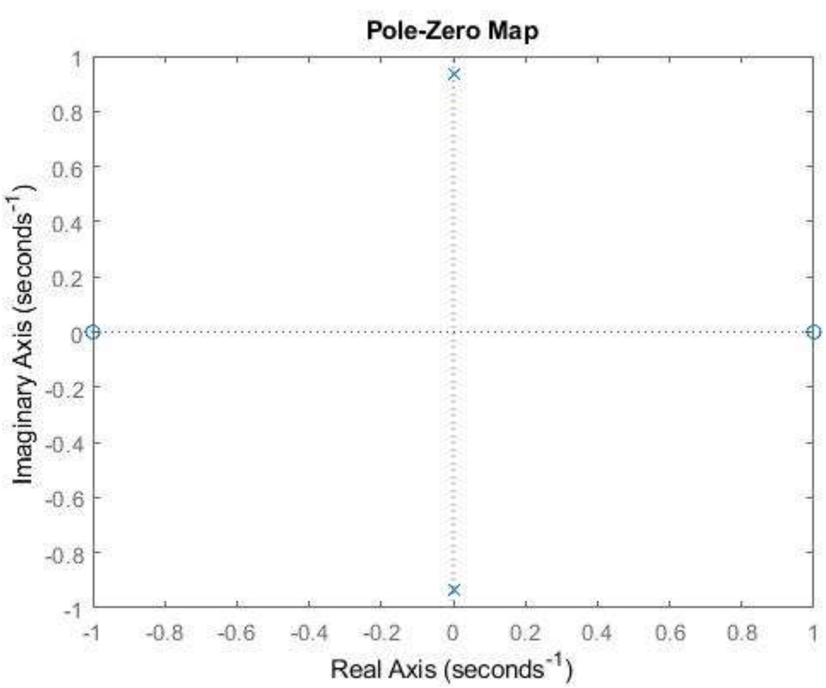
Code:

```
[P,Z] = pzmap([1 0 -1], [1 0 0.877]);%P = Pole, Z= Zero
```

```
pzmap([1 0 -1], [1 0 0.877]);
```

Output:





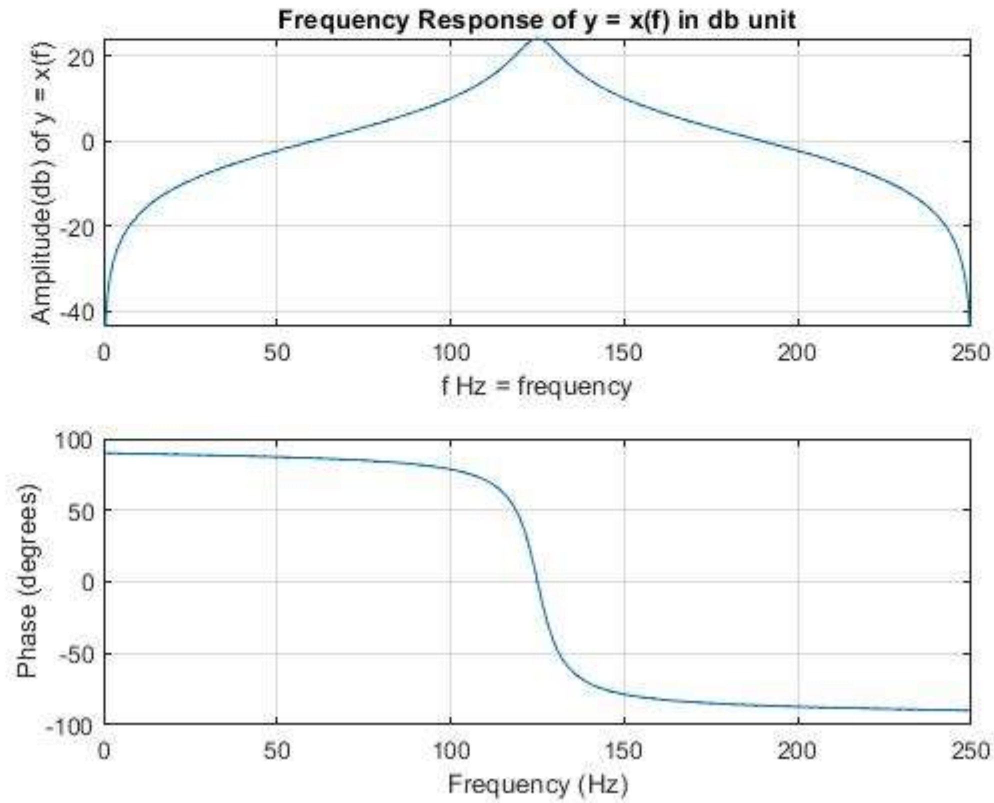
Observation:

The observations of the given transfer function is that if any transfer function is given then at first it should be converted into inverse z form so that we can easily find the Poles and Zeroes by applying the Matlab library function. Mention that the zeroes represent the cutoff frequency and the poles represent the peak value of the given frequency of that signal.

Code:

```
[P,Z] = pzmap([1 0 -1], [1 0 0.877]);%P = Pole, Z= Zero
pzmap([1 0 -1], [1 0 0.877]);

b = [1 0 -1];        % b coefficients of z transform
X(z)=b0+b1y1+b2y2/a0+a1x1+a2x2

a = [1 0 0.877];     % a coefficients of z transform
X(z)=b0+b1y1+b2y2/a0+a1x1+a2x2


Fs = 500;            % Fs = sampling frequency
freqz(b,a,512,Fs);% for frequency response in f-domain
                  % 501 = no of frequency samples
                  % Fs = sampling frequency used for
                  % normalization
title("Frequency Response of y = x(f) in db unit");
xlabel("f Hz = frequency");
ylabel("Amplitude(db) of y = x(f)");
```

Output:



Observation:

The observations of the given problem is that it represents the band-pass filter graph. At 100Hz to 200Hz it provides better band passing value.

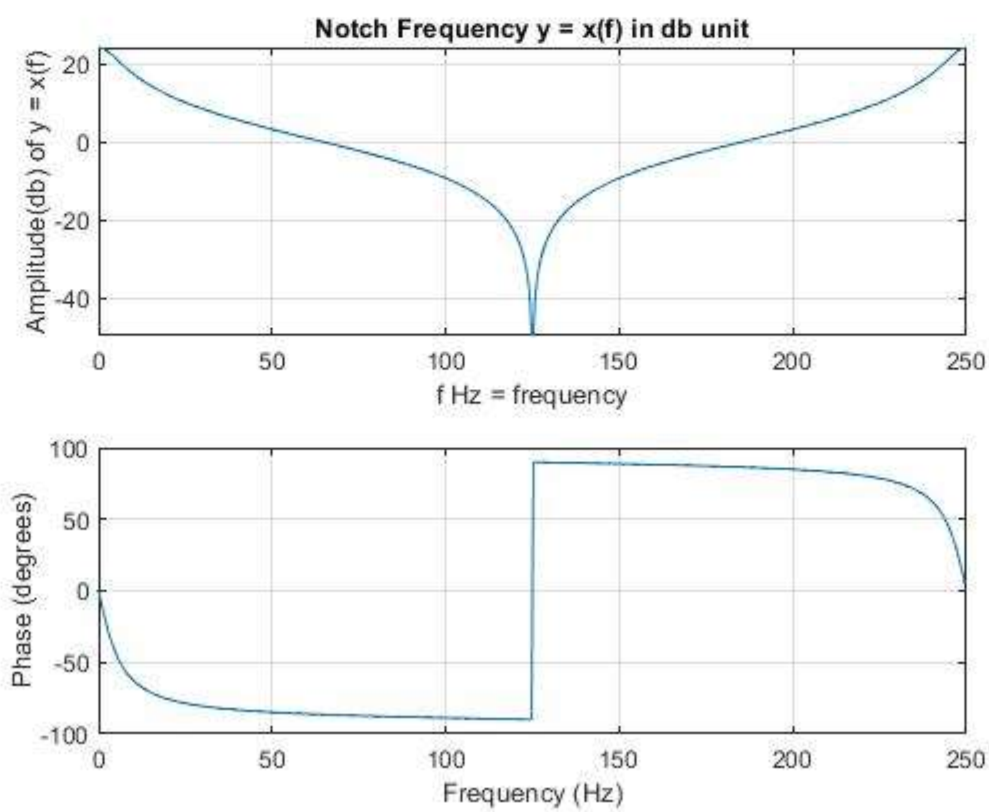
Question:

Observe the frequency response of, $H(z) = \dfrac{1-z^{-2}}{1-0.877z^{-2}}$ and comment on the result.

Solution:

```
% Notch Frequency
b = [1 0 1];        % b coefficients of z transform
X(z)=b0+b1y1+b2y2/a0+a1x1+a2x2
a = [1 0 -0.877]; % a coefficients of z transform
X(z)=b0+b1y1+b2y2/a0+a1x1+a2x2

Fs = 500;           % Fs = sampling frequency
freqz(b,a,501,Fs);% for frequency response in f-domain
                  % 501 = no of frequency samples
                  % Fs = sampling frequency used for
                  % normalization

title("Notch Frequency y = x(f) in db unit");
xlabel("f Hz = frequency");
ylabel("Amplitude(db) of y = x(f)");
```

Output:



Observation:

The observations of the given problem is that it represents the band-stop filter graph. At 100Hz to 200Hz it provides sharp notch frequency and at this point frequency is cutting off at all.