



Integrantes

- Henrique Augusto Gomes Togo - 18/0018531
- Vinícius Almeida Barros - 18/0028758

1 Resolução de questões

As resoluções das questões a seguir se referem àquelas encontradas no livro Biosignal and Medical Image Processing (3rd ed.)[1].

Questão 4.6

Nessa questão foi calculado espectro de magnitude, figura 1, e fase do sistema fornecido. Foram plotados tanto o espectro de magnitude quanto o espectro de fase desse sistema em decibéis com o eixo da frequência em escala logarítmica. Após isso foi calculada a resposta ao degrau do sistema, que pode ser vista na figura 2.

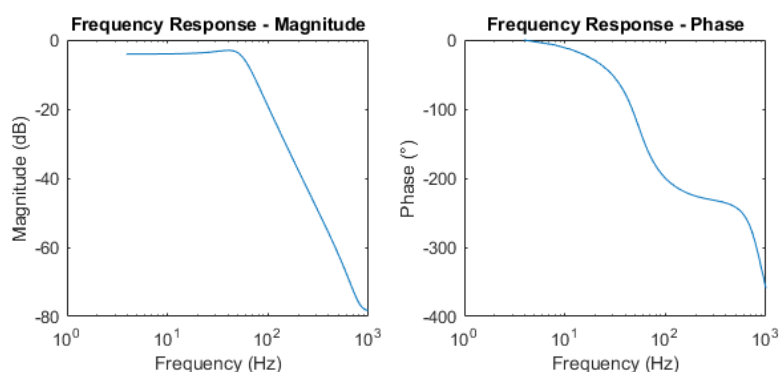


Figura 1:

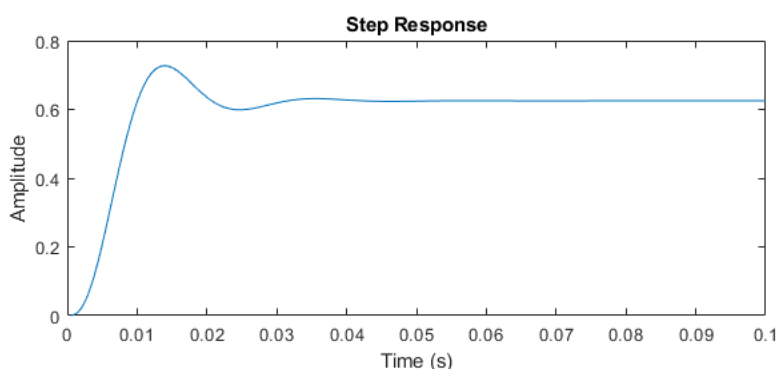


Figura 2:

O comportamento de $H(z)$ é correspondente ao de um filtro passa-baixas.

```
clear all;
close all;

fs = 2e3; % Frequency Sample
Ts = 1/fs; % Times Sample
N = 512; % Total points
b = [0.42e-3 1.25e-3 1.25e-3 -0.42e-3]; % Numerator filter coeffs
a = [1 -2.686 2.42 -0.73]; % Denominator filter coeffs
H = fft(b, N) ./ fft(a, N); % Filter Spectrum - Eq. 4.13
```

```

Hm = 20*log10(abs(H)); % Magnitude in dB
theta = unwrap(angle(H))*180/pi; % Phase in deg
f = (1:N)*fs/N; % Frequency vector
f_nyquist = f(1:N/2); % Nyquist frequency

figure(1); % Window 1
subplot(1,2,1); % Plot at the same window
semilogx(f_nyquist, Hm(1:N/2)); % Semilog (Log x axis) Magnitude
title("Frequency Response - Magnitude"); % Graph's title
ylabel("Magnitude (dB)"); % Y axis label
xlabel("Frequency (Hz)"); % X axis label
ylim([-80 0]); % Y scale from -80 to 0 dB
subplot(1,2,2); % Plot at the same window
semilogx(f_nyquist, theta(1:N/2)); % Semilog (Log x axis) Phase
title("Frequency Response - Phase"); % Graph's title
ylabel("Phase (°)"); % Y axis label
xlabel("Frequency (Hz)"); % X axis label

N_step = 200; % Number of points over a time period of 0.1s
t = (1:N_step) * Ts; % Time vector
x = ones(1,N_step); % Step
y = filter(b, a, x); % Step response
figure(2); % Window 2
plot(t, y); % Plot the step response
title("Step Response"); % Graph's title
ylabel("Amplitude"); % Y axis label
xlabel("Time (s)"); % X axis label

```

Questão 4.9

A questão pede o espectro de magnitude de um filtro FIR, figura 3. Para isso, foram utilizados dois métodos distintos: o primeiro foi feito utilizando a FFT com padding e o segundo foi passando um ruído branco por um filtro utilizando a rotina conv e depois estimando o espectro de amplitude pelo método de Welch. Pode-se perceber que os dois métodos utilizam escalas diferentes e, por isso, a escala vertical nos gráficos são ligeiramente diferentes.

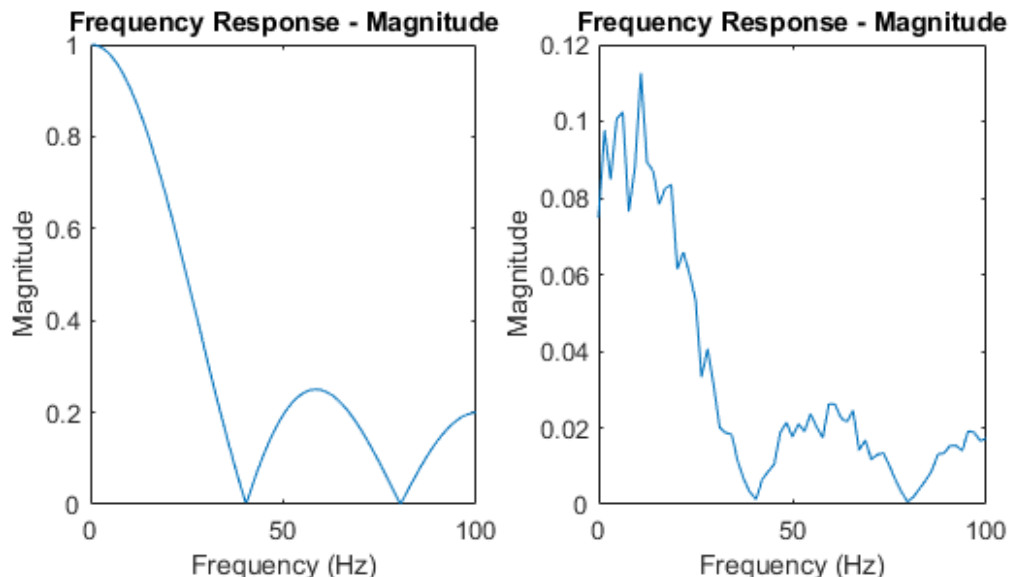


Figura 3:

```

clear all;
close all;

fs = 200; % Frequency Sample
N = 512; % Total points

```

```

b = [.2 .2 .2 .2 .2]; % Numerator filter coeffs
segment_size = 128;
noverlap = round(0.5*segment_size);

H = fft(b, N); % Filter Spectrum - Eq. 4.13
Hm = abs(H); % Magnitude
f = (1:N)*fs/N; % Frequency vector
f_nyquist = f(1:N/2); % Nyquist frequency
y_f = conv(b, randn(1,20000)); % Filter's output
[PS, f_welch] = pwelch(y_f, [], noverlap, segment_size, fs); % Welch filter's output
AS = sqrt(PS); % Convert power to amplitude spectrum

figure(1); % Window 1
subplot(1,2,1); % Plot at the same window
plot(f_nyquist, Hm(1:N/2)); % Semilog (Log x axis) Magnitude
title("Frequency Response - Magnitude"); % Graph's title
ylabel("Magnitude"); % Y axis label
xlabel("Frequency (Hz)"); % X axis label

subplot(1,2,2); % Plot at the same window
plot(f_welch, AS); % Semilog (Log x axis) Magnitude
title("Frequency Response - Magnitude"); % Graph's title
ylabel("Magnitude"); % Y axis label
xlabel("Frequency (Hz)"); % X axis label

```

Questão 4.12

Nessa questão, são comparadas três janelas no domínio da frequência: Blackman-Harris, Hamming e retangular. Foi construído um filtro passa-baixas com frequência de corte de 200 Hz e ordem 33. Após geradas as janelas, estas foram aplicadas aos coeficientes b . Depois disso, foi encontrado o espectro de magnitude das três janelas como pode ser visto na figura 4. Como pode ser visto a janela Hamming possui uma atenuação mais abrupta.

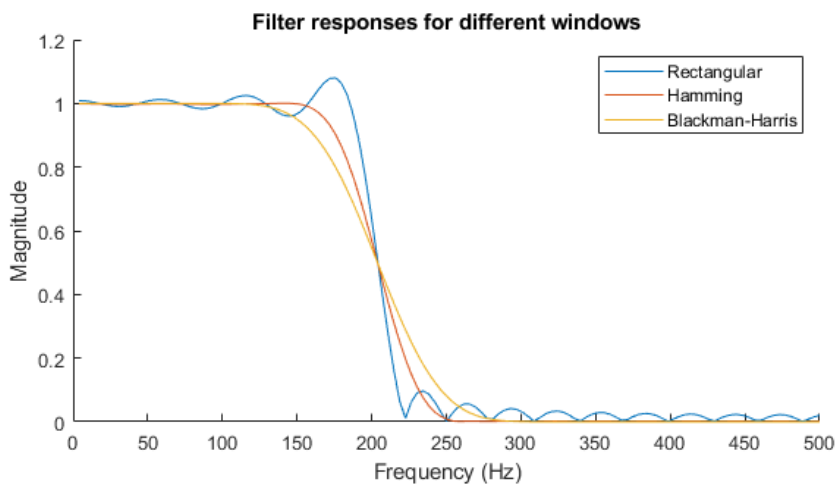


Figura 4:

```

close all;
clear all;

fs = 1e3; % Frequency sample
N = 256; % Pad to a total of N points
cutoff = 200; % Cutoff frequency
fc = cutoff/fs; % Relative cutoff frequency
f = (1:N) * fs/N;
L = 33; % Filter size
N = 256; % Padding to N points

```

```

k = -floor(L/2):-1; % negative values of k
b = sin(2*pi*fc*k)/(pi*k); % Calculate b coeffs for lowpass rectangular
b = [b 2*fc, fliplr(b)]; % b coeffs left, limit value, flip to right

n_ham = 1:L; % n vector to generate window
ham = 0.54 - 0.46*cos((2*pi*n_ham)/L); % Hamming window equation
n_black = -floor(L/2):floor(L/2); % n vector to generate window
a_window = [0.35875 0.48829 0.14128 0.01168]; % a coeffs for blackman-harris window
black = a_window(1) + a_window(2)*cos((2*pi*n_black)/L) + a_window(3)*cos(2*(2*pi*
    n_black)/L) + a_window(4)*cos(3*(2*pi*n_black)/L); % Blackman-Harris window
    equation
b_ham = b .* ham; % hamming windowed coeffs
b_black = b .* black; % blackman-harris windowed coeffs

H_rec = fft(b, N); % H spectrum rectangular window
H_rec_mag = abs(H_rec); % H rectangular magnitude spectrum
H_ham = fft(b_ham, N); % H spectrum rectangular window
H_ham_mag = abs(H_ham); % H rectangular magnitude spectrum
H_black = fft(b_black, N); % H spectrum rectangular window
H_black_mag = abs(H_black); % H rectangular magnitude spectrum

figure(1); % Window 1
hold on; % Plot at the same graph
plot(f(1:N/2), H_rec_mag(1:N/2)); % Plot rectangular window filter
plot(f(1:N/2), H_ham_mag(1:N/2)); % Plot rectangular window filter
plot(f(1:N/2), H_black_mag(1:N/2)); % Plot rectangular window filter
title("Filter responses for different windows"); % Graph's title
legend('Rectangular', 'Hamming', 'Blackman-Harris'); % Display legends
xlabel("Frequency (Hz)"); % X axis label
ylabel("Magnitude"); % Y axis label

```

Questão 4.13

Nessa questão, são comparadas três janelas no domínio do tempo: Blackman-Harris, Hamming e retangular. Foi construído um filtro passa-baixas com frequência de corte de 100 Hz e ordem 129. Após geradas as janelas, estas foram aplicadas aos coeficientes b . Depois disso, foi encontrado a resposta ao impulso das três janelas como pode ser visto na figura 5. Como pode ser visto a janela Hamming possui uma atenuação mais abrupta. Como pode ser visto, as respostas são praticamente idênticas. Na figura 6, é plotada a diferença entre a resposta ao impulso das janelas Blackman-Harris e Hamming e, como pode ser visto, a diferença é da ordem de 10^{-3} .

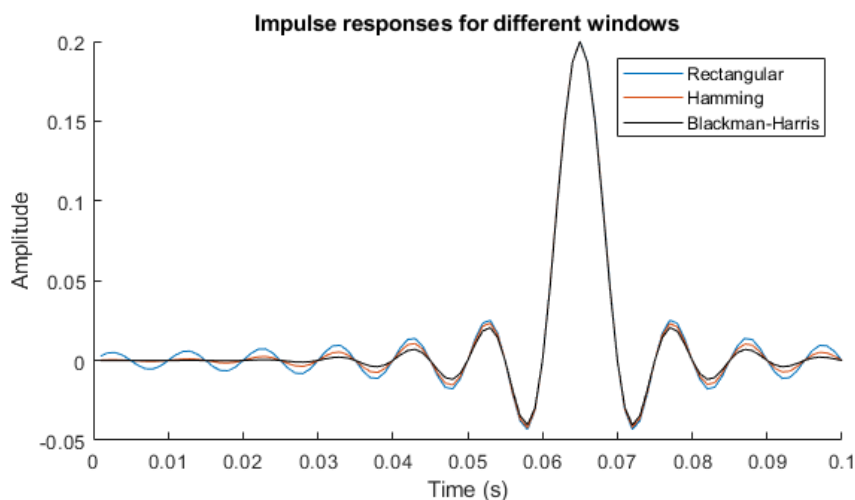


Figura 5:

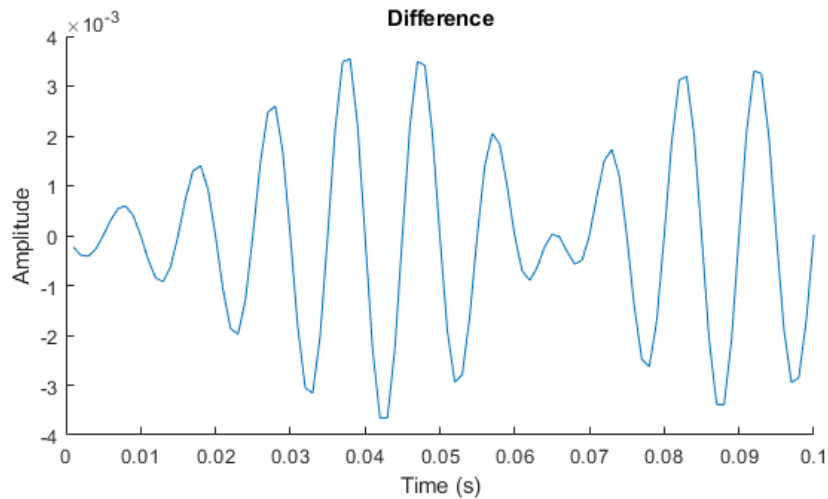


Figura 6:

```
close all;
clear all;

fs = 1e3; % Frequency sample
N = 256; % Pad to a total of N points
cutoff = 100; % Cutoff frequency
fc = cutoff/fs; % Relative cutoff frequency
f = (1:N) * fs/N;
L = 129; % Filter size
k = -floor(L/2):-1; % negative values of k
b = sin(2*pi*fc*k)./(pi*k); % Calculate b coeffs for lowpass rectangular
b = [b 2*fc, fliplr(b)]; % b coeffs left, limit value, flip to right

n_ham = 1:L; % n vector to generate window
ham = 0.54 - 0.46*cos((2*pi*n_ham)/L); % Hamming window equation
n_black = -floor(L/2):floor(L/2); % n vector to generate window
a_window = [0.35875 0.48829 0.14128 0.01168]; % a coeffs for blackman-harris window
black = a_window(1) + a_window(2)*cos((2*pi*n_black)/L) + a_window(3)*cos(2*(2*pi*
    n_black)/L) + a_window(4)*cos(3*(2*pi*n_black)/L); % Blackman-Harris window
    equation
b_ham = b .* ham; % hamming windowed coeffs
b_black = b .* black; % blackman-harris windowed coeffs

H_rec = fft(b, N); % H spectrum rectangular window
H_rec_mag = abs(H_rec); % H rectangular magnitude spectrum
H_ham = fft(b_ham, N); % H spectrum rectangular window
H_ham_mag = abs(H_ham); % H rectangular magnitude spectrum
H_black = fft(b_black, N); % H spectrum rectangular window
H_black_mag = abs(H_black); % H rectangular magnitude spectrum

impulse = [1 zeros(1,N-1)]; % Impulse vector
y_rec = filter(b, 1, impulse); % Impulse response rectangular window
y_ham = filter(b_ham, 1, impulse); % Impulse response Hamming window
y_black = filter(b_black, 1, impulse); % Impulse response Blackman-Harris window
t = (1:N)/fs; % Time vector

figure(1); % Window 1
hold on; % Plot at the same graph
plot(t, y_rec); % Plot impulse response rectangular window filter
plot(t, y_ham); % Plot impulse response hamming window filter
plot(t, y_black, 'k'); % Plot impulse response blackman-harris window filter
title("Impulse responses for different windows"); % Graph's title
legend('Rectangular', 'Hamming', 'Blackman-Harris'); % Display legends
xlabel("Time (s)"); % X axis label
```

```

ylabel("Amplitude"); % Y axis label
xlim([0 0.1]); % Sscale x axis from 0 up to 0.1s

figure(2); % Window 1
hold on; % Plot at the same graph
plot(t, y_black - y_ham); % Plot impulse response rectangular window filter
title("Difference"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label
xlim([0 0.1]); % Sscale x axis from 0 up to 0.1s

```

Questão 4.14

Nessa questão, foi carregado um sinal de eletrocardiograma de 9 segundos. Foi projetado um filtro passa-altas de ordem 65 e frequência de corte de 8 Hz com a janela Hamming. A figura 7 apresenta o espectro da magnitude do filtro passa altas. A figura 8 mostra o sinal de ECG antes da filtragem e a figura 9 mostra o sinal filtrado. Como pode ser observado, o sinal de baixa frequência foi eliminado do ECG, facilitando a interpretação dos picos.

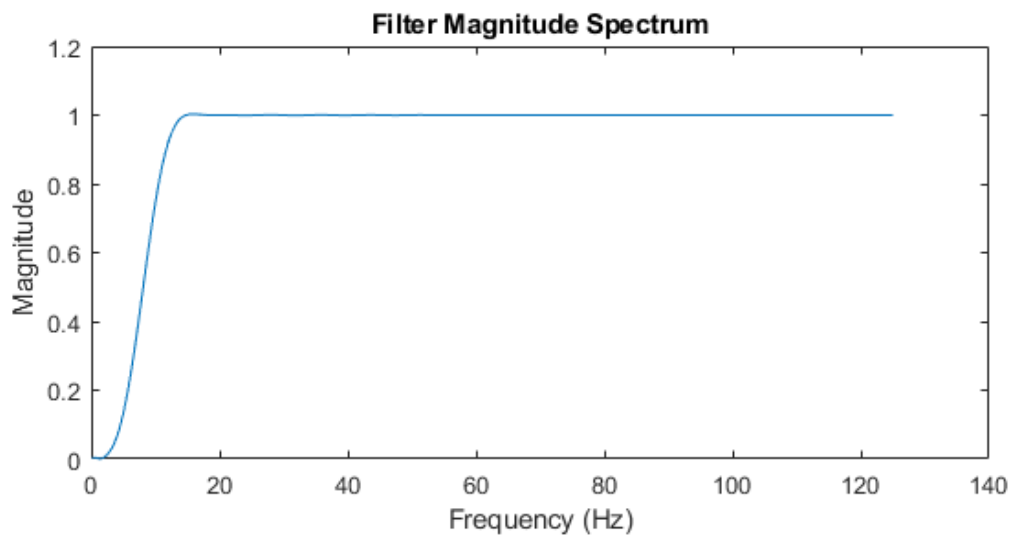


Figura 7:

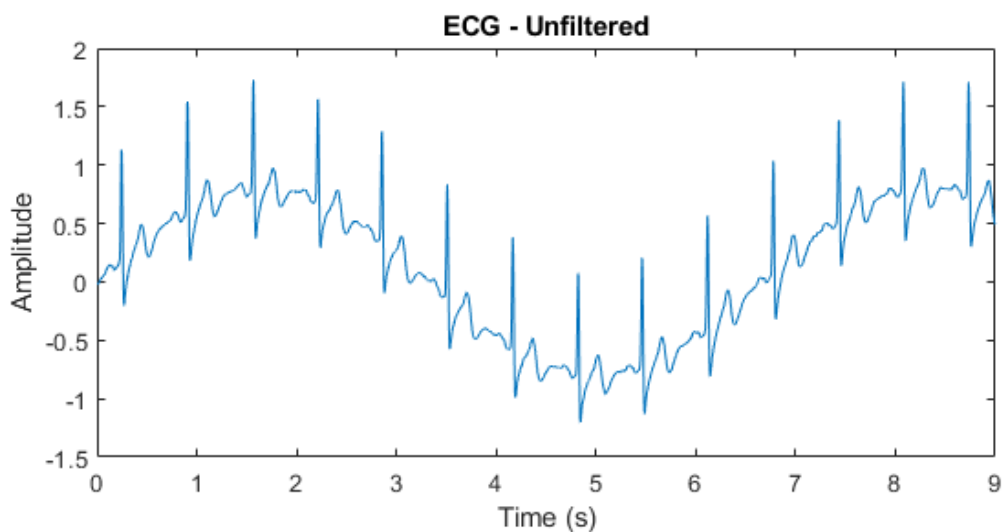


Figura 8:

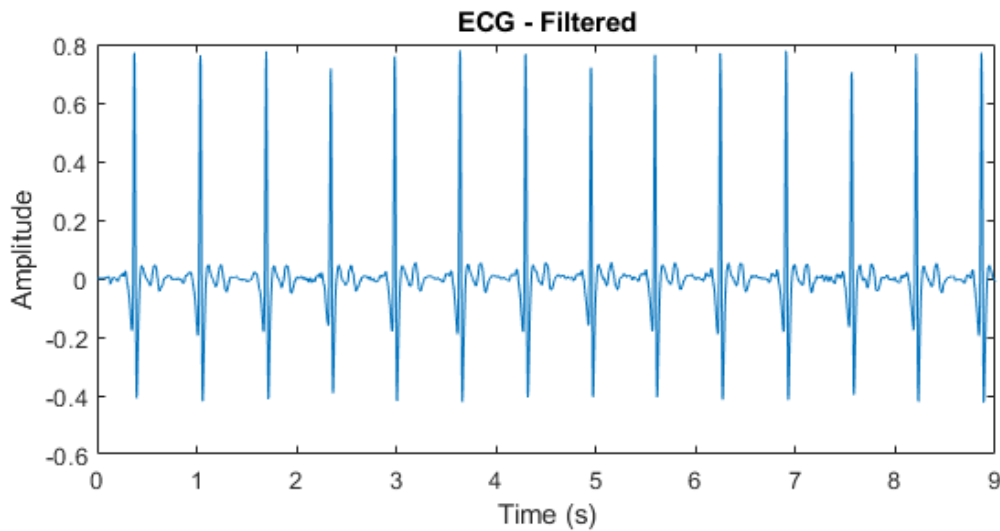


Figura 9:

```
close all;
clear all;

load ECG_9
fs = 250; % Sample Frequency

%Highpass Filter
N = length(x); % Pad to a total of N points
cutoff = 8; % Cutoff frequency
fc = cutoff/fs; % Relative cutoff frequency
f = (1:N) * fs/N; % Frequency vector
t = (1:N) / fs; % Time vector
L = 65; % Filter size
k = -floor(L/2):-1; % negative values of k
b = -sin(2*pi*fc*k)./(pi*k); % Calculate b coeffs for highpass rectangular
b = [b 1 - 2*fc, flipplr(b)]; % b coeffs left, limit value, flip to right
b_ham = b .* hamming(L)'; % hamming windowed coeffs
b_ham_mag = abs(fft(b_ham, N)); % Filter Magnitude Spectrum

x_filtered = filter(b_ham, 1, x); % Filtered ECG2

figure(1); %Window 1
plot(f(1:N/2), b_ham_mag(1:N/2)); % Plot magnitude spectrum
title("Filter Magnitude Spectrum"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Magnitude"); % Y axis label

figure(2); % Window 2
plot(t, x); % Plot ECG data - unfiltered
title("ECG - Unfiltered"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label

figure(3); % Window 3
plot(t, x_filtered); % Plot ECG data - unfiltered
title("ECG - Filtered"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label
```

Questão 4.15

Nessa questão foi carregado um sinal ECG com ruído de baixa e alta frequência como pode ser visto na figura 11. Para tentar eliminar os ruídos, tanto de alta quanto baixa frequência, foi projetado um filtro passa-bandas

do tipo FIR de ordem 65. Foi aplicada uma janela hamming nos coeficientes b. O espectro de magnitude do filtro pode ser visto na figura 10. Por fim, as frequência de corte do filtro foram determinadas experimentalmente em 6 e 14 Hz. O sinal do ECG filtrado pode ser visto na figura 12. Como pode ser notado, não se eliminou todo o ruído do ECG, mas já foi obtida uma grande melhoria do sinal.

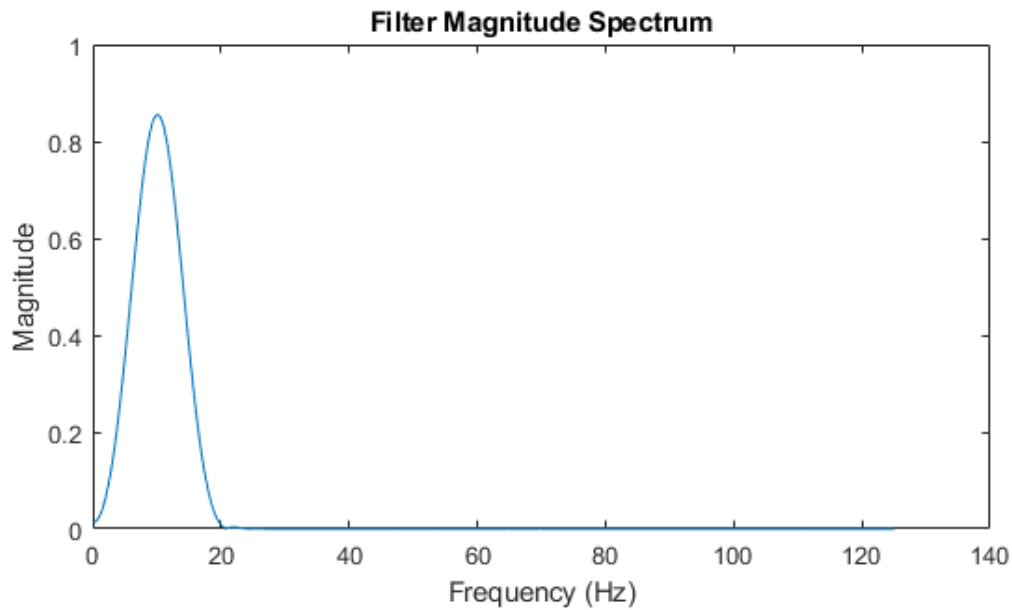


Figura 10:

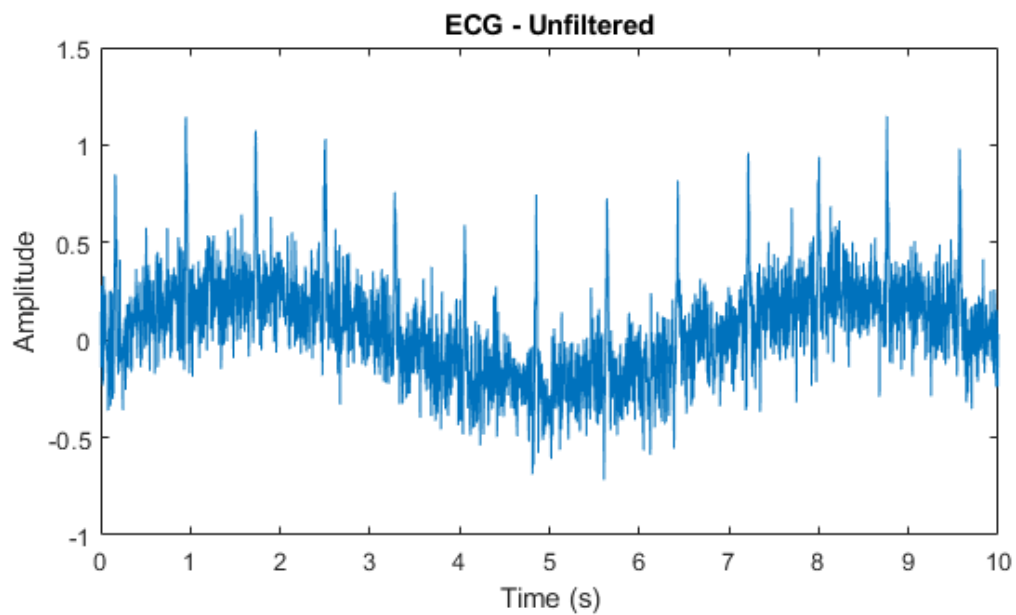


Figura 11:

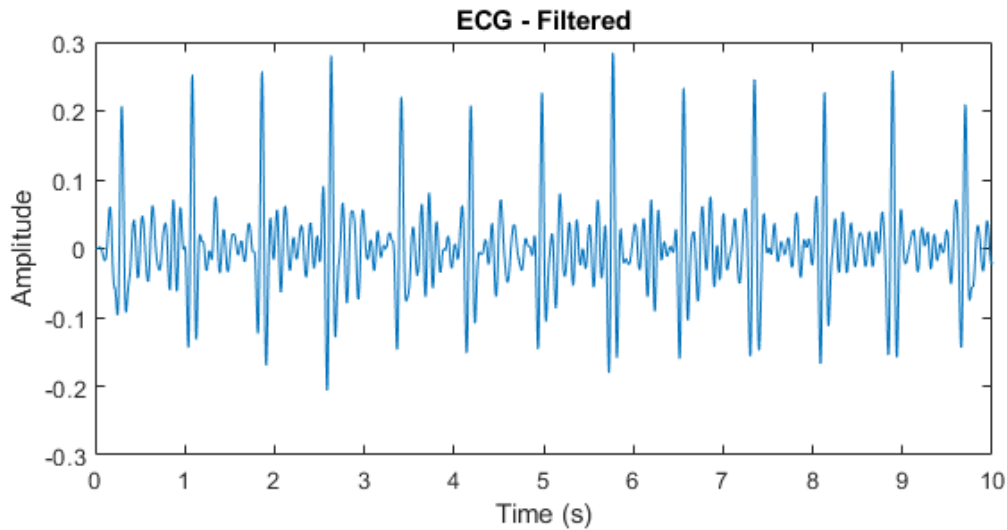


Figura 12:

```
close all;
clear all;

load ECG_noise
fs = 250; % Sample Frequency

%Bandpass filter
N = length(ecg); % Pad to a total of N points
cutoff_h = 14; % High Cutoff frequency
cutoff_l = 6; % Low Cutoff frequency
fh = cutoff_h/fs; % Relative cutoff frequency
fl = cutoff_l/fs; % Relative cutoff frequency
f = (1:N) * fs/N; % Frequency vector
t = (1:N) / fs; % Time vector
L = 65; % Filter size
k = -floor(L/2):-1; % negative values of k
b = sin(2*pi*f*h*k)./(pi*k)-sin(2*pi*f*l*k)./(pi*k); % Calculate b coeffs for bandpass
    rectangular
b = [b 2*(fh - fl), fliplr(b)]; % b coeffs left, limit value, flip to right
b_ham = b .* hamming(L)'; % hamming windowed coeffs
b_ham_mag = abs(fft(b_ham, N)); % Filter Magnitude Spectrum

ecg_filtered = filter(b_ham, 1, ecg); % Filtered ECG2

figure(1); %Window 1
plot(f(1:N/2), b_ham_mag(1:N/2)); % Plot magnitude spectrum
title("Filter Magnitude Spectrum"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Magnitude"); % Y axis label

figure(2); % Window 2
plot(t, ecg); % Plot ECG data - unfiltered
title("ECG - Unfiltered"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label

figure(3); % Window 3
plot(t, ecg_filtered); % Plot ECG data - unfiltered
title("ECG - Filtered"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label
```

Questão 4.19

Nessa questão, foi fornecido um sinal de ECG corrompido por um sinal de 60 Hz. Comparativamente à frequência do ECG, o sinal de 60 Hz está acima da frequência do sinal. Para tentar eliminar o ruído, foi projetado um filtro rejeita-faixas de ordem 127 do tipo FIR. A frequência central é de 60 Hz com uma banda de 10 Hz (± 5 Hz). A figura 13 representa o espectro de magnitude do filtro projetado. O sinal ruidoso do ECG, figura 14, foi filtrado com a rotina filter e o resultado pode ser visto na figura 15. Como pode ser observado, a utilização da rotina filter resulta em um atraso do sinal do ECG, mas de fato o ruído de 60 Hz é retirado do sinal. Dessa forma, é facilitada a interpretação do sinal de ECG.

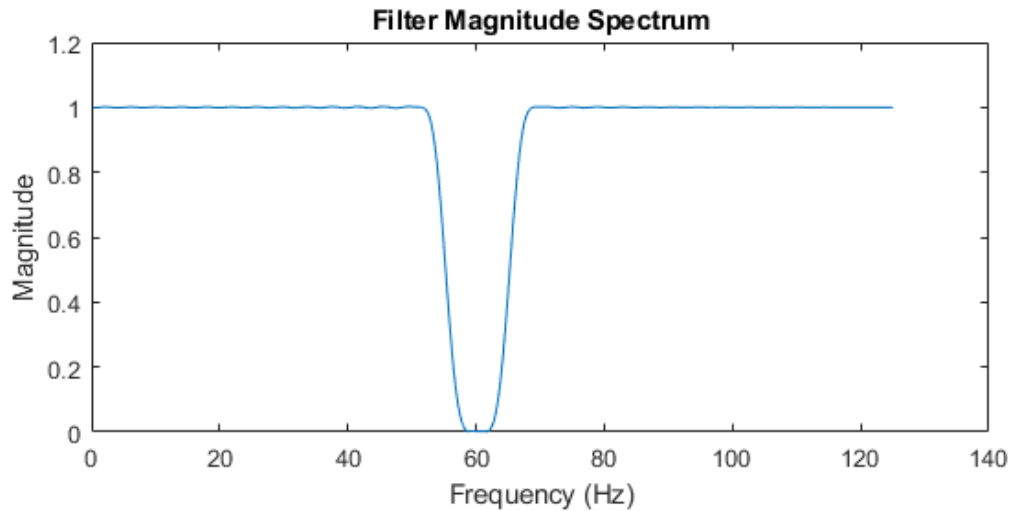


Figura 13:

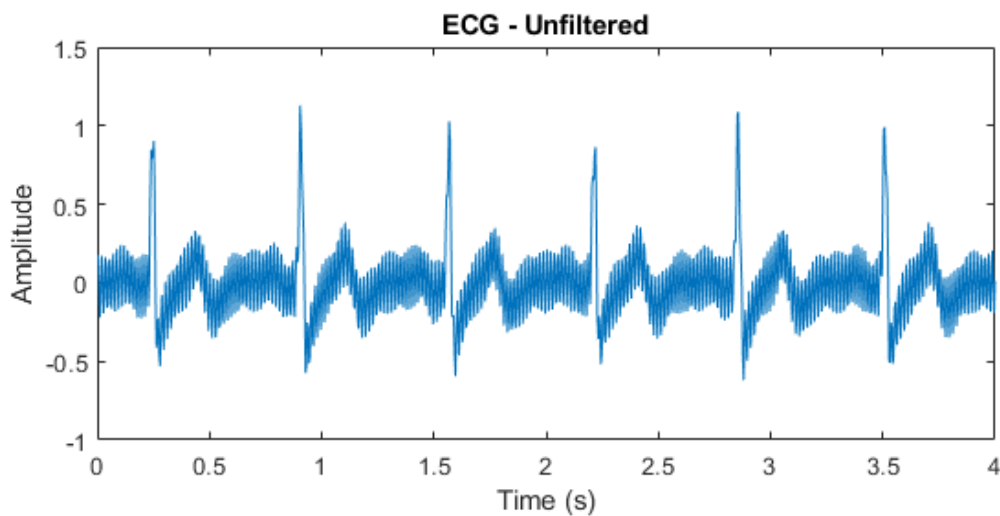


Figura 14:

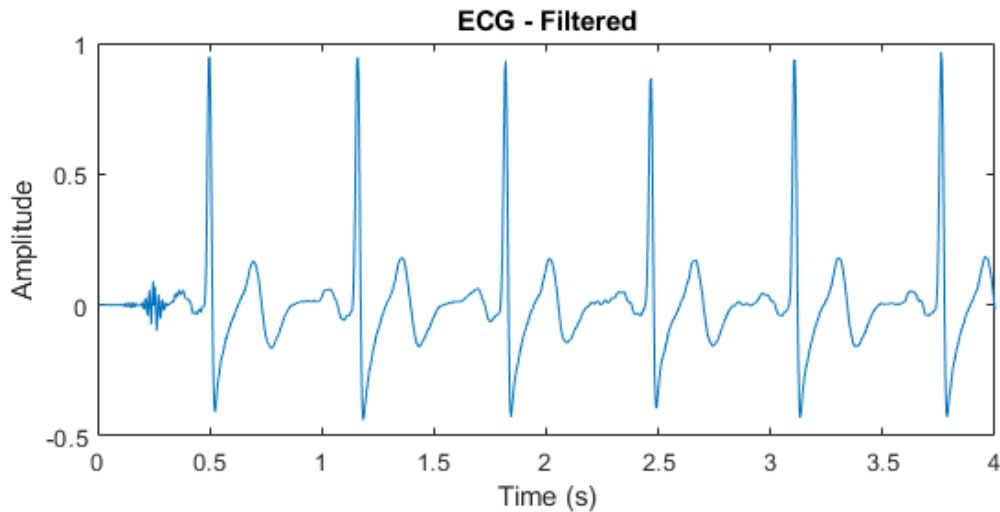


Figura 15:

```
close all;
clear all;

load ECG_60Hz_data
fs = 250; % Sample Frequency

%Bandstop Filter
N = length(x); % Pad to a total of N points
cutoff_h = 65; % High Cutoff frequency
cutoff_l = 55; % Low Cutoff frequency
fh = cutoff_h/fs; % Relative cutoff frequency
fl = cutoff_l/fs; % Relative cutoff frequency
f = (1:N) * fs/N; % Frequency vector
t = (1:N) / fs; % Time vector
L = 127; % Filter size
k = -floor(L/2):-1; % negative values of k
b = sin(2*pi*fl*k)./(pi*k)-sin(2*pi*fh*k)./(pi*k); % Calculate b coeffs for bandstop
    rectangular
b = [b 1 - 2*(fh - fl), fliplr(b)]; % b coeffs left, limit value, flip to right
b_ham = b .* hamming(L)'; % hamming windowed coeffs
b_ham_mag = abs(fft(b_ham, N)); % Filter Magnitude Spectrum

ecg_filtered = filter(b_ham, 1, x); % Filtered ECG2

figure(1); %Window 1
plot(f(1:N/2), b_ham_mag(1:N/2)); % Plot magnitude spectrum
title("Filter Magnitude Spectrum"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Magnitude"); % Y axis label

figure(2); % Window 2
plot(t, x); % Plot ECG data - unfiltered
title("ECG - Unfiltered"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label

figure(3); % Window 3
plot(t, ecg_filtered); % Plot ECG data - unfiltered
title("ECG - Filtered"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label
```

Questão 4.20

Nessa questão, foi fornecido um sinal ruidoso, figura 17, e o intuito era o de comparar a implementação de filtros causais e não-causais através das rotinas `filter` e `conv`, respectivamente. Foi projetado um filtro passa-baixas do tipo FIR de ordem 65 com uma frequência de corte de 40 Hz, figura 16. Ao filtro foi aplicada uma janela blackman-harris e posteriormente os filtros foram implementados de forma causal, figura 18 (`filter`), e de forma não-causal, figura 19 (`conv`). Como pode ser visto pelos gráficos, o filtro não-causal acaba gerando um atraso do sinal que é filtrado. Além disso, ao mesmo tempo em que os filtros removem grande parte do ruído, eles também removem a rápida transição do sinal (descontinuidades).

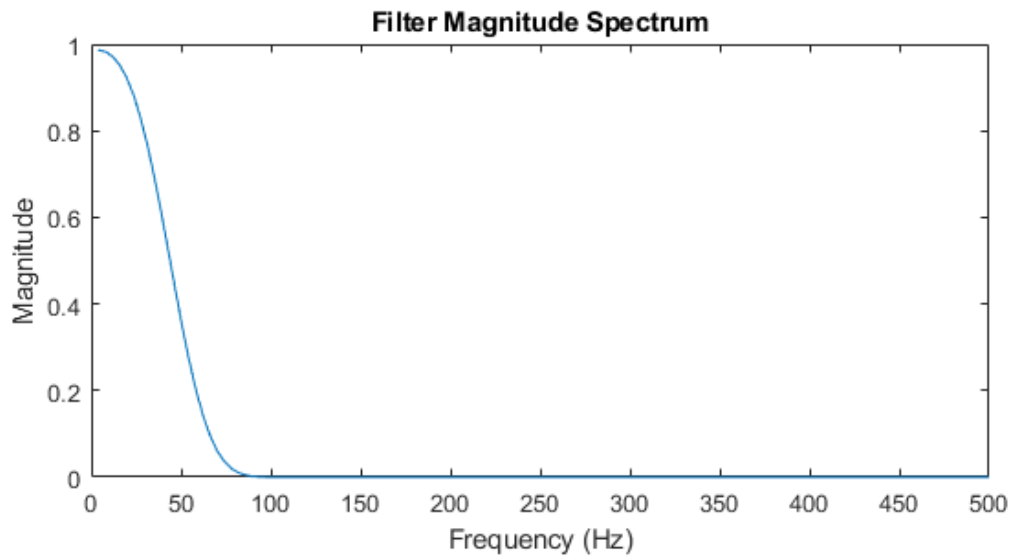


Figura 16:

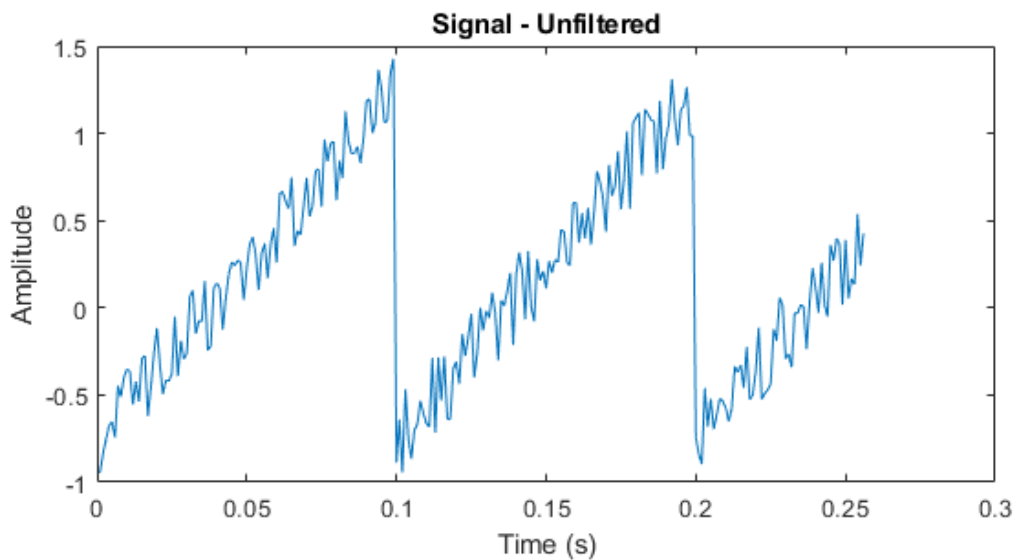


Figura 17:

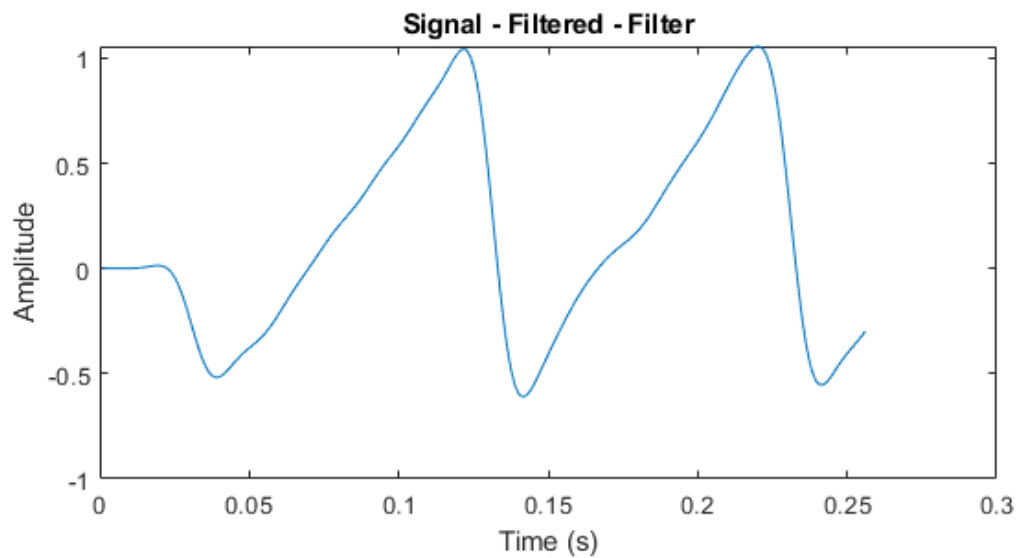


Figura 18:

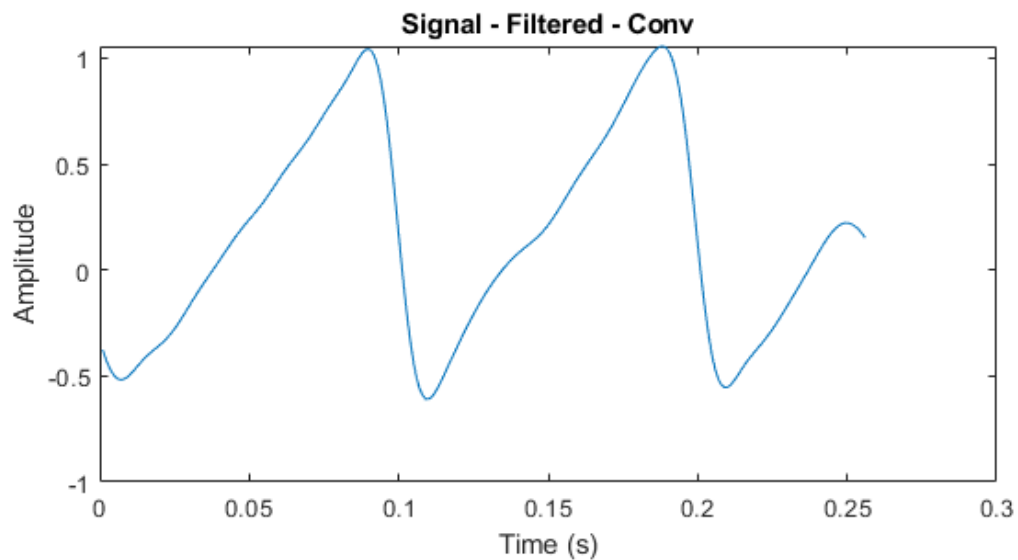


Figura 19:

```
close all;
clear all;

load sawth
fs = 1e3; % Sample Frequency

%Lowpass Filter
N = length(x); % Pad to a total of N points
cutoff = 40; % Cutoff frequency
fc = cutoff/fs; % Relative cutoff frequency
f = (1:N) * fs/N; % Frequency vector
t = (1:N) / fs; % Time vector
L = 65; % Filter size
k = -floor(L/2):-1; % negative values of k
b = sin(2*pi*fc*k)./(pi*k); % Calculate b coeffs for lowpass rectangular
b = [b 2*fc, fliplr(b)]; % b coeffs left, limit value, flip to right
b_black = b .* blackmanharris(L)'; % hamming windowed coeffs
b_black_mag = abs(fft(b_black, N)); % Filter Magnitude Spectrum

x_filtered = filter(b_black, 1, x); % Filtered signal (filter)
```

```

x_conv = conv(x, b_black, 'same'); % Filtered signal (conv)

figure(1); %Window 1
plot(f(1:N/2), b_black_mag(1:N/2)); % Plot magnitude spectrum
title("Filter Magnitude Spectrum"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Magnitude"); % Y axis label

figure(2); % Window 2
plot(t, x); % Plot ECG data - unfiltered
title("Signal - Unfiltered"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label

figure(3); % Window 3
plot(t, x_filtered); % Plot ECG data - unfiltered
title("Signal - Filtered - Filter"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label

figure(4); % Window 3
plot(t, x_conv); % Plot ECG data - unfiltered
title("Signal - Filtered - Conv"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label

```

Questão 4.21

Nessa questão, foi projetado um filtro de ordem 33 e uma frequência de corte de 100 Hz. A janela do tipo blackman-harris foi utilizada. Depois disso, foi gerada uma função impulso que consiste em 256 valores, sendo o primeiro igual a 1 e os demais iguais a 0. Em seguida, aplicou-se o filtro causal, figura 20, e o não causal, figura 21. Como pode ser observado, o filtro não causal acaba truncando alguns valores do sinal filtrado no domínio do tempo. Em seguida, foram obtidos os espectros de magnitude e fase dos sinais filtrados com a rotina filter, figura 22, e com a rotina conv, figura 23.

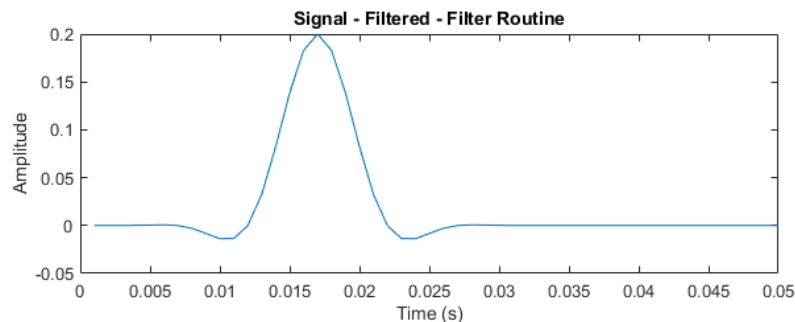


Figura 20:

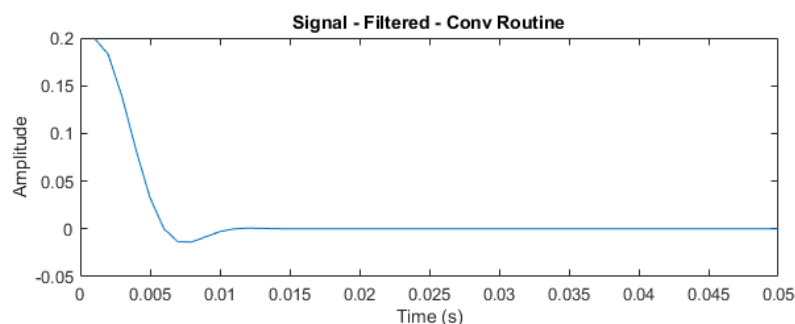


Figura 21:

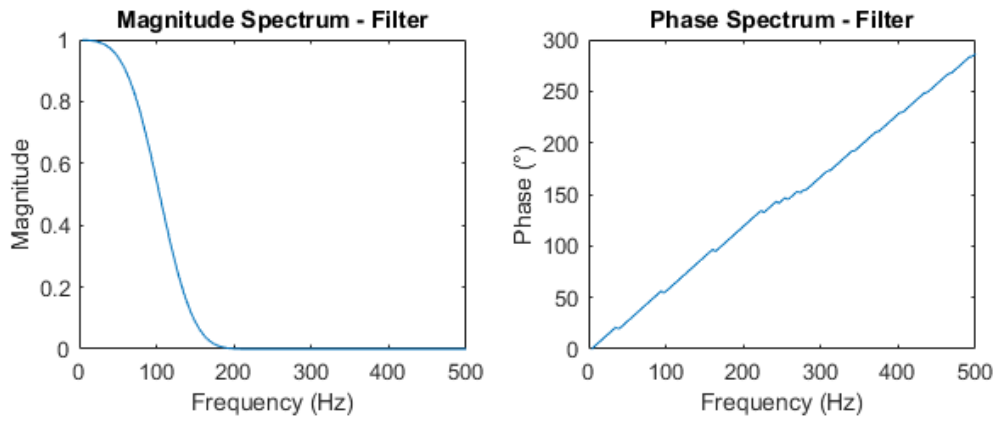


Figura 22:

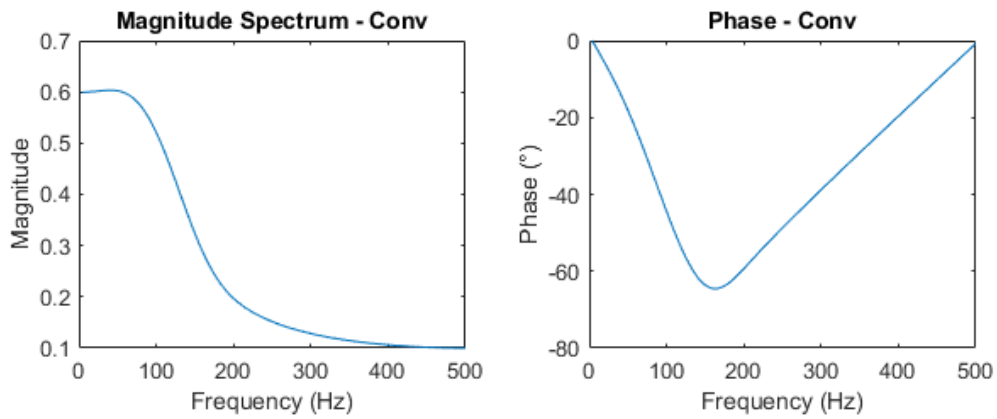


Figura 23:

Pode-se observar que o espectro gerado a partir do sinal filtrado com a rotina conv apresenta um espectro um tanto quanto estranho. Isso se deve ao fato de que parte do impulso foi truncado. Para comprovar esse comportamento, foi gerado um impulso com um atraso de 10 amostras e obteve-se o espectro de magnitude e fases, figuras 24 e 25. Dessa vez, é possível observar que os espectros de magnitude são iguais. Uma desvantagem do filtro não causal é a possibilidade da geração de artefatos devido ao truncamento da porção inicial do sinal de entrada.

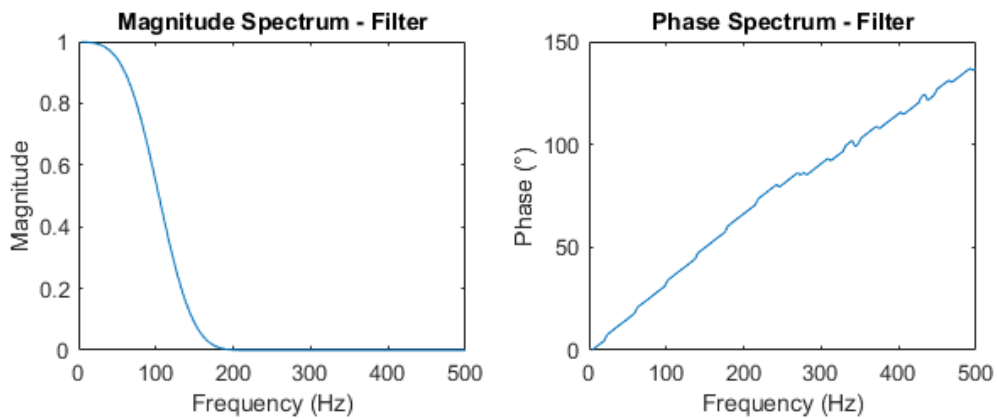


Figura 24:

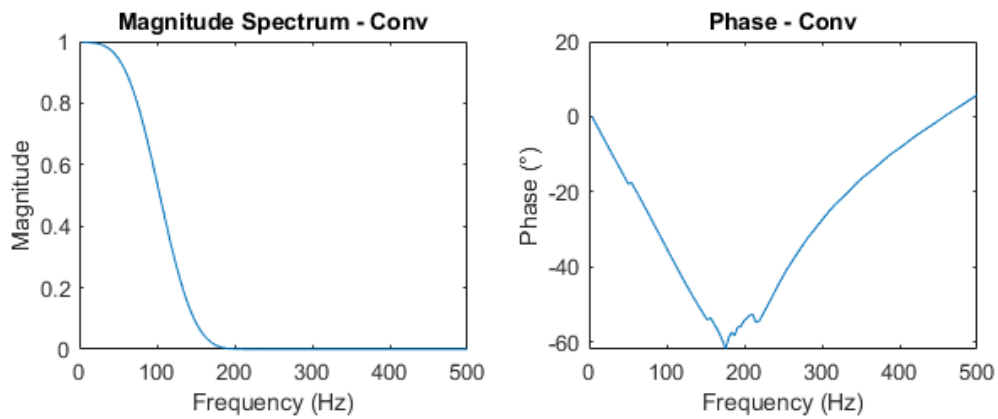


Figura 25:

```

close all;
clear all;

fs = 1e3; % Sample Frequency
%Lowpass Filter
N = 256; % Pad to a total of N points
cutoff = 100; % Cutoff frequency
fc = cutoff/fs; % Relative cutoff frequency
f = (1:N) * fs/N; % Frequency vector
t = (1:N) / fs; % Time vector
L = 33; % Filter size
k = -floor(L/2):-1; % negative values of k
b = sin(2*pi*fc*k)./(pi*k); % Calculate b coeffs for lowpass rectangular
b = [b 2*fc, fliplr(b)]; % b coeffs left, limit value, flip to right
b_black = b .* blackmanharris(L)'; % hamming windowed coeffs
b_black_mag = abs(fft(b_black, N)); % Filter Magnitude Spectrum

x = [1 zeros(1, N-1)]; % Impulse
x_delay = [zeros(1, 10) 1 zeros(1, 245)]; % Delayed Impulse
x_vector = [x; x_delay]; % Line 1: Impulse (No delay), Line 2: Impulse (Delay)

x_filtered = zeros(2,N); % Initialize vector
x_conv = zeros(2,N); % Initialize vector

for idx=1:2
    x_filtered(idx,:) = filter(b_black, 1, x_vector(idx,:)); % Filtered signal (
    filter)
    x_conv(idx,:) = conv(x_vector(idx,:), b_black, 'same'); % Filtered signal (conv)
end
figure(1); % Window 1
plot(t, x_filtered(1,:)); % Plot signal - filtered (causally)
title("Signal - Filtered - Filter Routine"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label
xlim([0 0.05]); % X axis scale

figure(2); % Window 2
plot(t, x_conv(1,:)); % Plot signal - filtered (non-causally)
title("Signal - Filtered - Conv Routine"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label
xlim([0 0.05]); % X axis scale

for idx=1:2
    X_filtered = fft(x_filtered(idx,:)); % FFT of impulse response (causal)
    X_conv = fft(x_conv(idx,:)); % FFT of impulse response (non-causal)
    X_filtered_mag = abs(X_filtered); % Magnitude of impulse response (causal)

```



```

X_conv_mag = abs(X_conv); % Magnitude of impulse response (non-causal)
X_filtered_phase = unwrap(angle(X_filtered)*180/pi); % Phase of impulse spectrum
(causal)
X_conv_phase = unwrap(angle(X_conv)*180/pi); % Phase of impulse spectrum (non-
causal)

figure, % Window 3
subplot(1,2,1); % Plot at the same window
plot(f(1:N/2), X_filtered_mag(1:N/2)); % Plot Magnitude Spectrum - Causal
title("Magnitude Spectrum - Filter"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Magnitude"); % Y axis label
subplot(1,2,2); % Plot at the same window
plot(f(1:N/2), X_filtered_phase(1:N/2)); % Plot Phase Spectrum - Causal
title("Phase Spectrum - Filter"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Phase (°)"); % Y axis label

figure, % Window 4
subplot(1,2,1); % Plot at the same window
plot(f(1:N/2), X_conv_mag(1:N/2)); % Plot Magnitude Spectrum - Non-Causal
title("Magnitude Spectrum - Conv"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Magnitude"); % Y axis label
subplot(1,2,2); % Plot at the same window
plot(f(1:N/2), X_conv_phase(1:N/2)); % Plot Phase Spectrum - Non-Causal
title("Phase - Conv"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Phase (°)"); % Y axis label
end

```

Questão 4.24

Para essa questão, buscou-se observar uma comparação entre um filtro causal e um não causal do tipo IIR. Para isso, foi disponibilizado um sinal de respiração contendo ruído, figura 27 e projetou-se um filtro de ordem 14 do tipo Butterworth com uma frequência de corte de $0.15 \frac{f_s}{2}$. Nesse caso, como a rotina utilizada foi a butter para projetar o filtro, a frequência de corte é relativa a $\frac{f_s}{2}$. A magnitude do espectro do filtro projetado pode ser observada na figura 26.

Após isso, o sinal de respiração foi filtrado utilizando as rotinas filter (causal) e a filtfilt (não causal). É possível observar os dois sinais resultantes nas figuras 28 e 29, respectivamente. Do ponto de vista da forma de onda dos sinais filtrados, percebe-se que eles são praticamente iguais. O que realmente faz a diferença nesses casos é o atraso que é introduzido pelo filtro causal.

Por fim, na figura 30 é possível ver o sinal sem nenhum tipo de ruído, fornecido pela questão. Como pode ser visto, o sinal sem ruído e os filtrados são muito parecidos em quesito da sua forma de onda, tendo em vista que os filtros conseguiram remover grande parte do ruído.

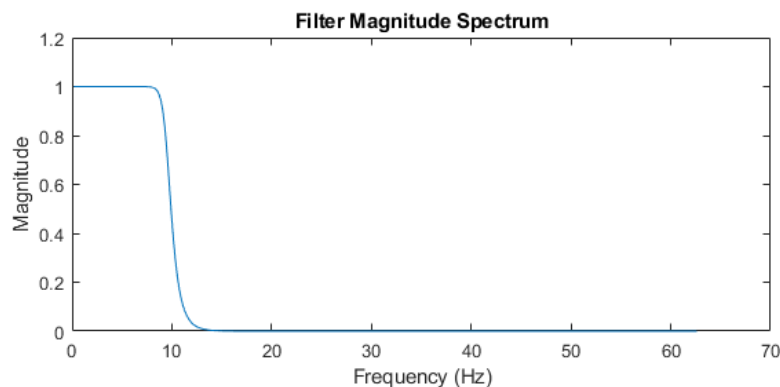


Figura 26:

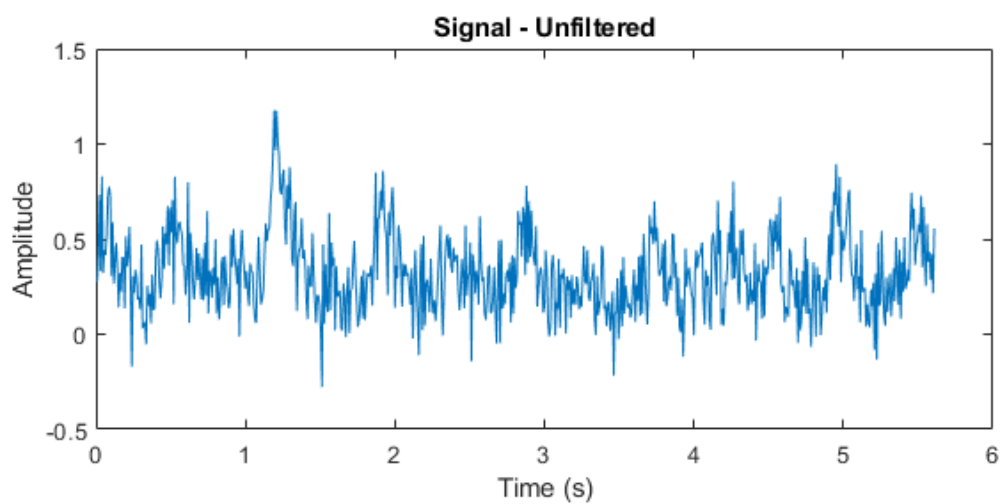


Figura 27:

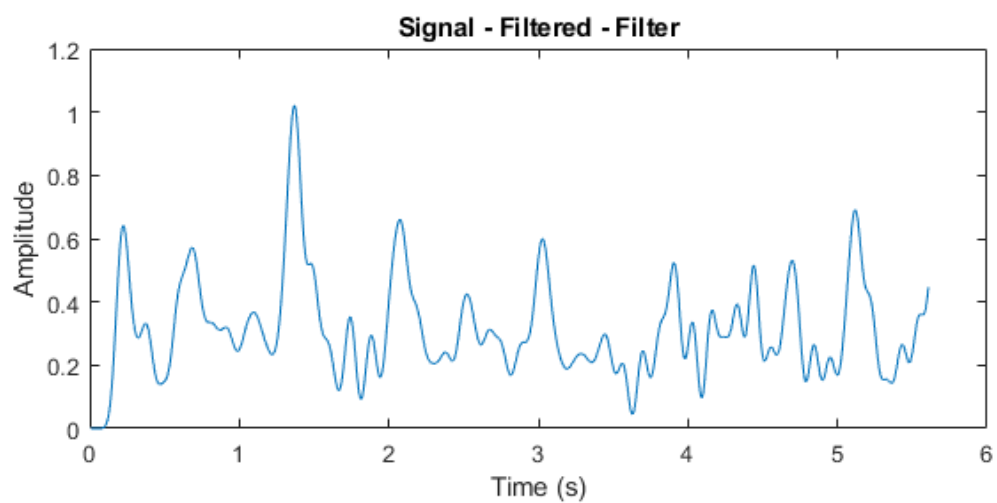


Figura 28:

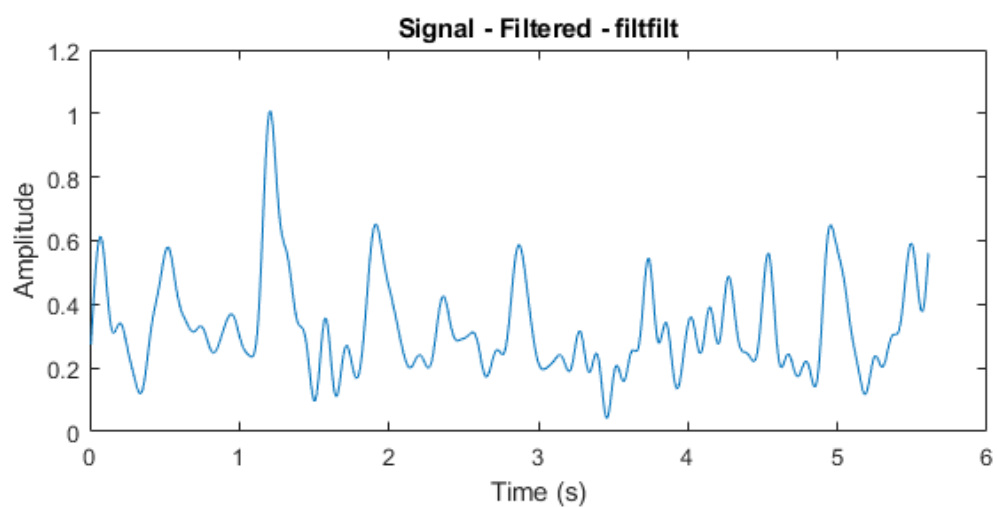


Figura 29:

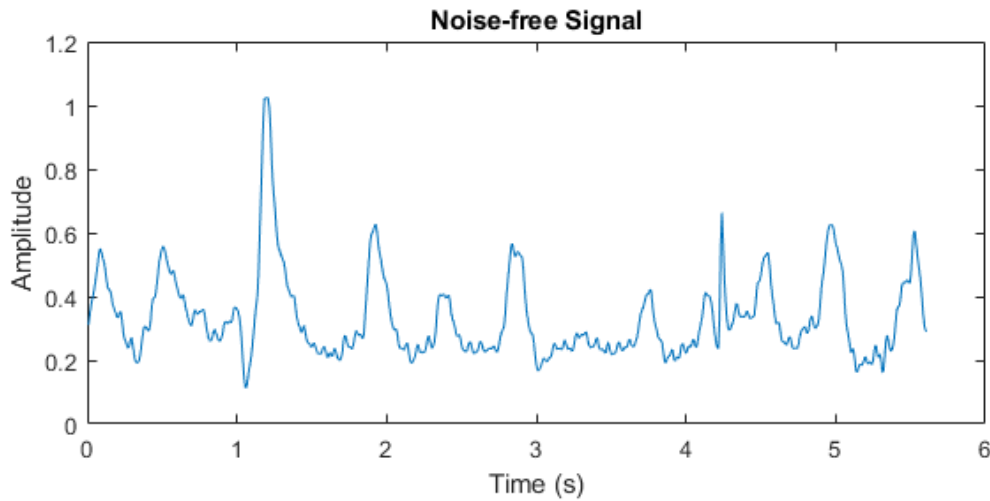


Figura 30:

```
close all;
clear all;

load Resp_noise1

fs = 125; % Frequency Sample
x = resp_noise1; % Respiration noisy signal
x_no_noise = resp; % Noise-free signal
N = length(x); % Total points
t = (1:N)/fs; % Time vector
fc = 0.15; % Relative cutoff frequency
L = 14; % Filter order
[b, a] = butter(L, fc); % Lowpass butterworth filter
H_mag = abs(fft(b, N)./fft(a, N)); % Filter Frequency Response Magnitude
f = (1:N)*fs/N; % Frequency vector
y_f = filter(b, a, x); % Filtered signal FILTER
y_filt = filtfilt(b, a, x); % Filtered signal FILTFILT

figure(1); %Window 1
plot(f(1:round(N/2)), H_mag(1:round(N/2))); % Plot magnitude spectrum
title("Filter Magnitude Spectrum"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Magnitude"); % Y axis label

figure(2); % Window 2
plot(t, x); % Plot signal - unfiltered
title("Signal - Unfiltered"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label

figure(3); % Window 3
plot(t, y_f); % Plot signal - filter
title("Signal - Filtered - Filter"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label

figure(4); % Window 4
plot(t, y_filt); % Plot signal - filtfilt
title("Signal - Filtered - filtfilt"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label

figure(5); % Window 5
plot(t, x_no_noise); % Plot noise-free signal
title("Noise-free Signal"); % Graph's title
```

```
xlabel("Time (s)"); % X axis label  
ylabel("Amplitude"); % Y axis label
```

Questão 4.25

Nessa questão, foi projetado um filtro de ordem 8 do tipo IIR Butterworth e uma frequência de corte de 100 Hz. Depois disso, foi gerada uma função impulso que consiste em 256 valores, sendo o primeiro igual a 1 e os demais iguais a 0. Em seguida, aplicou-se o filtro causal, figura 31, e o não causal, figura 32. Como pode ser observado, o filtro não causal acaba truncando alguns valores do sinal filtrado no domínio do tempo. Em seguida, foram obtidos os espectros de magnitude e fase dos sinais filtrados com a rotina filter, figura 33, e com a rotina filtfilt, figura 34.

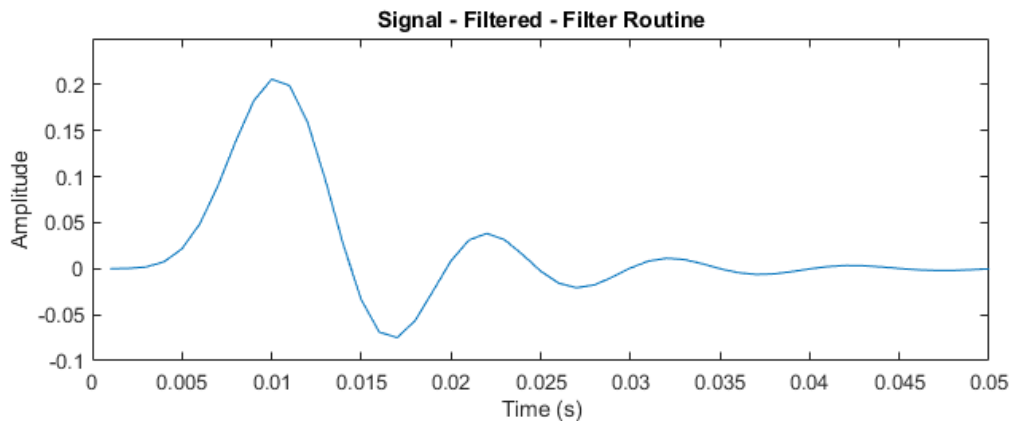


Figura 31:

Pode-se observar que o espectro gerado a partir do sinal filtrado com a rotina filtfilt apresenta um espectro um tanto quanto estranho. Isso se deve ao fato de que parte do impulso foi truncado. Para comprovar esse comportamento, foi gerado um impulso com um atraso de 10 amostras e obteve-se o espectro de magnitude e fases, figuras 35 e 36. Dessa vez, é possível observar que os espectros de magnitude são mais parecidos, no entanto, ainda assim eles apresentam uma leve diferença evidenciada pelo ripple presente na banda de passagem do filtro não causal. Uma desvantagem do filtro não causal é a possibilidade da geração de artefatos devido ao truncamento da porção inicial do sinal de entrada.

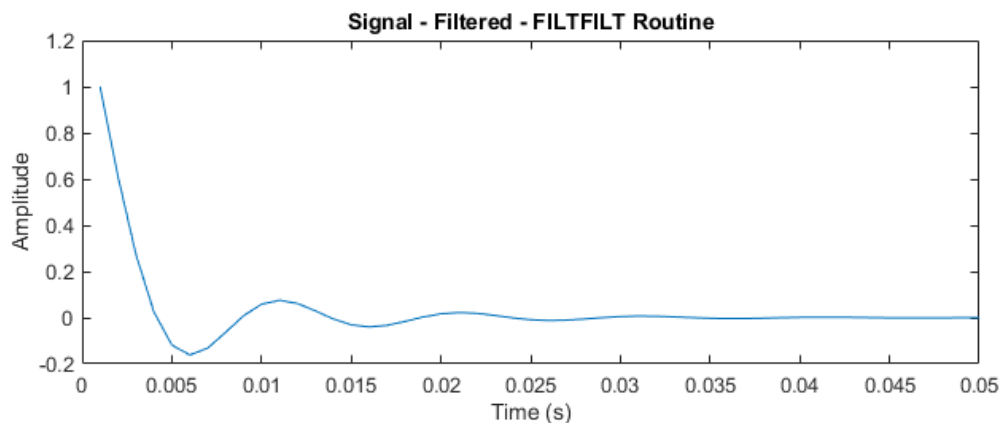


Figura 32:

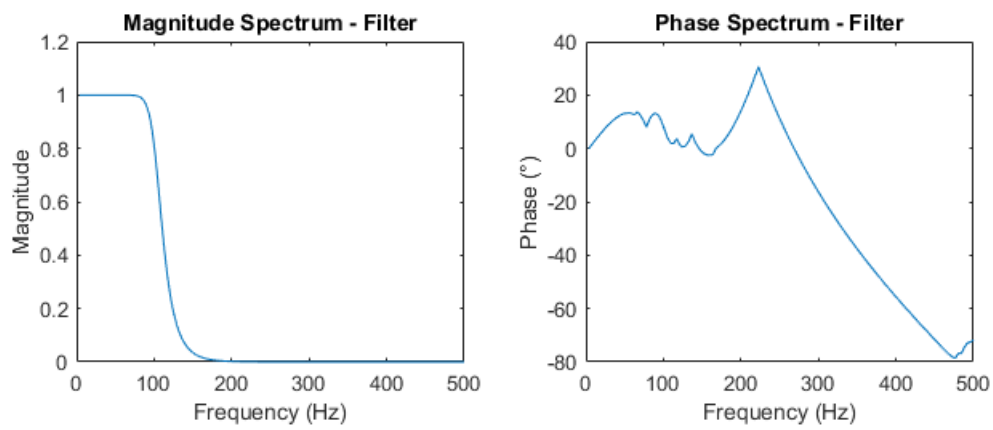


Figura 33:

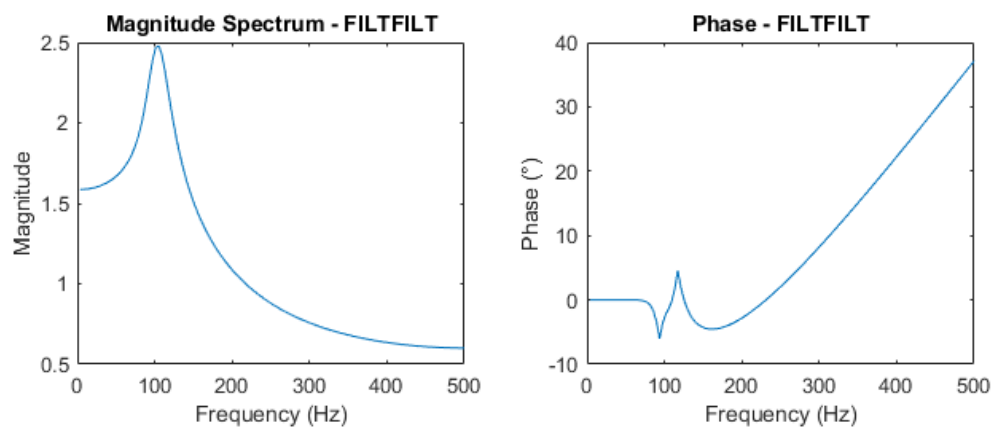


Figura 34:

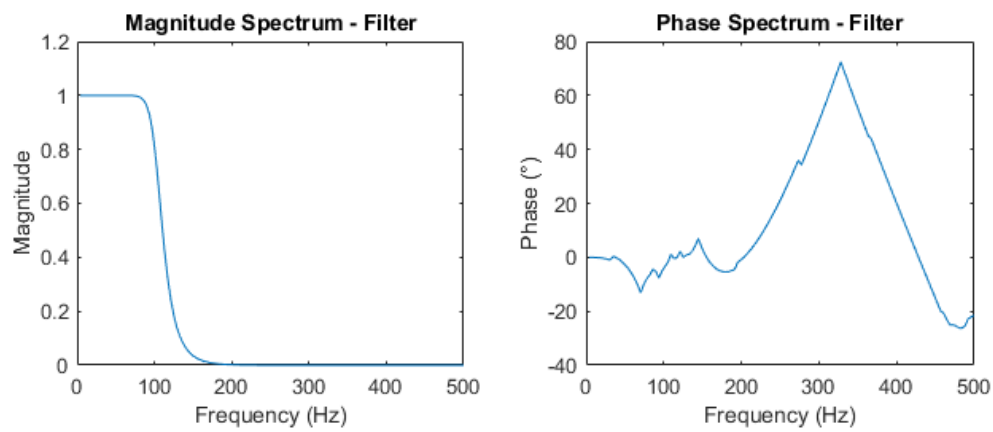


Figura 35:

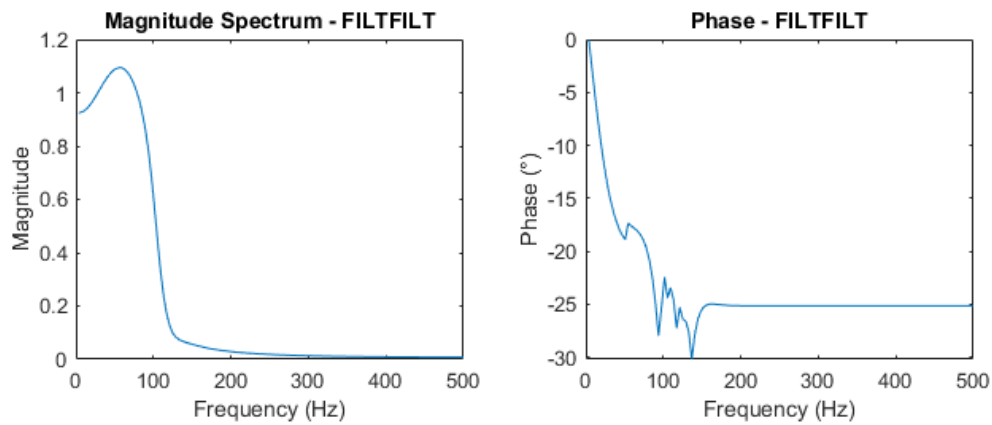


Figura 36:

```

close all;
clear all;

fs = 1e3; % Sample Frequency
%Lowpass Filter
N = 256; % Pad to a total of N points
cutoff = 100; % Cutoff frequency
fc = cutoff/(fs/2); % Relative cutoff frequency
f = (1:N) * fs/N; % Frequency vector
t = (1:N) / fs; % Time vector
L = 8; % Filter size
[b, a] = butter(L, fc); % IIR Butterworth lowpass

x = [1 zeros(1, N-1)]; % Impulse
x_delay = [zeros(1, 10) 1 zeros(1, 245)]; % Delayed Impulse
x_vector = [x; x_delay]; % Line 1: Impulse (No delay), Line 2: Impulse (Delay)

x_filtered = zeros(2,N); % Initialize vector
x_filtfilt = zeros(2,N); % Initialize vector

for idx=1:2
    x_filtered(idx,:) = filter(b, a, x_vector(idx,:)); % Filtered signal (filter)
    x_filtfilt(idx,:) = filtfilt(b, a, x_vector(idx,:)); % Filtered signal (filtfilt)
end
figure(1); % Window 1
plot(t, x_filtered(1,:)); % Plot signal - filtered (causally)
title("Signal - Filtered - Filter Routine"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label
xlim([0 0.05]); % X axis scale

figure(2); % Window 2
plot(t, x_filtfilt(1,:)); % Plot signal - filtered (non-causally)
title("Signal - Filtered - FILTFILT Routine"); % Graph's title
xlabel("Time (s)"); % X axis label
ylabel("Amplitude"); % Y axis label
xlim([0 0.05]); % X axis scale

for idx=1:2

    X_filtered = fft(x_filtered(idx,:)); % FFT of impulse response (causal)
    X_conv = fft(x_filtfilt(idx,:)); % FFT of impulse response (non-causal)
    X_filtered_mag = abs(X_filtered); % Magnitude of impulse response (causal)
    X_conv_mag = abs(X_conv); % Magnitude of impulse response (non-causal)
    X_filtered_phase = unwrap(angle(X_filtered)*180/pi); % Phase of impulse spectrum
    (causal)
    X_conv_phase = unwrap(angle(X_conv)*180/pi); % Phase of impulse spectrum (non-

```

```

causal)

figure, % Window 3
subplot(1,2,1); % Plot at the same window
plot(f(1:N/2), X_filtered_mag(1:N/2)); % Plot Magnitude Spectrum - Causal
title("Magnitude Spectrum - Filter"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Magnitude"); % Y axis label
subplot(1,2,2); % Plot at the same window
plot(f(1:N/2), X_filtered_phase(1:N/2)); % Plot Phase Spectrum - Causal
title("Phase Spectrum - Filter"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Phase (°)"); % Y axis label

figure, % Window 4
subplot(1,2,1); % Plot at the same window
plot(f(1:N/2), X_conv_mag(1:N/2)); % Plot Magnitude Spectrum - Non-Causal
title("Magnitude Spectrum - FILTFILT"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Magnitude"); % Y axis label
subplot(1,2,2); % Plot at the same window
plot(f(1:N/2), X_conv_phase(1:N/2)); % Plot Phase Spectrum - Non-Causal
title("Phase - FILTFILT"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Phase (°)"); % Y axis label
end

```

Questão 4.27

Essa questão busca demonstrar a diferença entre filtros FIR e IIR no sentido de observar a inclinação da reta. Para isso, foi projetado um filtro de ordem 12 do tipo Butterworth (passa-altas) com a frequência de 80 Hz. Na construção das funções, por se utilizar as rotinas do MATLAB, foi necessário utilizar a frequência relativa a $\frac{f_s}{2}$. O outro filtro utilizado é do tipo FIR e a sua ordem foi determinada experimentalmente de forma a obter uma inclinação similar à do filtro IIR de ordem 12. Como pode ser observado pela figura 37, para ordem 65 do filtro FIR, as inclinações dos filtros foram similares. Isso mostra que para obter a inclinação de um filtro IIR com um filtro FIR, faz-se necessário utilizar um filtro de ordem muito maior.

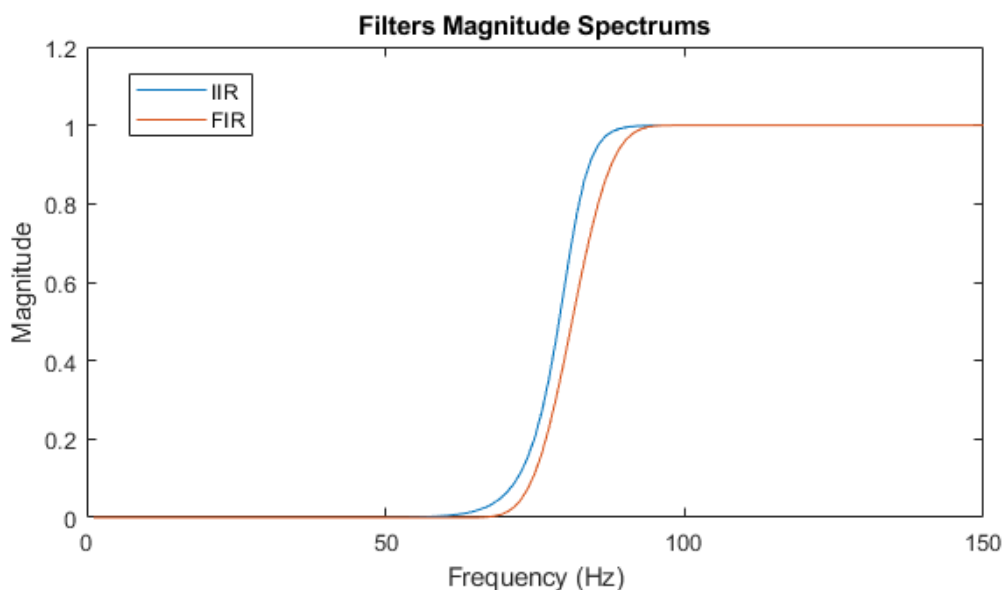


Figura 37:

```

close all;
clear all;

```

```

fs = 300; % Frequency Sample
N = 256; % Total points
t = (1:N)/fs; % Time vector
f = (1:N)*fs/N; % Frequency vector
cutoff = 80; % Cutoff Frequency
fc_iir = cutoff/(fs/2); % Relative cutoff frequency
L_iir = 12; % Filter order
[b, a] = butter(L_iir, fc_iir, 'high'); % Highpass butterworth filter

L_fir = 65; % Filter order
fc_fir = cutoff/fs; % Relative cutoff frequency
k = -floor(L_fir/2):-1; % ngative values of k
b_fir = -sin(2*pi*fc_fir*k)./(pi*k); % Calculate b coeffs (negative half)
b_fir = [b_fir 1 - 2*fc_fir, fliplr(b_fir)]; % Mirror b coeffs and k = 0
b_fir_black = b_fir .* blackmanharris(L_fir)'; % Windowed b coeff

H_mag_iir = abs(fft(b, N)./fft(a, N)); % Filter Frequency Response Magnitude
H_mag_fir = abs(fft(b_fir_black, N)); % Filter Frequency Response Magnitude

figure(1); %Window 1
plot(f(1:round(N/2)), H_mag_iir(1:round(N/2))); % Plot magnitude spectrum
hold on; % Plot at the same graph
plot(f(1:round(N/2)), H_mag_fir(1:round(N/2))); % Plot magnitude spectrum
title("Filters Magnitude Spectrums"); % Graph's title
xlabel("Frequency (Hz)"); % X axis label
ylabel("Magnitute"); % Y axis label
legend("IIR", "FIR");

fprintf("b coeffs fir filter = %d\n", length(b_fir));
fprintf("b coeffs iir filter = %d\n", length(b), length(a))
;

```

Referências bibliográficas

- [1] Semmlow, J. L., & Griffel, B. (2014). Biosignal and Medical Image Processing (3rd ed.). CRC Press.