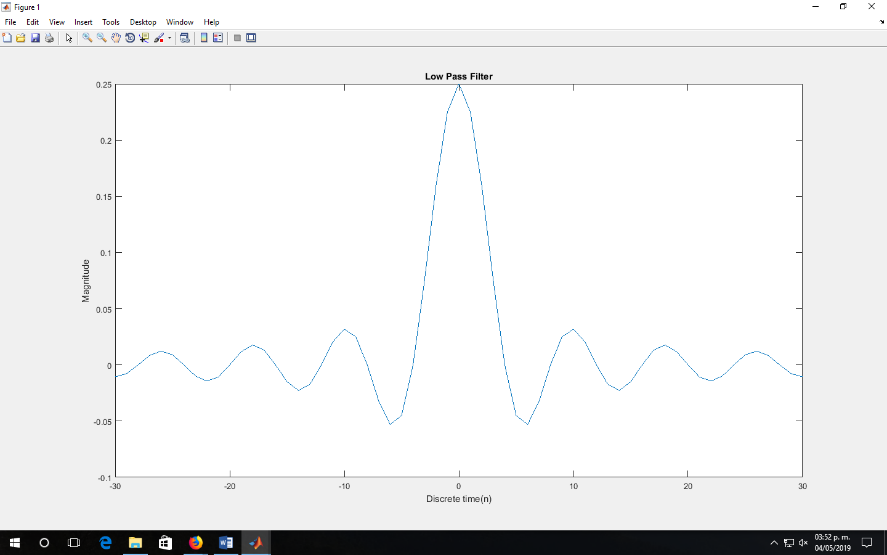
**Design of a FIR Digital Filter**

Andrea Perez Huizar  
IE698276

**Part I**

Design a low pass filter that has a cut off frequency of π/4, firstly having a length of 11 elements and later on of 17 elements.

1. Calculate the ideal h(n) for the desired cut off frequency
   1. Define n from -30 to 30
   2. Define h(n) for low pass filter
   3. Graph h(n)

%Define h(n)

n=-30:30;

wc=pi/4;

h1=(wc/pi)\*sinc(wc\*n/pi);

figure;

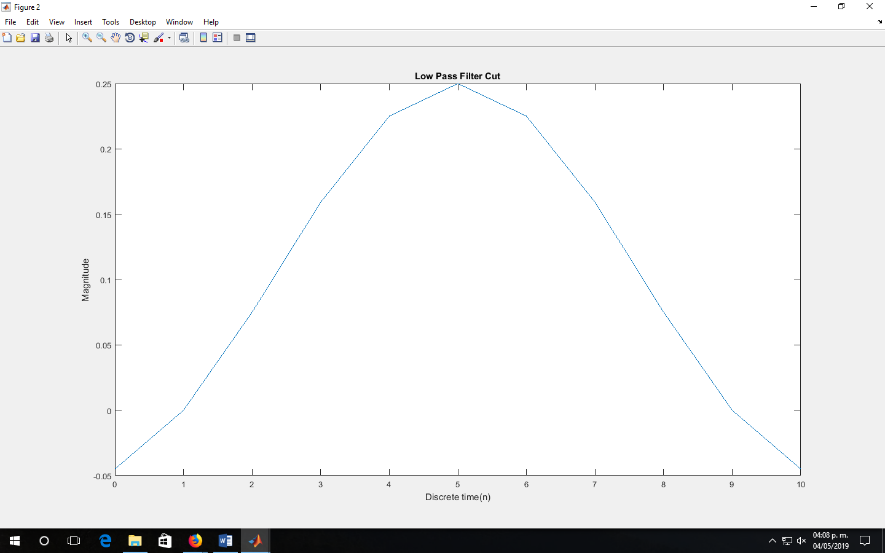
plot(n,h1);

xlabel('Discrete time(n)');

ylabel('Magnitude');

title('Low Pass Filter');

1. Truncate h(n) using a rectangular window that has a length of 11 elements
   1. Define n from 0 to 10
   2. Define new h(n) for truncate and transfer impulse with a length of 11 elements
   3. Copy values from first h(n) to new h(n), from n equals -5 to 5
   4. Graph new h(n)

%Truncate and transfer

n1=0:10;

h2=zeros(1,11);

h2=h1(26:36);

figure;

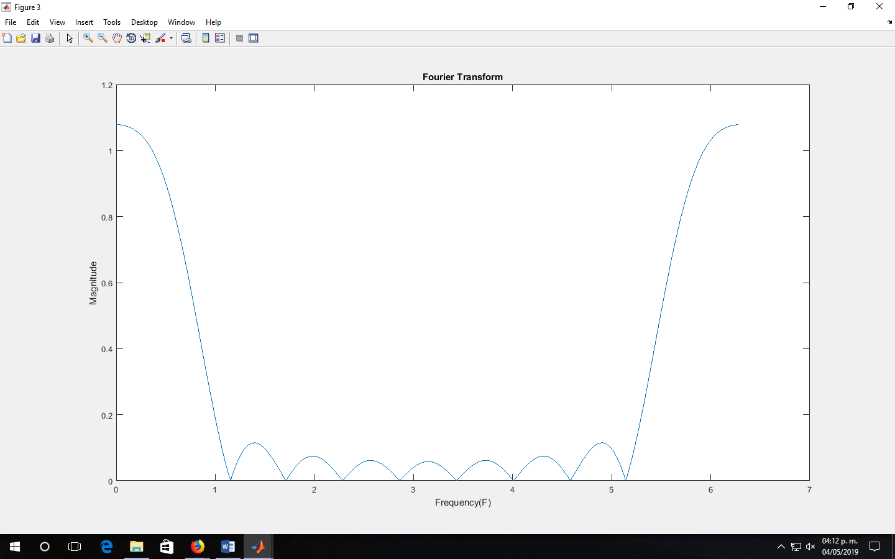
plot(n1,h2);

xlabel('Discrete time(n)')

ylabel('Magnitude')

title('Low Pass Filter Cut')

1. Check frequency response of the calculated filter
   1. Define angular frequency variable from 0 to 2π with 512 points
   2. Calculate system frequency for new h(n) for the FFT with 512 points
   3. Graph magnitude and angle
   4. Analyze the results

%FFT for h(n)

f=linspace(0,(2\*pi),512);

fft\_y=fft(h2,length(f));

mfft\_y=abs(fft\_y);

figure;

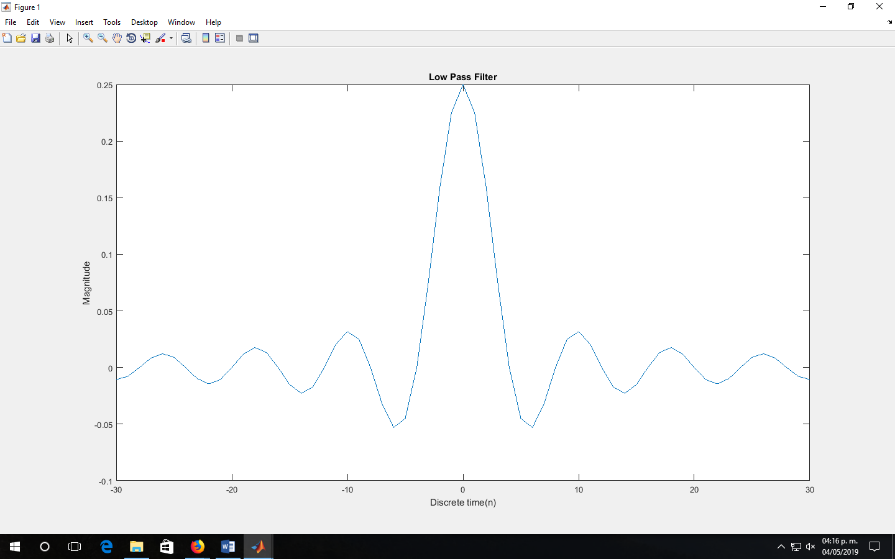
plot(f,mfft\_y);

xlabel('Frequency(F)')

ylabel('Magnitude')

title('Fourier Transform')

1. Redo steps for a filter of a 17 elements length

%Define h(n)

n=-30:30;

wc=pi/4;

h1=(wc/pi)\*sinc(wc\*n/pi);

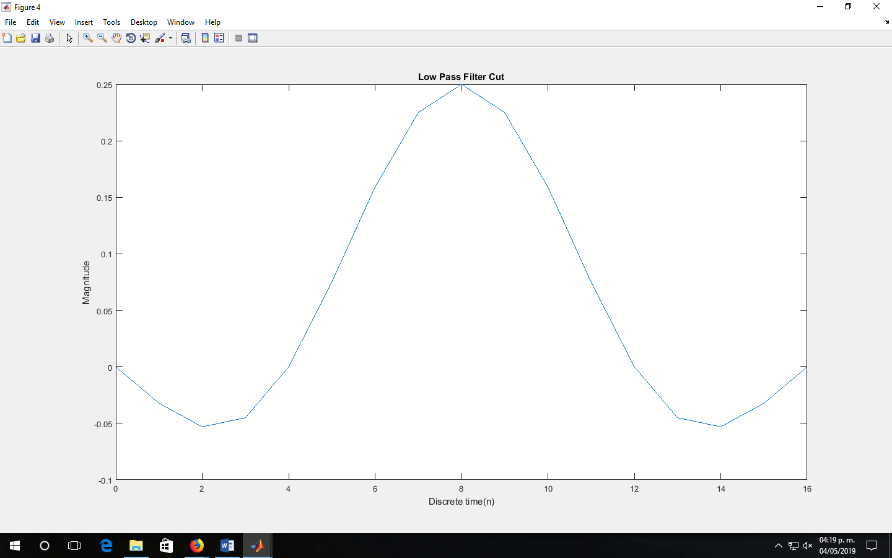
figure;

plot(n,h1);

xlabel('Discrete time(n)');

ylabel('Magnitude');

title('Low Pass Filter');

%Truncate and transfer

n1=0:16;

h2=zeros(1,17);

h2=h1(23:39);

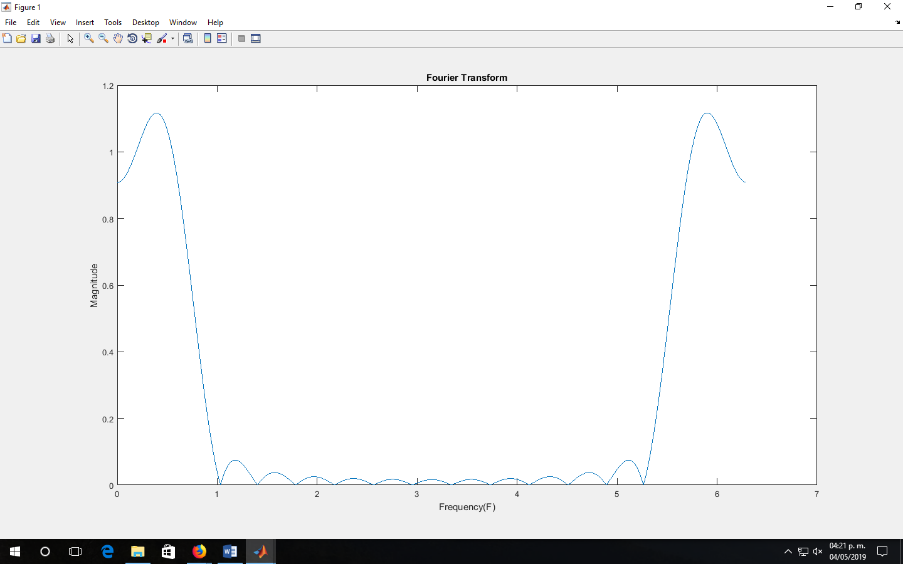
figure;

plot(n1,h2);

xlabel('Discrete time(n)')

ylabel('Magnitude')

title('Low Pass Filter Cut')

%FFT for h(n)

f=linspace(0,(2\*pi),512);

fft\_y=fft(h2,length(f));

mfft\_y=abs(fft\_y);

figure;

plot(f,mfft\_y);

xlabel('Frequency(F)')

ylabel('Magnitude')

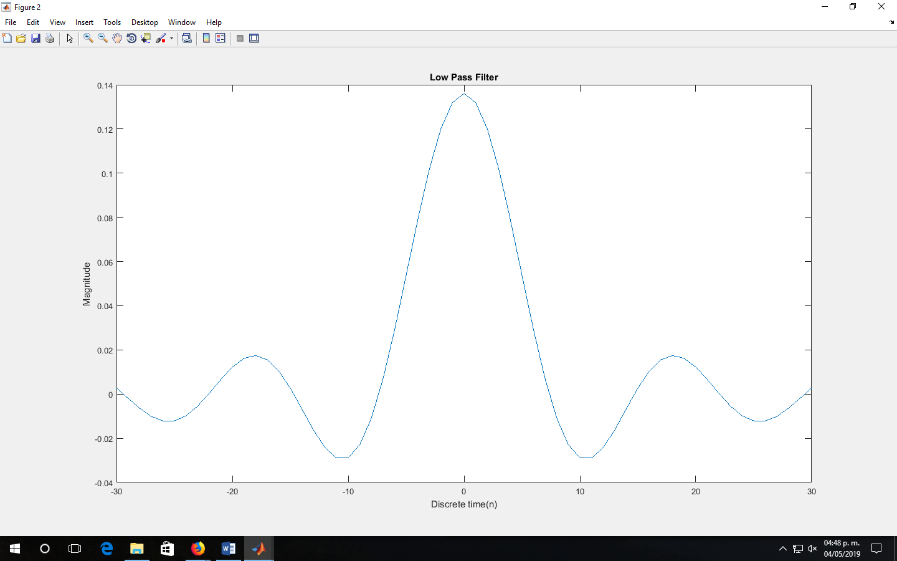
title('Fourier Transform')

Both filter even though they have different lengths are working properly since we can see in the graphs that the values that exceed π/4 are almost equal to zero, this means the higher frequencies are non-existent. We can assume the filter are correct.

**Part II**

Design and apply a low pass filter to an audio signal

1. Calculate the cut off frequency of a filter to apply it to an audio signal sampled at 44100m/s so that the cut off frequency in hertz corresponds to a 3KHz signal
   1. Operation to obtain wc frequency
   2. Design filter of a 17 element length
   3. Check frequency response in hertz
   4. Analyze frequency response

%Calculate wc

fs=44100;

fc=3000;

wc=2\*pi\*(fc/fs);

%Define h(n)

n=-30:30;

h1=(wc/pi)\*sinc(wc\*n/pi);

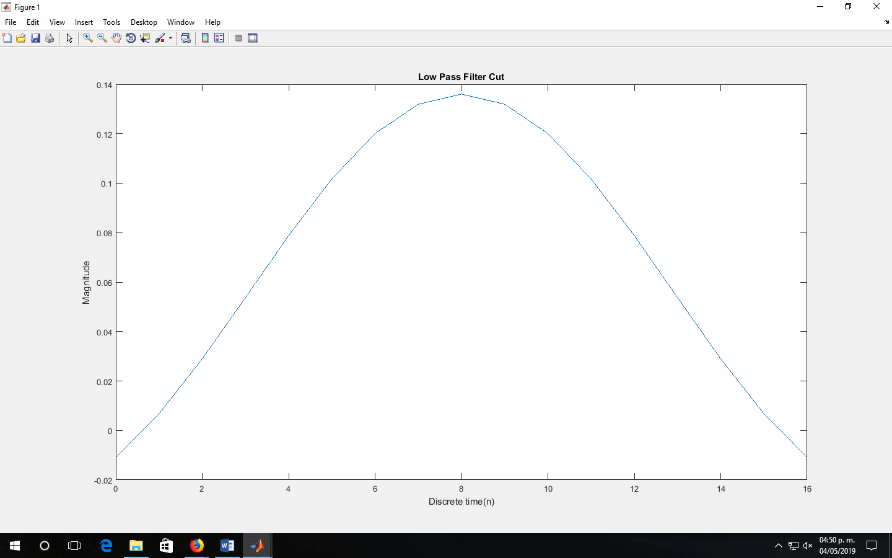
figure;

plot(n,h1);

xlabel('Discrete time(n)');

ylabel('Magnitude');

title('Low Pass Filter');

%Truncate and transfer

n1=0:16;

h2=zeros(1,17);

h2=h1(23:39);

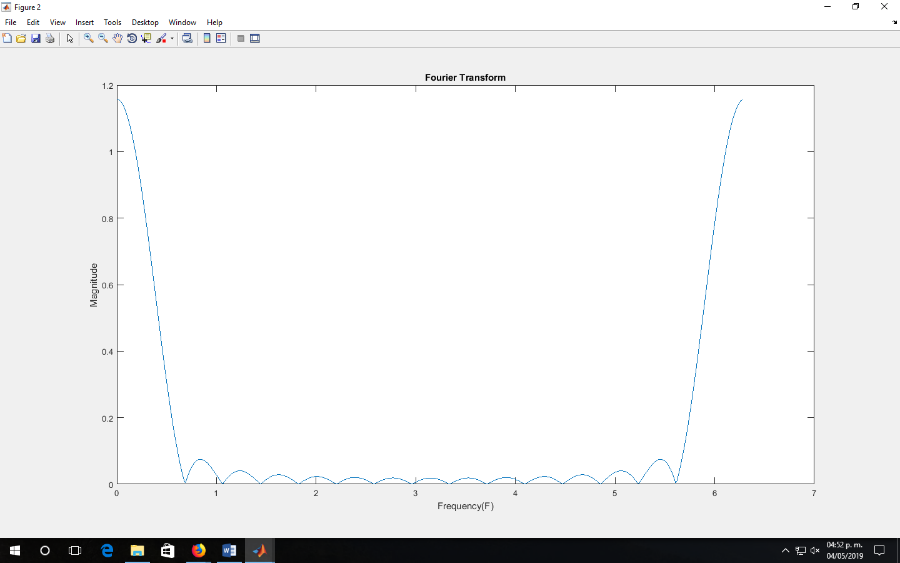
figure;

plot(n1,h2);

xlabel('Discrete time(n)')

ylabel('Magnitude')

title('Low Pass Filter Cut')

%FFT for h(n)

f=linspace(0,(2\*pi),512);

fft\_y=fft(h2,length(f));

mfft\_y=abs(fft\_y);

figure;

plot(f,mfft\_y);

xlabel('Frequency(F)')

ylabel('Magnitude')

title('Fourier Transform')