Chapter 2

Problem 2.1

1) To derive the Fourier series coefficients in the expansion of $x(t)$, we have

$$\begin{aligned} x_n &= \frac{1}{4}\int_{-1}^{1} e^{-j2\pi nt/4}\,dt \\ &= \frac{1}{-2j\pi n}\left[e^{-j2\pi n/4} - e^{j2\pi n/4}\right] \qquad (1) \\ &= \frac{1}{2}\,\mathrm{sinc}\left(\frac{n}{2}\right) \qquad (2) \end{aligned}$$

where $\mathrm{sinc}(x)$ is defined as

$$\mathrm{sinc}(x) = \frac{\sin(\pi x)}{\pi x} \qquad (3)$$

2) Obviously, all the x_n's are real (since $x(t)$ is real and even), so

$$\begin{cases} a_n = \mathrm{sinc}\left(\frac{n}{2}\right) \\ b_n = 0 \\ c_n = \left|\mathrm{sinc}\left(\frac{n}{2}\right)\right| \\ \theta_n = 0, \pi \end{cases} \qquad (4)$$

Note that for even n's, $x_n = 0$ (with the exception of $n = 0$, where $a_0 = c_0 = 1$ and $x_0 = \frac{1}{2}$). Using these coefficients, we have

$$\begin{aligned} x(t) &= \sum_{n=-\infty}^{\infty} \frac{1}{2}\,\mathrm{sinc}\left(\frac{n}{2}\right) e^{j2\pi nt/4} \\ &= \frac{1}{2} + \sum_{n=1}^{\infty} \mathrm{sinc}\left(\frac{n}{2}\right)\cos\left(2\pi t\frac{n}{4}\right) \qquad (5) \end{aligned}$$

A plot of the Fourier series approximations to this signal over one period for $n = 0, 1, 3, 5, 7, 9$ is shown in Figure 1.

3) Note that x_n is always real. Therefore, depending on its sign, the phase is either zero or π. The magnitude of the x_n's is $\frac{1}{2}\left|\mathrm{sinc}\left(\frac{n}{2}\right)\right|$. The discrete and phase spectrum are shown in Figure 2.

Problem 2.2

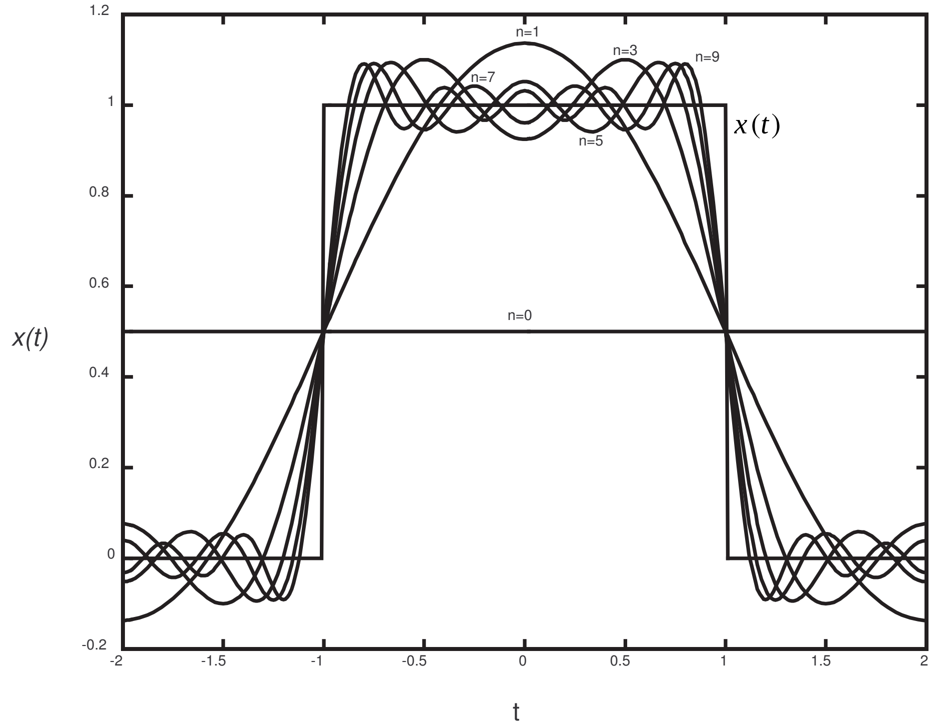

Figure 1: Various Fourier series approximations for the rectangular pulse in Computer Problem 1

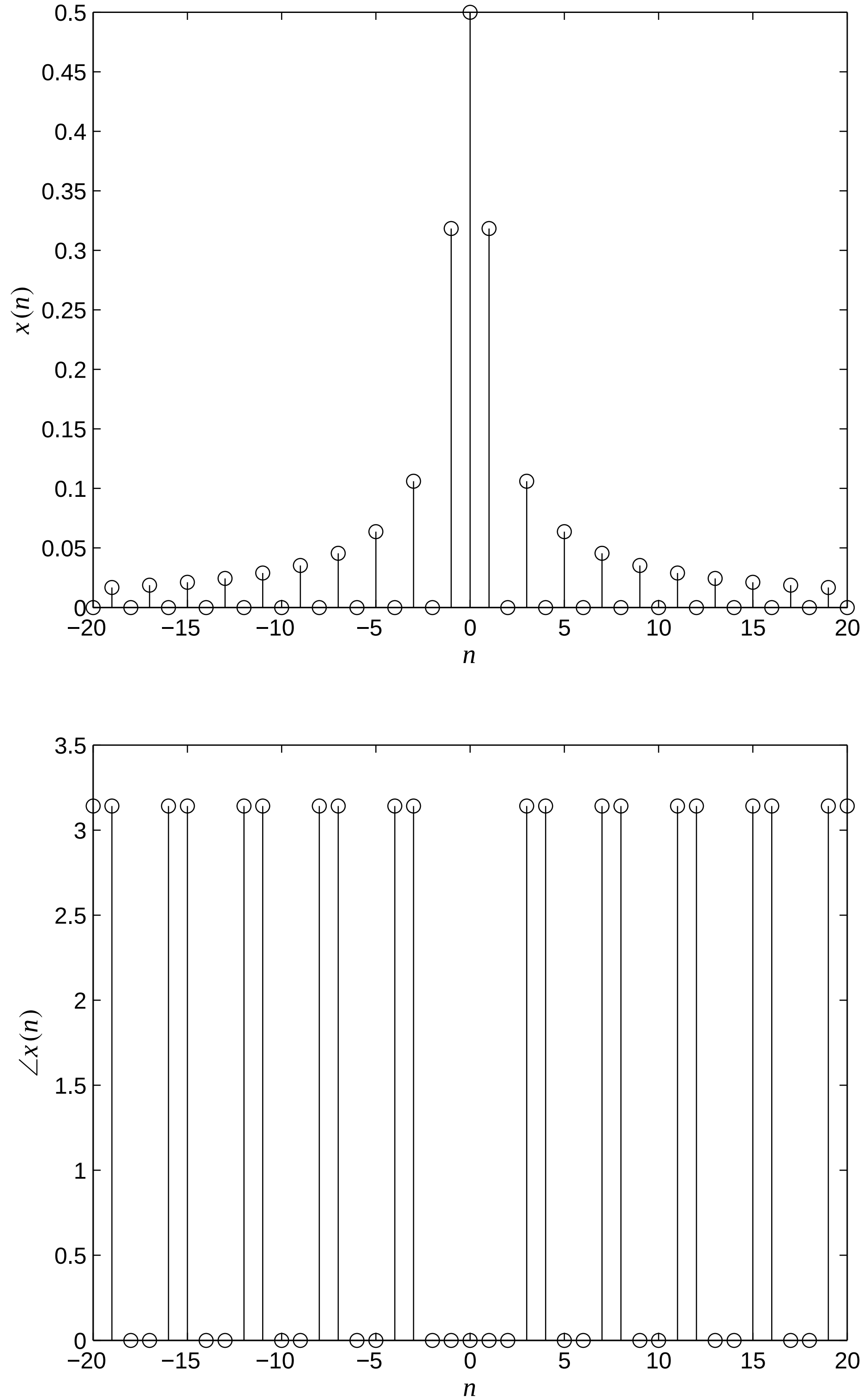

Figure 2: The discrete and phase spectrum of the signal in Computer Problem 1

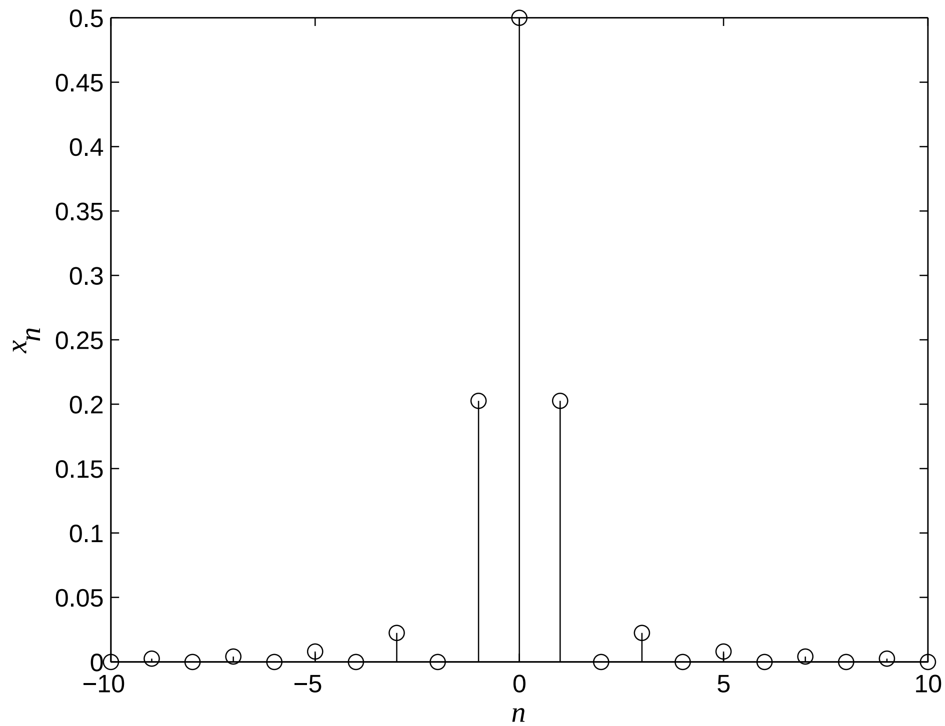

Figure 3: The discrete spectrum of the signal

1) We have

$$
\begin{aligned}
x_n &= \frac{1}{T_0}\int_{-T_0/2}^{T_0/2} x(t)e^{-j2\pi nt/T_0}dt && (6)\\
&= \frac{1}{2}\int_{-1}^{1}\Lambda(t)e^{-j\pi nt}dt && (7)\\
&= \frac{1}{2}\int_{-\infty}^{+\infty}\Lambda(t)e^{-j\pi nt}dt && (8)\\
&= \frac{1}{2}\mathcal{F}[\Lambda(t)]_{f=n/2} && (9)\\
&= \frac{1}{2}\,\mathrm{sinc}^2(\frac{n}{2}) && (10)\\
& && (11)
\end{aligned}
$$

where we have used the facts that $\Lambda(t)$ vanishes outside the $[-1, 1]$ interval and that the Fourier transform of $\Lambda(t)$ is $\mathrm{sinc}^2(f)$. This result can also be obtained by using the expression for $\Lambda(t)$ and integrating by parts. Obviously, we have $x_n = 0$ for all even values of n except for $n = 0$.
2) A plot of the discrete spectrum of $x(t)$ is presented in Figure 3
3) A plot of the discrete spectrum $\{y_n\}$ is presented in Figure 4
The MATLAB script for this problem is given next.

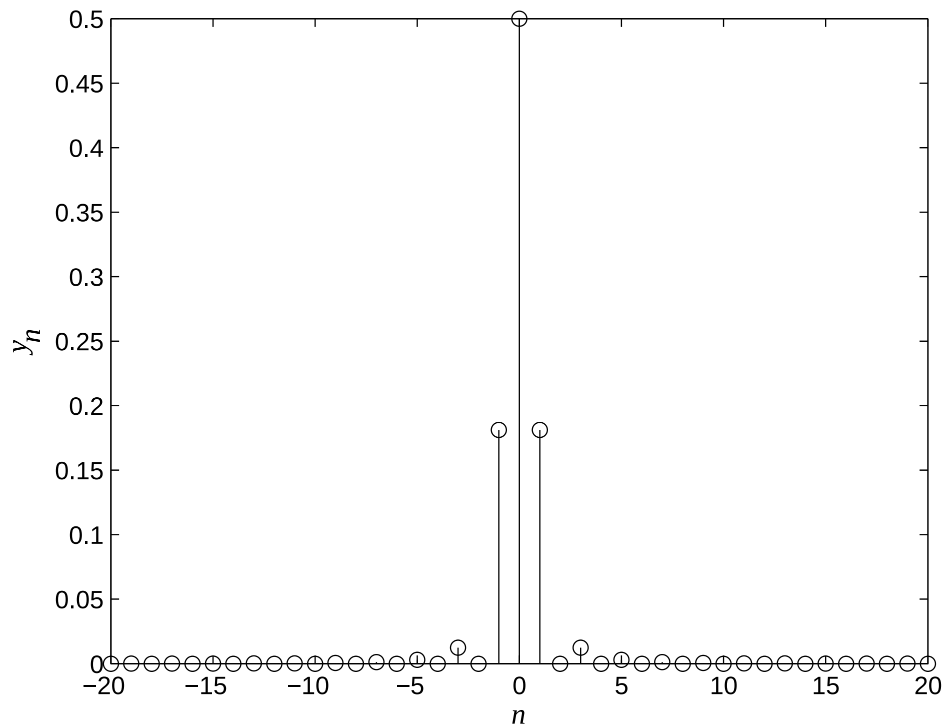

Figure 4: The discrete spectrum of the signal

```
% MATLAB script for Computer Problem 2.2.
echo on
n=[−20:1:20];
% Fourier series coefficients of x(t) vector
x=.5*(sinc(n/2)).^2;
% sampling interval
ts=1/40;
% time vector
t=[−.5:ts:1.5];
% impulse response
fs=1/ts;
h=[zeros(1,20),t(21:61),zeros(1,20)];
% transfer function
H=fft(h)/fs;
% frequency resolution
df=fs/80;
f=[0:df:fs]−fs/2;
% rearrange H
H1=fftshift(H);
y=x.*H1(21:61);
% Plotting commands follow.
```

Problem 2.3

The common magnitude spectrum is presented in Figure 5. The two phase spectrum of the two signals plotted on the same axes are given in Figure 6.
The MATLAB script for this problem follows.

```
% MATLAB script for Computer Problem 2.3.
df=0.01;
fs=10;
ts=1/fs;
t=[−5:ts:5];
x1=zeros(size(t));
x1(41:51)=t(41:51)+1;
x1(52:61)=ones(size(x1(52:61)));
x2=zeros(size(t));
x2(51:71)=x1(41:61);
[X1,x11,df1]=fftseq(x1,ts,df);
[X2,x21,df2]=fftseq(x2,ts,df);
X11=X1/fs;
X21=X2/fs;
f=[0:df1:df1*(length(x11)−1)]−fs/2;
plot(f,fftshift(abs(X11)))
figure
plot(f(500:525),fftshift(angle(X11(500:525))),f(500:525),fftshift(angle(X21(500:525))),'--')
```

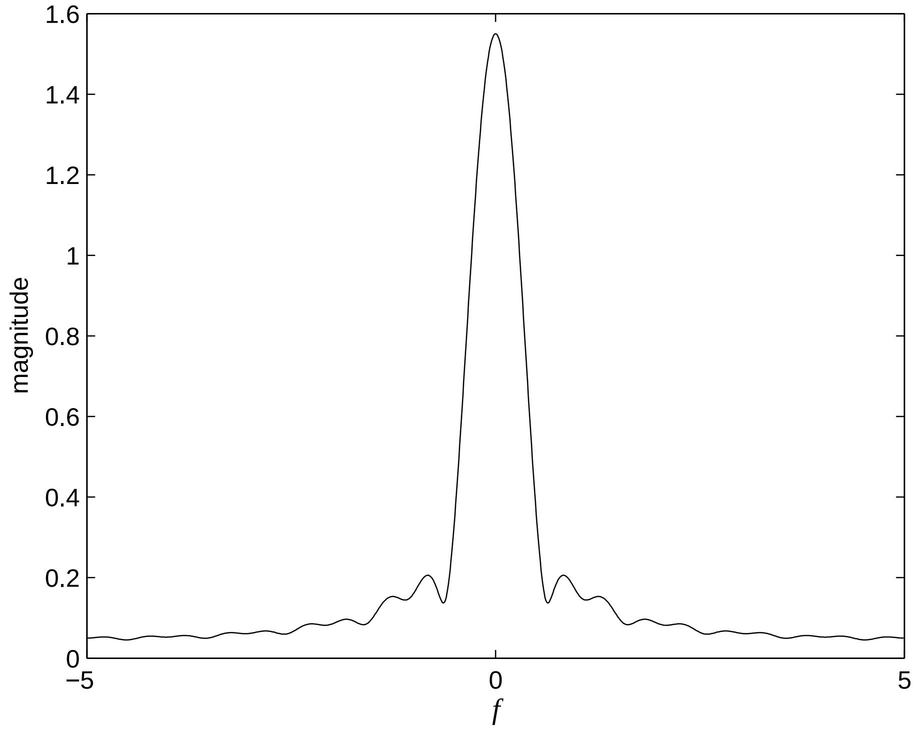

Figure 5: The common magnitude spectrum of the signals $x_1(t)$ and $x_2(t)$

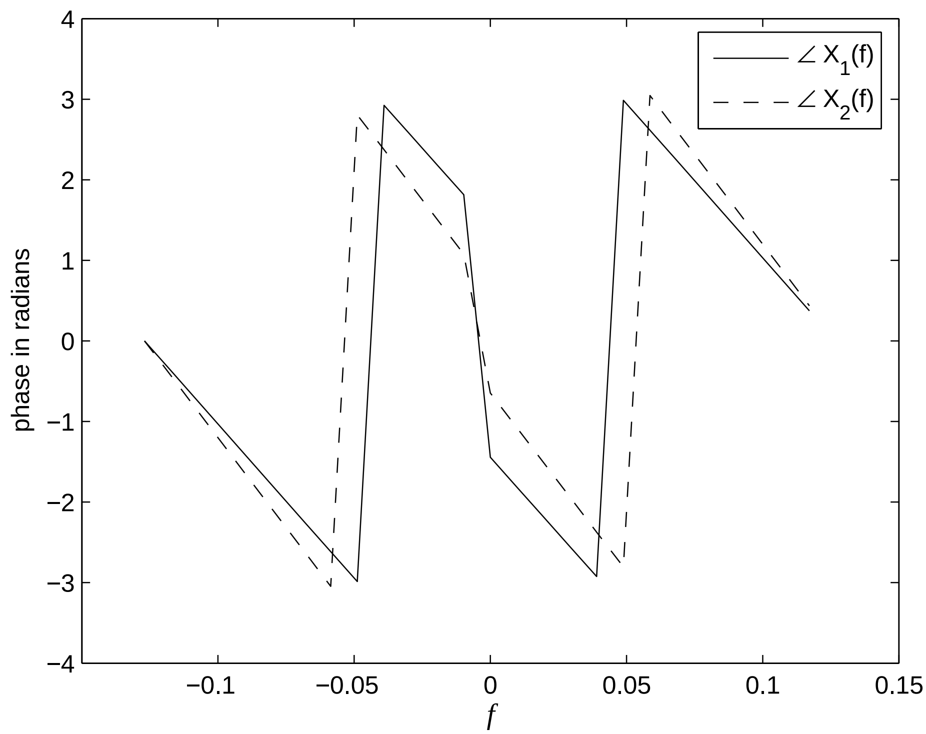

Figure 6: The phase spectrum of the signals $\Delta x_1(t)$ and $\Delta x_2(t)$

Problem 2.4
The Fourier transform of the signal $x(t)$ is

$$\frac{1}{1+j2\pi f}$$

Figures 7 and 8 present the magnitude and phase spectrum of the input signal $x(t)$.
2) The fourier transform of the output signal $y(t)$ is

$$y(f)=\begin{cases}\frac{1}{1+j2\pi f} & |f|\le 1.5\\ 0 & \text{otherwise}\end{cases}$$

The magnitude and phase spectrum of $y(t)$ is given in Figures 9 and 10, respectively.
3) The inverse Fourier transform of the output signal is parented in Figure 11
The MATLAB script for this problem is given next

```
% MATLAB script for Computer Problem 2.4.
df= 0.01;
f = -4:df:4;
x_f = 1./(1+2*pi*i*f);
plot(f, abs(x_f));
figure;
plot(f, angle(x_f));
indH = find(abs(f) <= 1.5);
H_f = zeros(1, length(x_f));
H_f(indH) = cos(pi*f(indH)./3);
y_f = x_f.*H_f;
figure;
plot(f,abs(y_f));
axis([-1.5 1.5 0 16]);
figure;
plot(f, angle(y_f));

y_f(401) = 10^30;
y_t = ifft(y_f, 'symmetric');
figure;
plot(y_t)
```

Problem 2.5
Choosing the sampling interval to be $t_s = 0.001$ s, we have a sampling frequency of $f_s = 1/t_s = 1000$ Hz. Choosing a desired frequency resolution of $df = 0.5$ Hz, we have the following.
1) Plots of the signal and its magnitude spectrum are given in Figures 12 and 13, respectively. Plots are generated by Matlab.
2) Choosing $f_0 = 200$ Hz, we find the lowpass equivalent to $x(t)$ by using the loweq.m function. Then using fftseq.m, we obtain its spectrum; we plot its magnitude spectrum in Figure 14. The MATLAb functions loweq.m and fftseq.m are given next.

```
function [M,m,df]=fftseq(m,ts,df)
%         [M,m,df]=fftseq(m,ts,df)
%         [M,m,df]=fftseq(m,ts)
%FFTSEQ   generates M, the FFT of the sequence m.
```

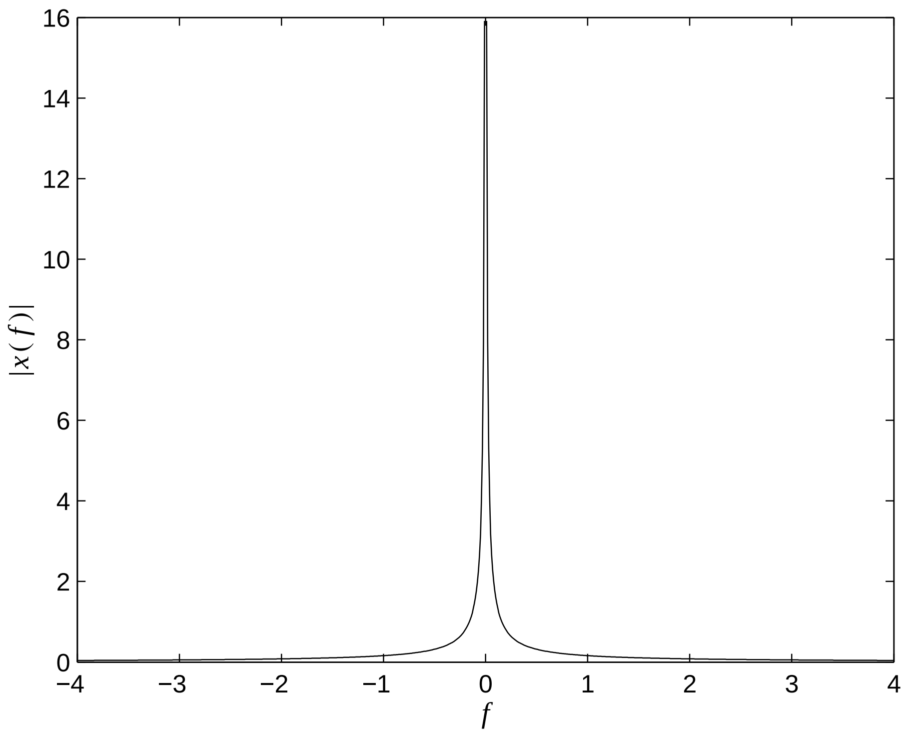

Figure 7: Magnitude spectrum of $x(t)$

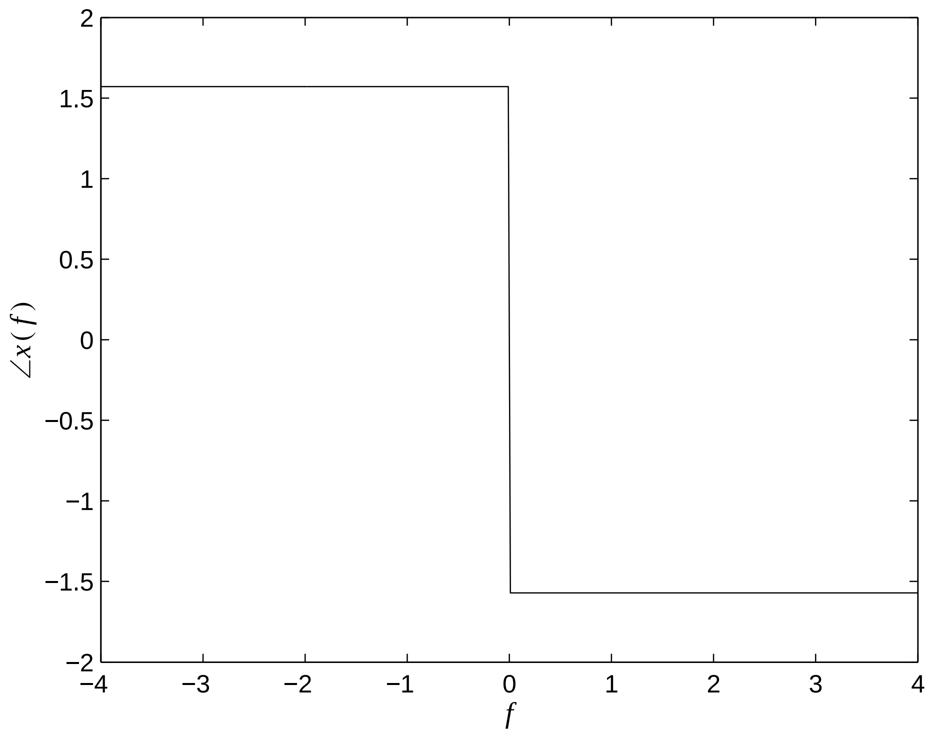

Figure 8: Phase spectrum of $x(t)$

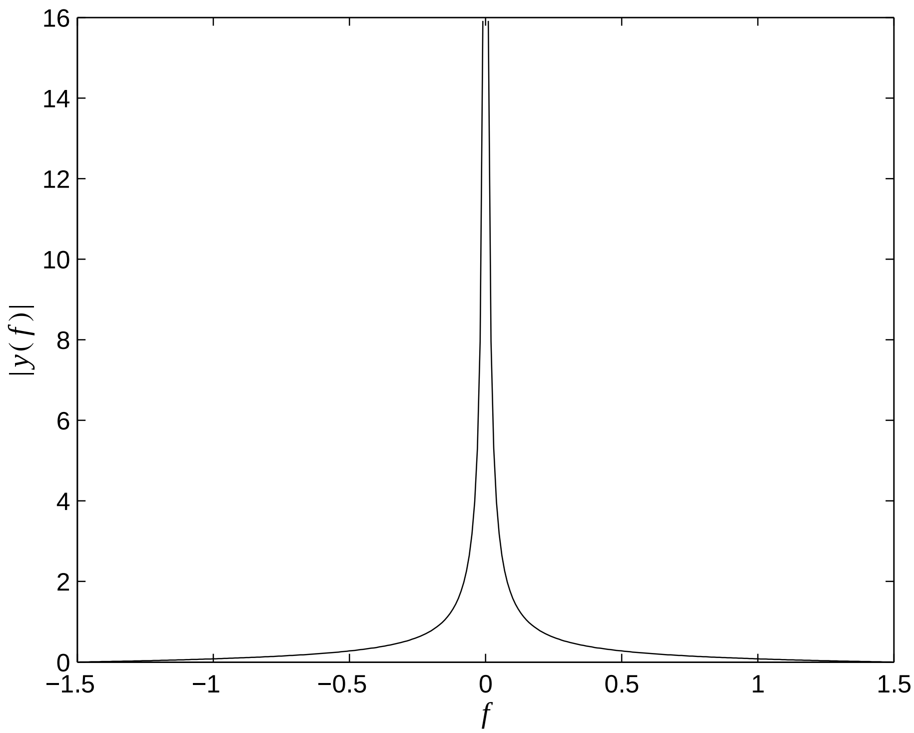

Figure 9: Magnitude spectrum of $y(t)$

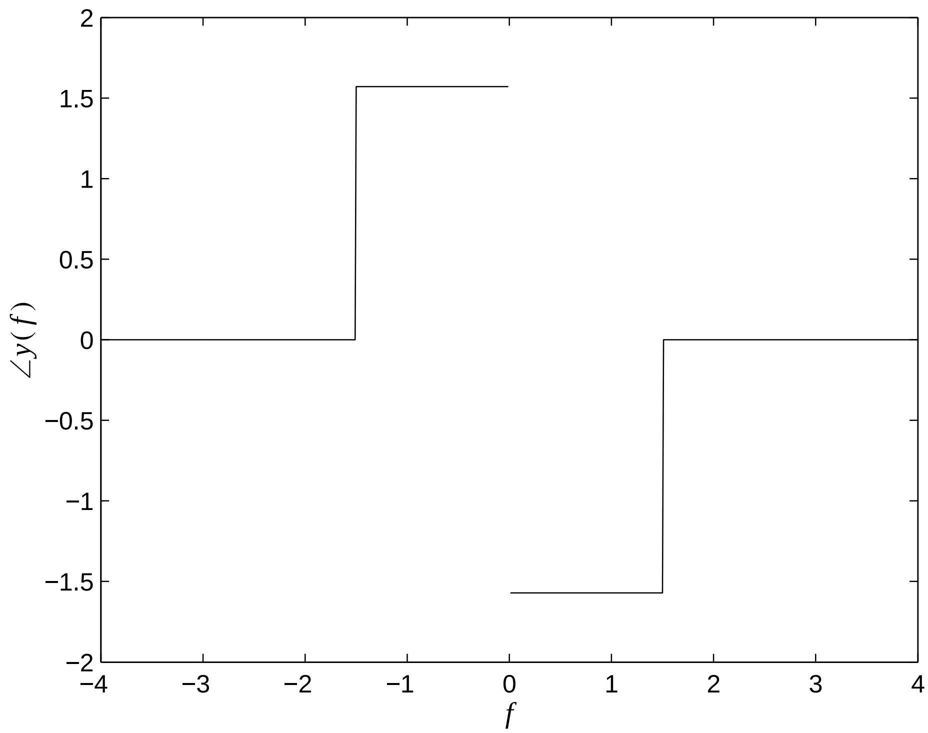

Figure 10: Phase spectrum of $y(t)$

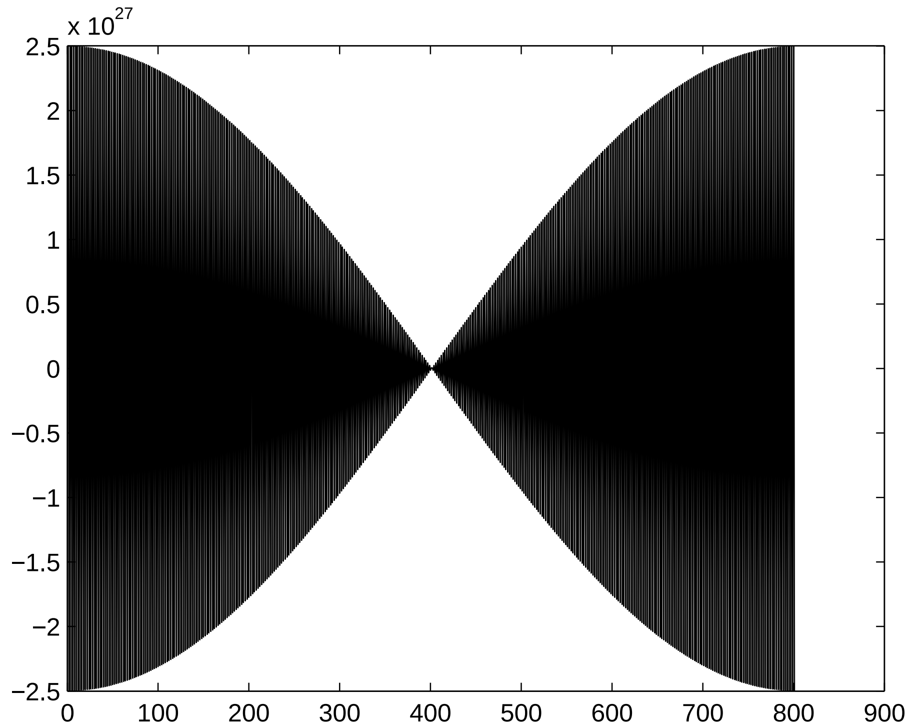

Figure 11: Inverse Fourier transform

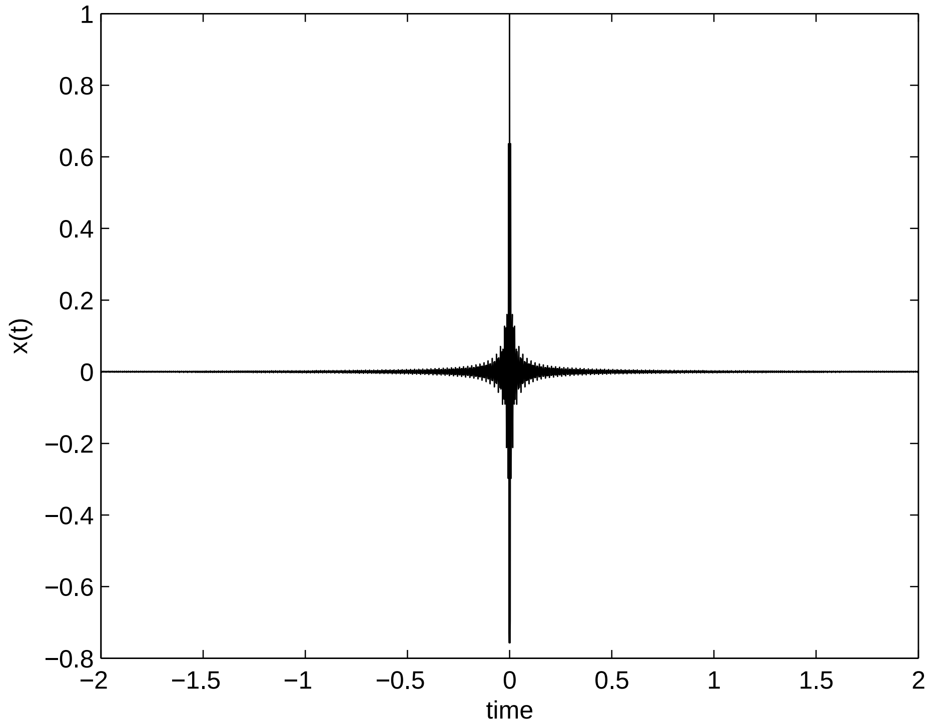

Figure 12: The signal $x(t)$

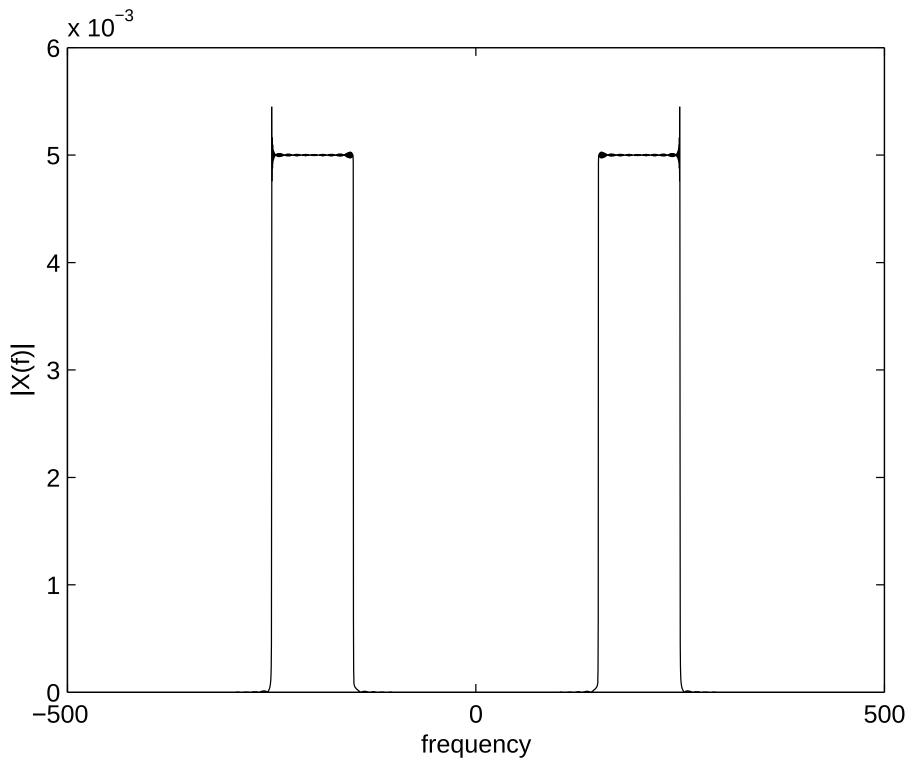

Figure 13: The magnitude spectrum of $x(t)$

```
%            The sequence is zero-padded to meet the required frequency resolution df.
%            ts is the sampling interval. The output df is the final frequency resolution.
%            Output m is the zero-padded version of input m. M is the FFT.
fs=1/ts;
if nargin == 2
  n1=0;
else
  n1=fs/df;
end
n2=length(m);
n=2^(max(nextpow2(n1),nextpow2(n2)));
M=fft(m,n);
m=[m,zeros(1,n-n2)];
df=fs/n;
```

```
function xl=loweq(x,ts,f0)
%            xl=loweq(x,ts,f0)
%LOWEQ       returns the lowpass equivalent of the signal x
%            f0 is the center frequency.
%            ts is the sampling interval.
%
t=[0:ts:ts*(length(x)-1)];
z=hilbert(x);
xl=z.*exp(-j*2*pi*f0*t);
```

It is seen that the magnitude spectrum is an even function in this case because we can write

$$x(t) = Re[\mathrm{sinc}(100t)e^{j\times 400\pi t}] \tag{12}$$

Comparing this to

$$x(t) = Re[x_l(t)e^{j2\pi\times f_0 t}] \tag{13}$$

we conclude that

$$x_l(t) = sinc(100t) \tag{14}$$

which means that the lowpass equivalent signal is a real signal in this case. This, in turn, means that $x_c(t) = x_l(t)$ and $x_s(t) = 0$. Also, we conclude that

$$\begin{cases} V(t) &= |x_c(t)| \\ \Theta &= \begin{cases} 0, & x_c(t) \geq 0 \\ \pi, & x_c(t) < 0 \end{cases} \end{cases} \tag{15}$$

Plots of $x_c(t)$ and $V(t)$ are given in Figures 15 and 16, respectively. Note that choosing f_0 to be the frequency with respect to which $X(f)$ is symmetric result in these figures.

Problem 2.6

The Remez algorithm requires that we specify the length of the FIR filter M, the passband edge frequency f_p, the stopband edge frequency f_s, and the ratio δ_2/δ_1. Here, δ_1 and δ_2 denote passband and stopband ripples, respectively. The filter length M can be approximated using

$$\hat{M} = \frac{-20\log_{10}\sqrt{\delta_1\delta_2} - 13}{14.6\Delta f} + 1$$

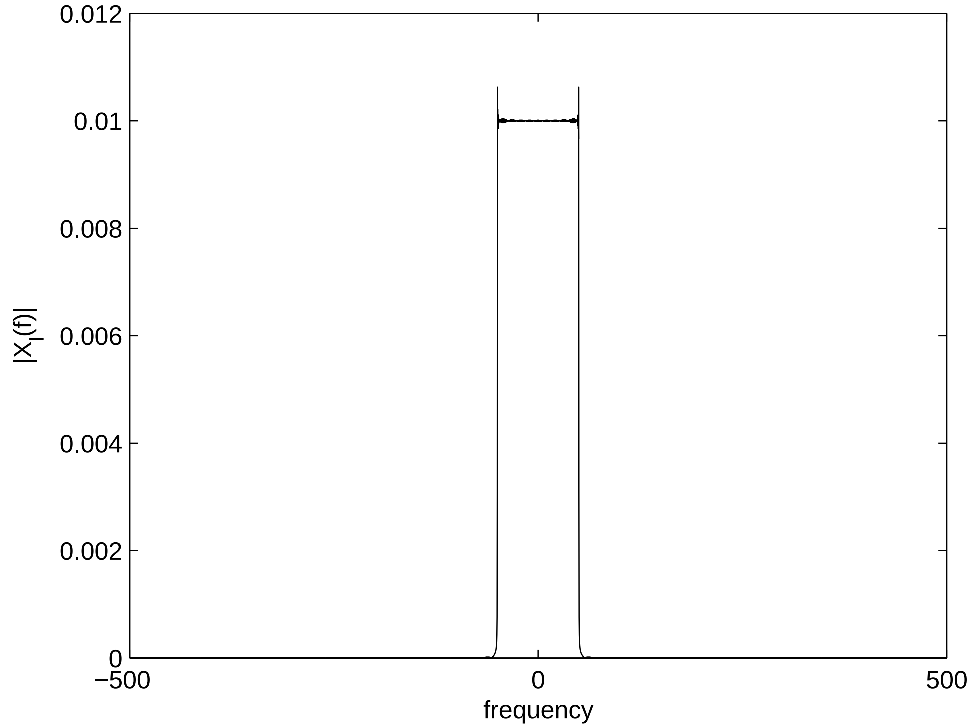

Figure 14: The magnitude spectrum of $x_l(t)$

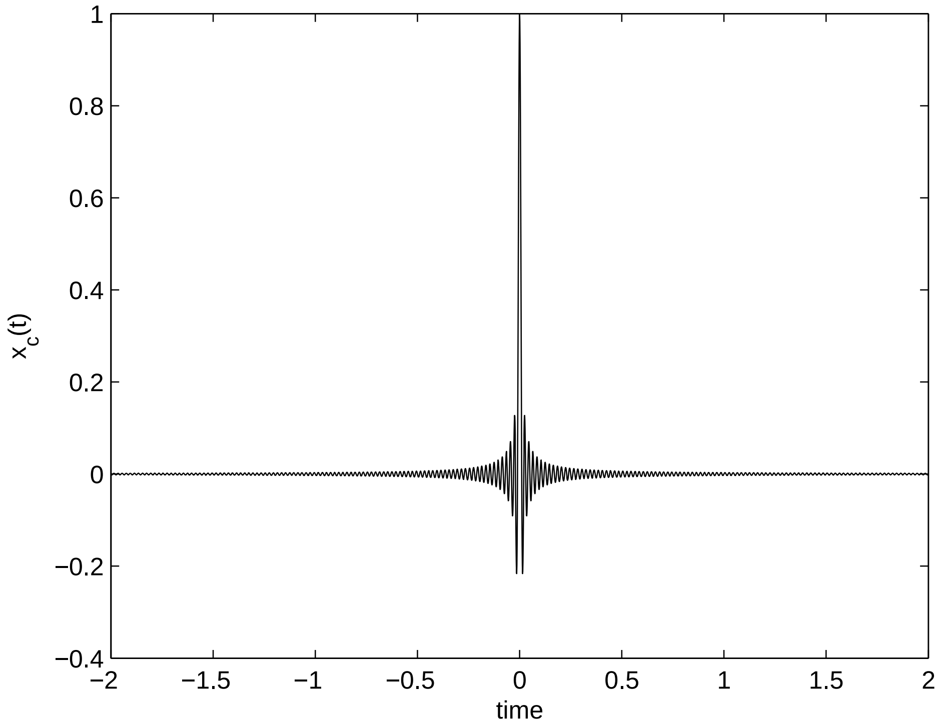

Figure 15: The signal $x_C(t)$

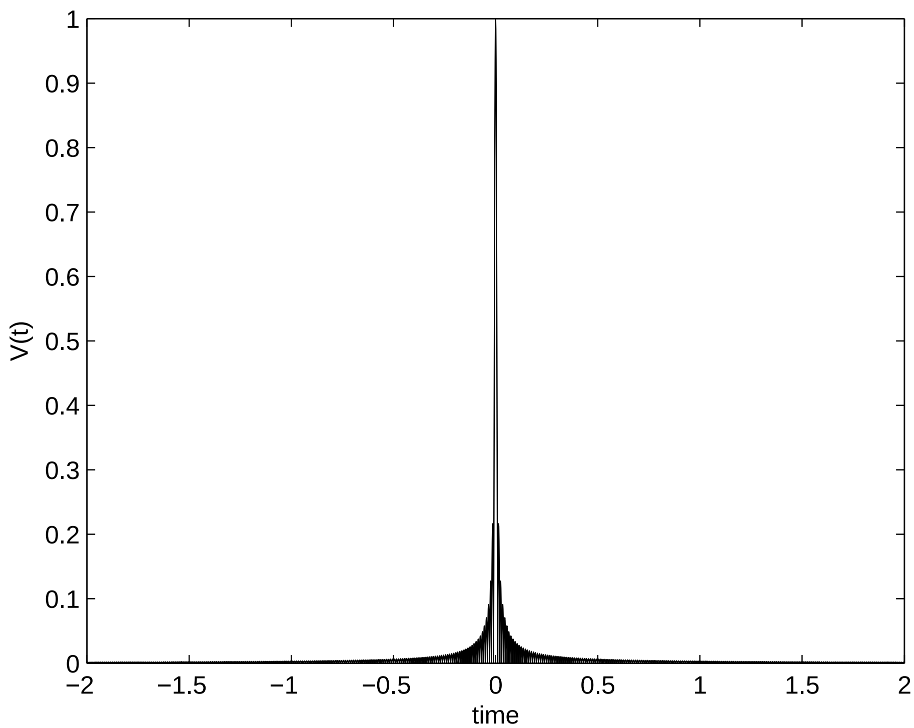

Figure 16: The signal $V(t)$

where Δf is the transition bandwidth $\Delta f = f_s - f_p$
1) Figure 17 shows the impulse response coefficients of the FIR filter.
2) Figures 18 and 19 show the magnitude and phase of the frequency response of the filter, respectively.
The MATLAB script for this problem is given next

```
% MATLAB script for Computer Problem 2.6.
fp = 0.4;
fs = 0.5;
df = fs-fp;
Rp = 0.5;
As = 40;
delta1=(10^(Rp/20)-1)/(10^(Rp/20)+1);
delta2=(1+delta1)*(10^(-As/20));
%Calculate approximate filter length
Mhat=ceil((-20*log10(sqrt(delta1*delta2))-13)/(14.6*df)+1);
f=[0 fp fs 1];
m=[1 1 0 0];
w=[delta2/delta1 1];
h=remez(Mhat+20,f,m,w);
[H,W]=freqz(h,[1],3000);
db = 20*log10(abs(H));
% plot results
stem(h);
figure;
plot(W/pi, db)
figure;
plot(W/pi, angle(H));
```

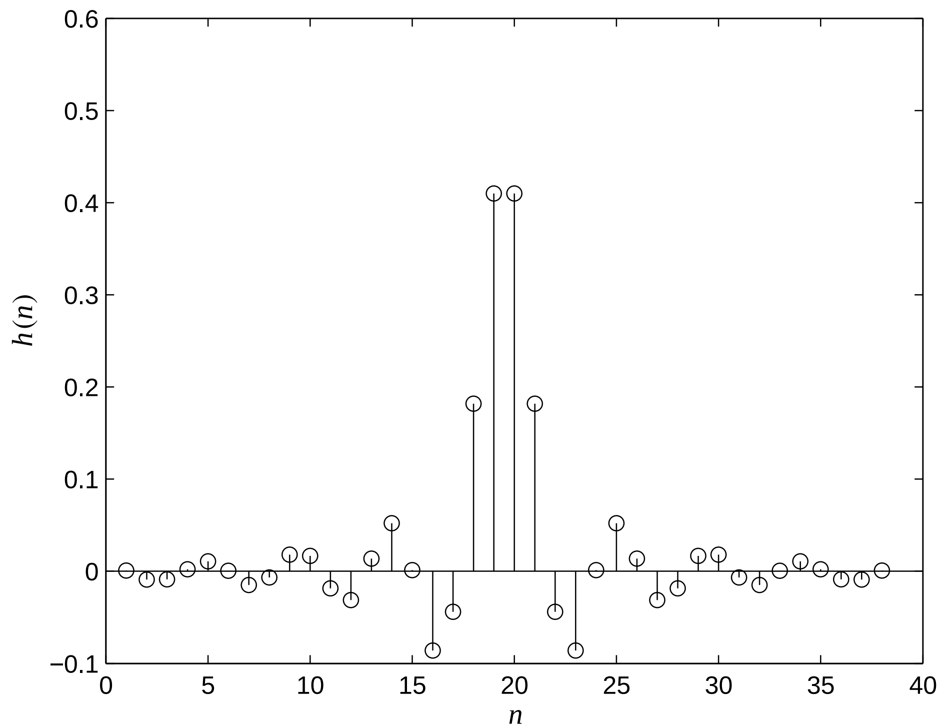

Figure 17: Impulse response coefficients of the FIR filter

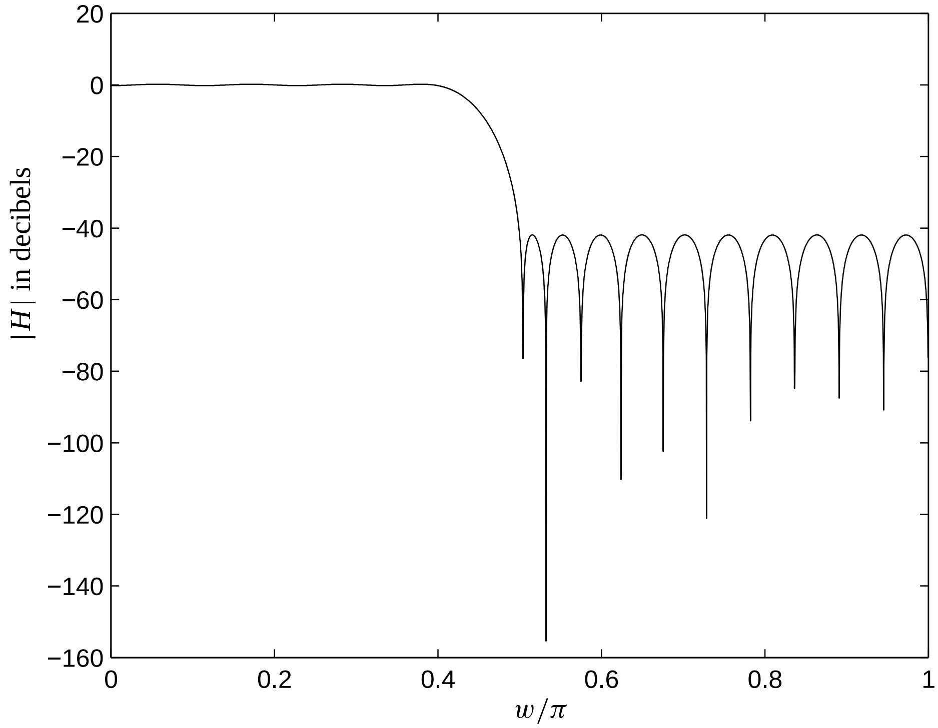

Figure 18: Magnitude of the frequency response of the FIR filter

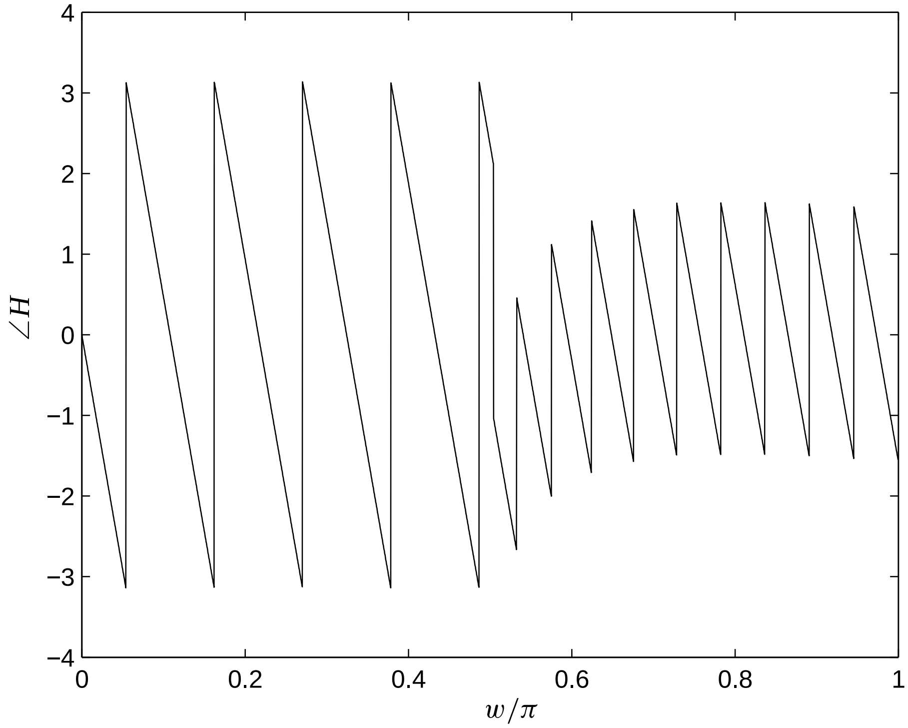

Figure 19: Phase of the frequency response of the FIR filter

Problem 2.7
1) The impulse response coefficients of the filter is presented in Figure 20.
2) The magnitude of the frequency response of the filter is given in Figure 21.
The MATLAB script for this problem is given next

```
% MATLAB script for Computer Problem 2.7.
f=[0 0.01 0.1 0.5 0.6 1];
m=[0 0 1 1   0  0];
delta1 = 0.01;
delta2 = 0.01;
df = 0.1 − 0.01;
Mhat=ceil((−20*log10(sqrt(delta1*delta2))−13)/(14.6*df)+1);
w=[1 delta2/delta1 1];
h=remez(Mhat+20,f,m,w,'hilbert');

[H,W]=freqz(h,[1],3000);
db = 20*log10(abs(H));
% plot results
stem(h);
figure;
plot(W/pi, db)
figure;
```

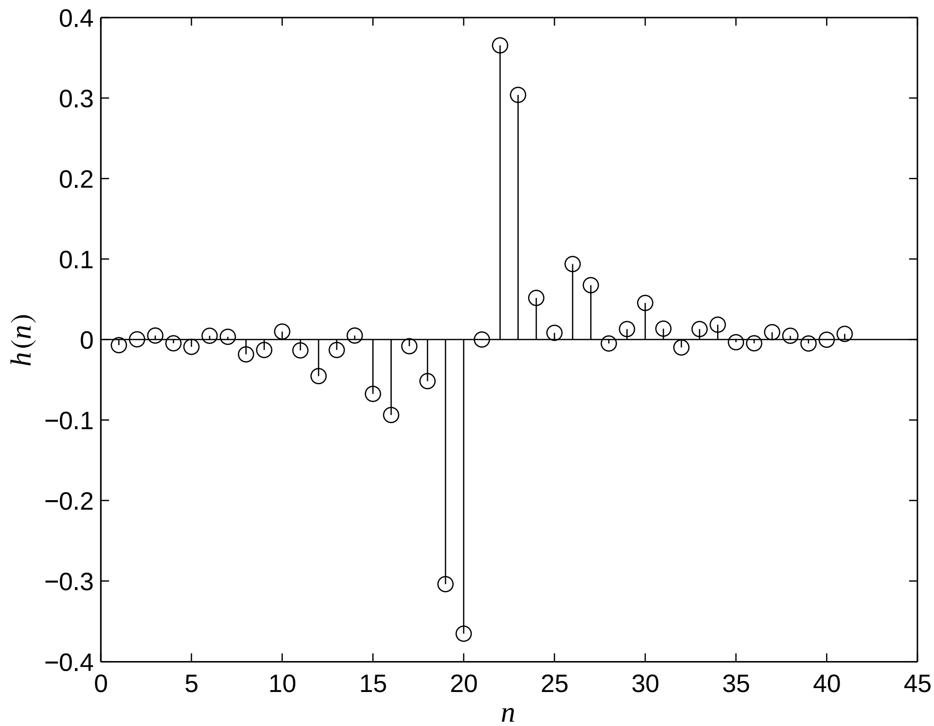

Figure 20: The impulse response coefficients of the filter

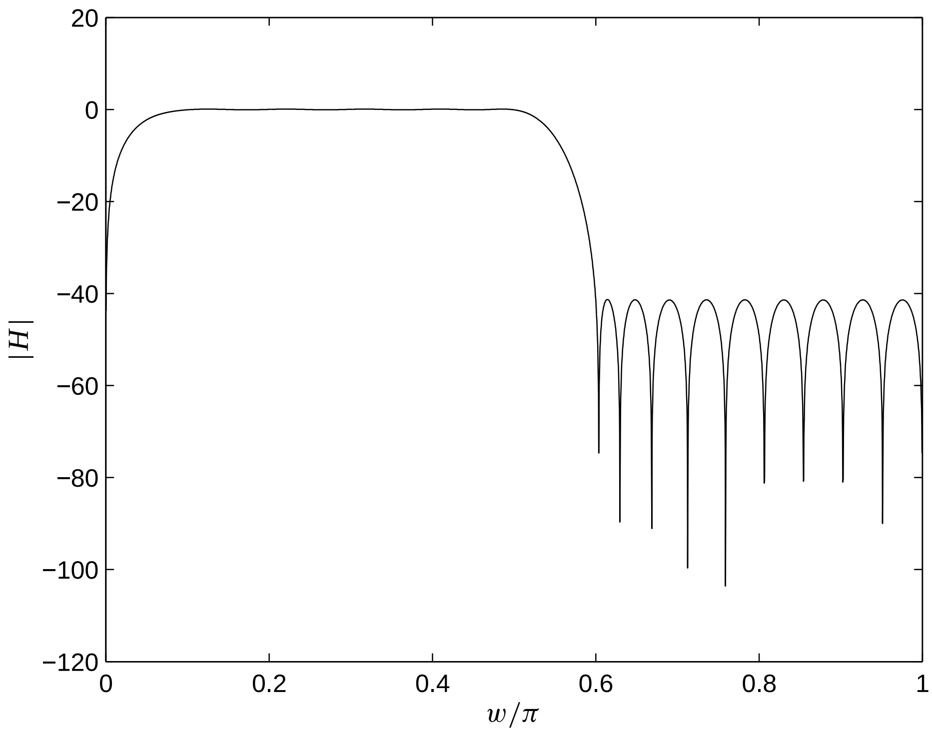

Figure 21: The magnitude of the frequency response of the filter

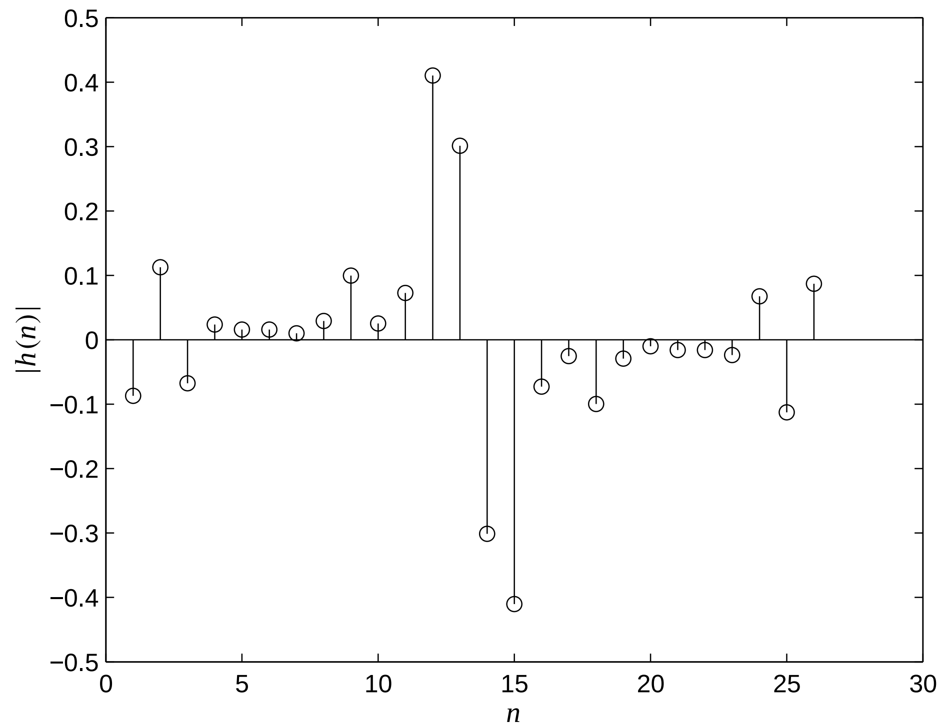

Figure 22: Impulse response of the filter

plot(W/**pi**, **angle**(H));

20

Problem 2.8
1) The impulse response of the filter is given in Figure 22.
2) The magnitude of the frequency response of the filter is presented in Figure 23.
3) The filter output $y(n)$ and $x(n)$ are presented in Figure 24. It should be noted that $y(n)$ is the derivative of $x(n)$.

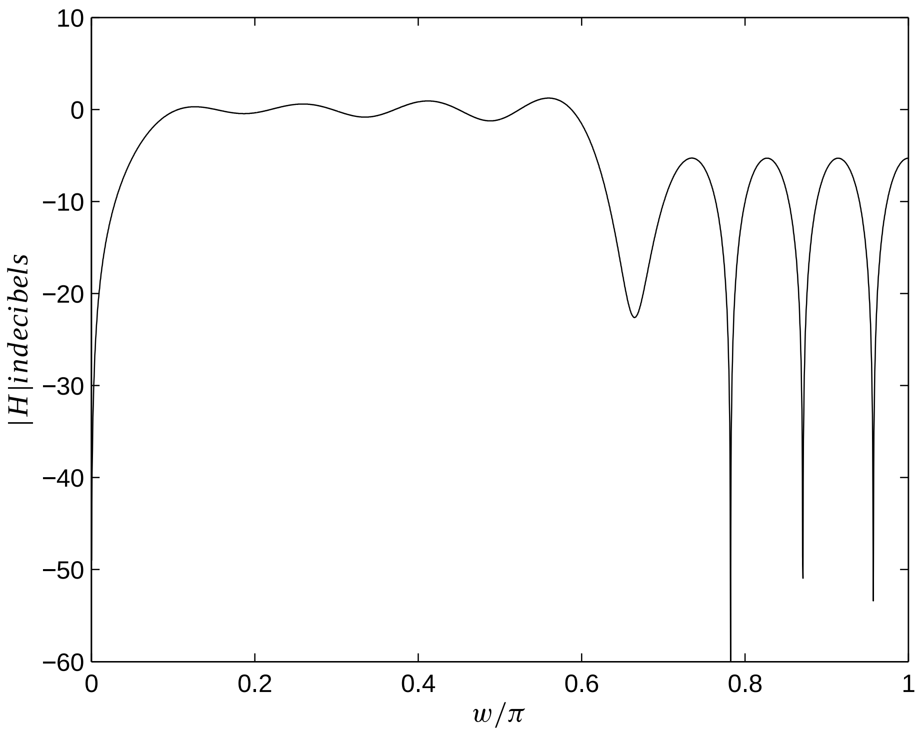

Figure 23: Magnitude of the frequency response of the filter

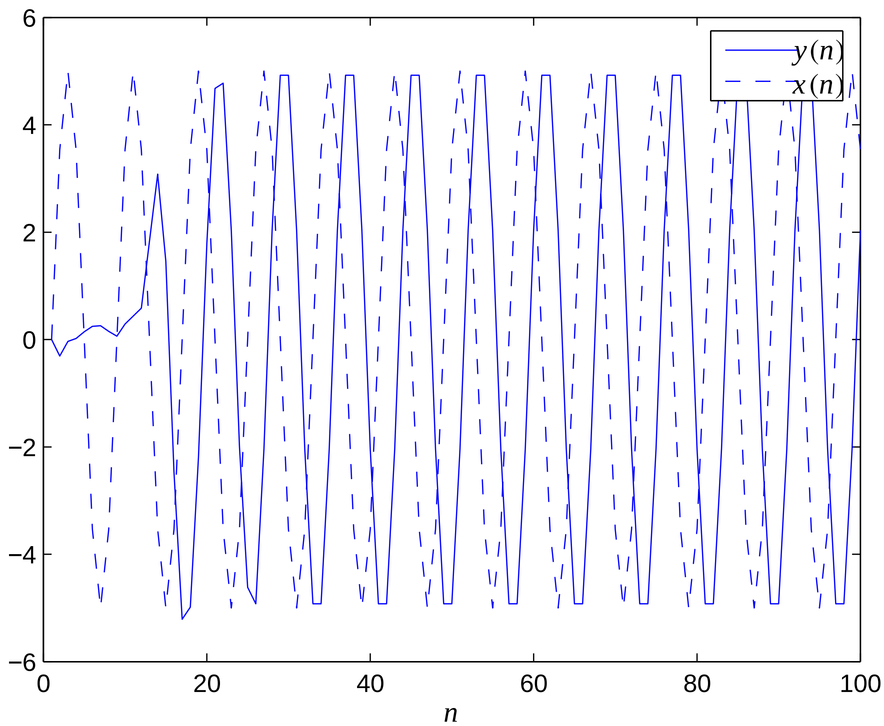

Figure 24: Signals $x(n)$ and $y(n)$