DSP Lab Assignment 5

Minsung Cho

Linear Phase Finite Impulse Response (FIR) Frequency Selective Filters

1. Modify your MATLAB function, FIRdesign, so that it can calculate even and odd length impulse responses.

Code:

```
%finite impulse response frequency selective filter function
function [y, ny] = FIRdesign(wl, wu, N)
  m= mod(N,2)
  if m~=0
    N=N-1
  end
  for n = 1:2*N
       h(n) = (sin(wu*n)-sin(wl*n))/(pi*n);
  end

  %the h is finite with the double the length of the window
  n = [1:2*N];

  [h_r, n_r] = reverse(h, n);

  [h_comp, h_r_comp, n_comp] = compsig(h, n, h_r, n_r);
  plot(n_comp,h_comp)
  plot(n_comp,h_r_comp)
  h_comp = h_comp + h_r_comp;

  for n = 1:N
    box(n) = 1;
  end
  n_box = -(N-1)/2:(N-1)/2;

  y=conv(h_comp, box);
  ny = [-length(y)/2+1:length(y)/2];
end
```

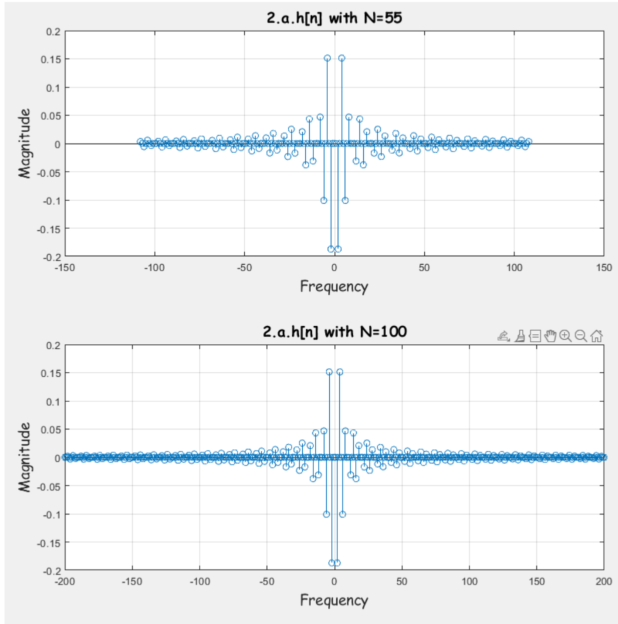
2. Consider an FIR filter which minimizes the mean square error between the desired zero phase frequency response, $H_d(w)$, and the filter's zero phase frequency response, $H_d(w)$, where $H_d(w)$ = 1 for $0.4\pi \le |w| \le 0.6\pi$ and 0 otherwise. Determine an impulse response, h[n], of a causal linear phase FIR filter which satisfies this criterion. Let h[n] have the lengths N = 55 and N = 100.
    a. Using stem, plot h[n].

Code:

```
N=55;
m= mod(N,2);
if m~=0
```

```
   N=N-1
end
u = 0.6*pi;
l = 0.4*pi;

for n = 1:2*N
       h(n) = (sin(u*n)-sin(l*n))/(pi*n);
end
n = [1:2*N];

[h_r, n_r] = reverse(h, n);

[h_comp, h_r_comp, n_comp] = compsig(h, n, h_r, n_r);
h_comp = h_comp + h_r_comp;
figure(1)
subplot(2,1,1)
stem(n_comp,h_comp)
title({'2.a.h[n] with N=55'},'fontname', 'Comic Sans MS', 'fontsize',14)
grid on
xlabel('Frequency','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Magnitude','fontname', 'Comic Sans MS', 'fontsize',14)

N2=100;
m= mod(N2,2);
if m~=0
   N2=N2-1
end
for n2 = 1:2*N2
       h2(n2) = (sin(u*n2)-sin(l*n2))/(pi*n2);
end
n2 = [1:2*N2];
[h_r2, n_r2] = reverse(h2, n2);

[h_comp2, h_r_comp2, n_comp2] = compsig(h2, n2, h_r2, n_r2);
h_comp2 = h_comp2 + h_r_comp2;
subplot(2,1,2)
stem(n_comp2,h_comp2)
title({'2.a.h[n] with N=100'},'fontname', 'Comic Sans MS', 'fontsize',14)
grid on
xlabel('Frequency','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Magnitude','fontname', 'Comic Sans MS', 'fontsize',14)
```

Plots:



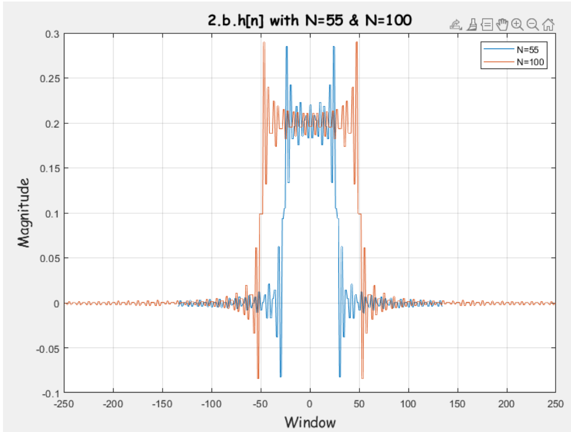
b. If you used one of the above windows, indicate which window, and plot the magnitude of that window's frequency spectrum.

I used both of the windows.

Code:

```
u = 0.6*pi;
l = 0.4*pi;

[y55,ny55] = FIRdesign(u, l, 55);
[y100,ny100] = FIRdesign(u, l, 100);
figure(1)
plot(ny55, y55, ny100, y100)
title({'2.a.h[n] with N=55 & N=100'},'fontname', 'Comic Sans MS', 'fontsize',14)
grid on
xlabel('Window','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Magnitude','fontname', 'Comic Sans MS', 'fontsize',14)
legend('N=55','N=100')
```

Plot:



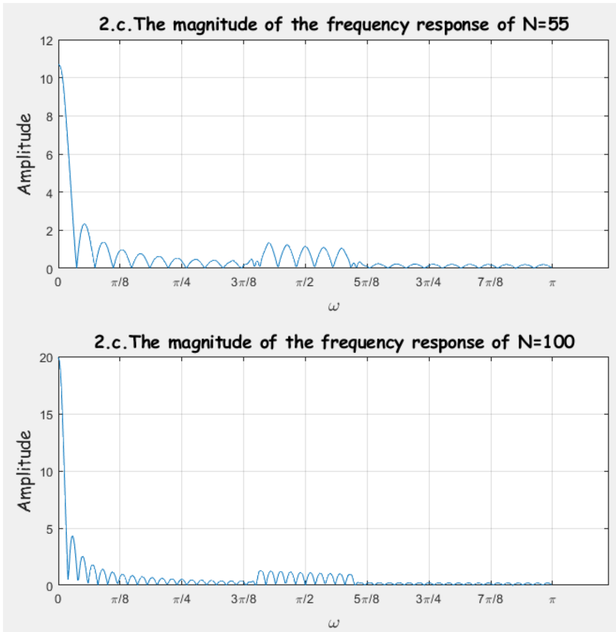
c. Plot the magnitude of the filter's frequency response.

Code:

```
u = 0.6*pi;
l = 0.4*pi;
[y55,ny55] = FIRdesign(u, l, 55);
[y100,ny100] = FIRdesign(u, l, 100);

[h1, w1] = freqz(y55, 1, 512);
mag1 = abs(h1);
rad2deg = 180/pi;
ang_h1 = wrapTo180(unwrap(angle(h1))*rad2deg);
[h2, w2] = freqz(y100, 1, 512);
mag2 = abs(h2);
rad2deg = 180/pi;
ang_h2 = wrapTo180(unwrap(angle(h2))*rad2deg);

figure(1)
subplot(2,1,1)
plot(w1,mag1)
grid on
title('2.c.The magnitude of the frequency response of N=55', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('\omega', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xticks([0 pi/8 pi/4 3*pi/8 pi/2 5*pi/8 3*pi/4 7*pi/8 pi])
xticklabels({'0' '\pi/8' '\pi/4' '3\pi/8' '\pi/2' '5\pi/8' '3\pi/4' '7\pi/8' '\pi'})

subplot(2,1,2)
plot(w2,mag2)
grid on
title('2.c.The magnitude of the frequency response of N=100', 'fontname', 'Comic Sans MS', 'fontsize',
14)
xlabel('\omega', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
```

```
xticks([0 pi/8 pi/4 3*pi/8 pi/2 5*pi/8 3*pi/4 7*pi/8 pi])
xticklabels({'0' '\pi/8' '\pi/4' '3\pi/8' '\pi/2' '5\pi/8' '3\pi/4' '7\pi/8' '\pi'})
```

Plots:

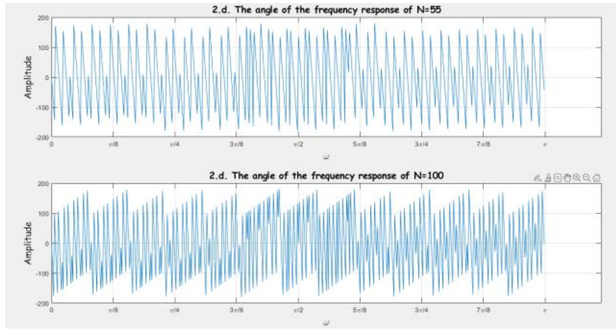


d. Give an expression as a function of w for filter's phase response.

Code:

```
figure(1)
subplot(2,1,1)
plot(w1,ang_h1)
grid on
title('2.d. The angle of the frequency response of N=55', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('\omega', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xticks([0 pi/8 pi/4 3*pi/8 pi/2 5*pi/8 3*pi/4 7*pi/8 pi])
xticklabels({'0' '\pi/8' '\pi/4' '3\pi/8' '\pi/2' '5\pi/8' '3\pi/4' '7\pi/8' '\pi'})

subplot(2,1,2)
plot(w2,ang_h2)
grid on
title('2.d. The angle of the frequency response of N=100', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('\omega', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xticks([0 pi/8 pi/4 3*pi/8 pi/2 5*pi/8 3*pi/4 7*pi/8 pi])
xticklabels({'0' '\pi/8' '\pi/4' '3\pi/8' '\pi/2' '5\pi/8' '3\pi/4' '7\pi/8' '\pi'})
```

Plots:



3. Write MATLAB functions called bartlett, hanning2, hamming2 and blackman which generate Bartlett, Hanning, Hamming and Blackman windows respectively.
    a. For each of the windows, plot w[n] for N = 55 and N = 100.

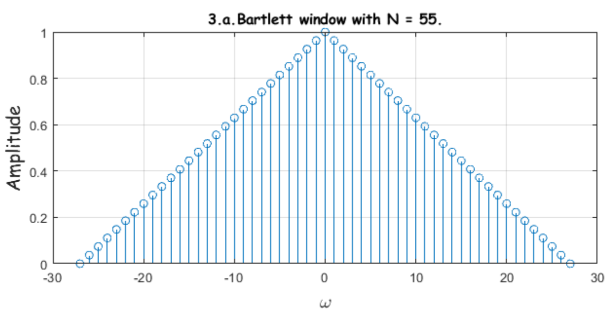
Bartlett

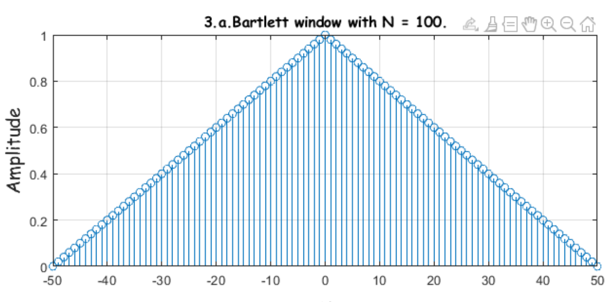
Code:

```
N = 55;
modulus = mod(N, 2);
if modulus ==0
   N = N+1;
end
n=1;
for i = -(N-1)/2:0
   temp1(n)=2*i/(N-1)+1;
   n = n+1;
end
n = [-(N-1)/2: 0];
figure(1)

m=1;
for j = 0:(N-1)/2
   temp2(m)=1-2*j/(N-1);
   m = m+1;
end
m=[0:(N-1)/2];
figure(2)

[s1, s2, n] = compsig(temp1, n, temp2, m);
s1(find(s1==1)) = s1(find(s1==1))-1;
s1 = s2 + s1;
stem(n,s1)
N = 100;
modulus = mod(N, 2);
if modulus ==0
   N = N+1;
end
n=1;
```

```
for i = -(N-1)/2:0
   temp1(n)=2*i/(N-1)+1;
   n = n+1;
end
n = [-(N-1)/2: 0];
figure(1)

m=1;
for j = 0:(N-1)/2
   temp2(m)=1-2*j/(N-1);
   m = m+1;
end
m=[0:(N-1)/2];
figure(2)

[s1, s2, n] = compsig(temp1, n, temp2, m);
s1(find(s1==1)) = s1(find(s1==1))-1;
s1 = s2 + s1;
stem(n,s1)
```

Plots:





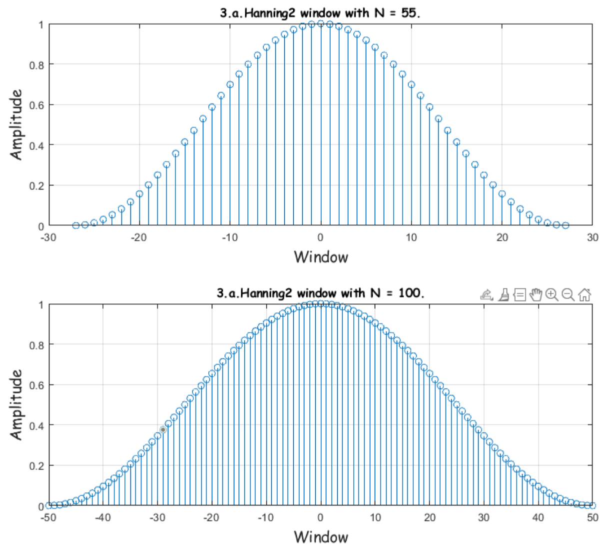
Hanning2

Code:

```
clearvars
```

```
N = 55;
modulus = mod(N, 2);
if modulus ==0
   N = N+1;
end
n=1;
for i = -(N-1)/2:(N-1)/2
   temp1(n)=0.5+0.5*cos(2*pi*i/(N-1));
   n = n+1;
end
n = [-(N-1)/2: (N-1)/2];
figure(1)

subplot(2,1,1)
stem(n,temp1)
grid on
title('3.a.Hanning2 window with N = 55.', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('Window', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)

clearvars

N = 100;
modulus = mod(N, 2);
if modulus ==0
   N = N+1;
end
n=1;
for i = -(N-1)/2:(N-1)/2
   temp1(n)=0.5+0.5*cos(2*pi*i/(N-1));
   n = n+1;
end
n = [-(N-1)/2: (N-1)/2];
subplot(2,1,2)
stem(n,temp1)
grid on
title('3.a.Hanning2 window with N = 100.', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('Window', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
```

Plots:



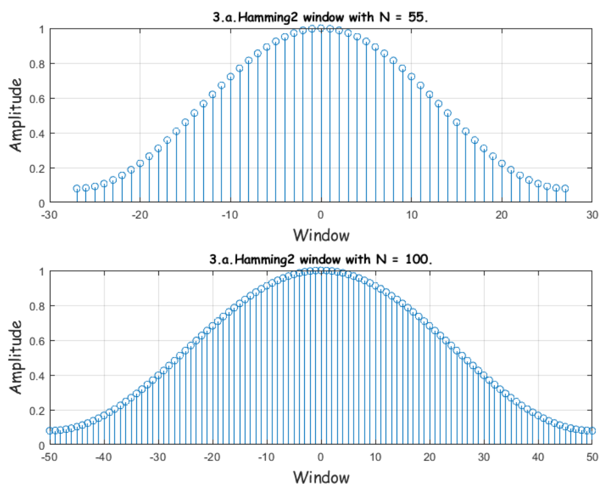
Hamming2

Code:

```
clearvars

N = 55;
modulus = mod(N, 2);
if modulus ==0
   N = N+1;
end
n=1;
for i = -(N-1)/2:(N-1)/2
   temp1(n)=0.54+0.46*cos(2*pi*i/(N-1));
   n = n+1;
end
n = [-(N-1)/2: (N-1)/2];
figure(1)

subplot(2,1,1)
stem(n,temp1)
grid on
title('3.a.Hamming2 window with N = 55.', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('Window', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)

clearvars

N = 100;
modulus = mod(N, 2);
```

```
if modulus ==0
   N = N+1;
end
n=1;
for i = -(N-1)/2:(N-1)/2
   temp1(n)=0.54+0.46*cos(2*pi*i/(N-1));
   n = n+1;
end
n = [-(N-1)/2: (N-1)/2];
subplot(2,1,2)
stem(n,temp1)
grid on
title('3.a.Hamming2 window with N = 100.', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('Window', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
```

Plots:



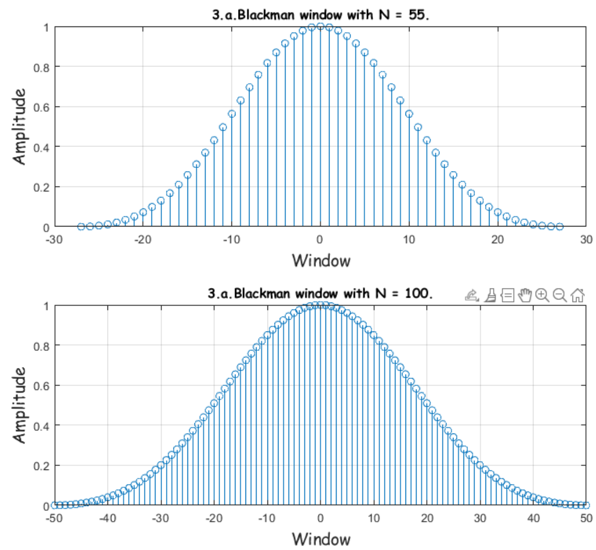
Blackman

Code:

```
clearvars

N = 55;
modulus = mod(N, 2);
if modulus ==0
   N = N+1;
end
n=1;
for i = -(N-1)/2:(N-1)/2
   temp1(n)=0.42+0.5*cos(2*pi*i/(N-1))+0.08*cos(4*pi*i/(N-1));
```

```
   n = n+1;
end
n = [-(N-1)/2: (N-1)/2];
figure(1)

subplot(2,1,1)
stem(n,temp1)
grid on
title('3.a.Blackman window with N = 55.', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('Window', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)

clearvars

N = 100;
modulus = mod(N, 2);
if modulus ==0
   N = N+1;
end
n=1;
for i = -(N-1)/2:(N-1)/2
   temp1(n)=0.42+0.5*cos(2*pi*i/(N-1))+0.08*cos(4*pi*i/(N-1));
   n = n+1;
end
n = [-(N-1)/2: (N-1)/2];
subplot(2,1,2)
stem(n,temp1)
grid on
title('3.a.Blackman window with N = 100.', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('Window', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
```

Plots:



b. Plot the magnitude of the frequency spectrum of each of the windows when N = 100.

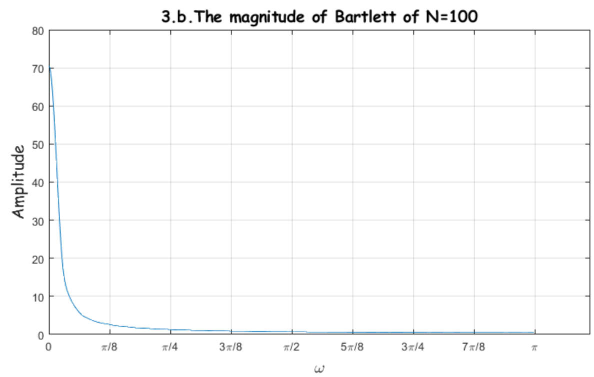
Bartlett

Code:

```
N = 100;
modulus = mod(N, 2);
if modulus ==0
   N = N+1;
end
n=1;
for i = -(N-1)/2:0
   temp1(n)=2*i/(N-1)+1;
   n = n+1;
end
n = [-(N-1)/2: 0];
figure(1)

m=1;
for j = 0:(N-1)/2
   temp2(m)=1-2*j/(N-1);
   m = m+1;
end
m=[0:(N-1)/2];
figure(2)

[s1, s2, n] = compsig(temp1, n, temp2, m);
s1(find(s1==1)) = s1(find(s1==1))-1;
s1 = s2 + s1;

[h1, w1] = freqz(s1, 1, 512);
```

```
mag1 = abs(h1);
rad2deg = 180/pi;
ang_h1 = wrapTo180(unwrap(angle(h1))*rad2deg);

plot(w1,mag1)
grid on
title('3.b.The magnitude of Bartlett of N=100', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('\omega', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xticks([0 pi/8 pi/4 3*pi/8 pi/2 5*pi/8 3*pi/4 7*pi/8 pi])
xticklabels({'0' '\pi/8' '\pi/4' '3\pi/8' '\pi/2' '5\pi/8' '3\pi/4' '7\pi/8' '\pi'})
```

Plot:



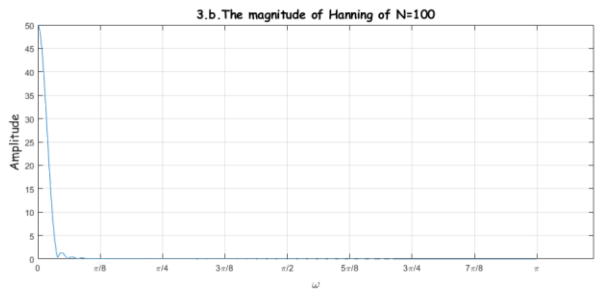
Hanning

Code:

```
N = 100;
modulus = mod(N, 2);
if modulus ==0
   N = N+1;
end
n=1;
for i = -(N-1)/2:(N-1)/2
   temp1(n)=0.5+0.5*cos(2*pi*i/(N-1));
   n = n+1;
end
n = [-(N-1)/2: (N-1)/2];

[h1, w1] = freqz(temp1, 1, 512);
mag1 = abs(h1);
rad2deg = 180/pi;
ang_h1 = wrapTo180(unwrap(angle(h1))*rad2deg);
```

```
plot(w1,mag1)
grid on
title('3.b.The magnitude of Hanning of N=100', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('\omega', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xticks([0 pi/8 pi/4 3*pi/8 pi/2 5*pi/8 3*pi/4 7*pi/8 pi])
xticklabels({'0' '\pi/8' '\pi/4' '3\pi/8' '\pi/2' '5\pi/8' '3\pi/4' '7\pi/8' '\pi'})
```

Plot:

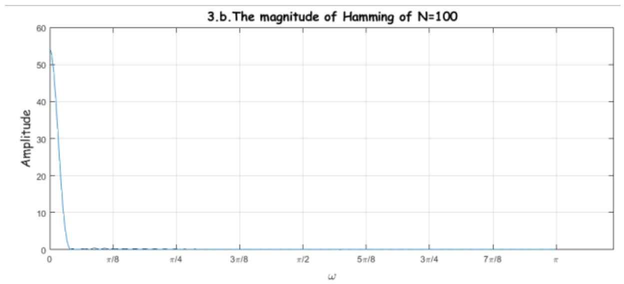


Hamming

Code:

```
N = 100;
modulus = mod(N, 2);
if modulus ==0
   N = N+1;
end
n=1;
for i = -(N-1)/2:(N-1)/2
   temp1(n)=0.54+0.46*cos(2*pi*i/(N-1));
   n = n+1;
end
n = [-(N-1)/2: (N-1)/2];

[h1, w1] = freqz(temp1, 1, 512);
mag1 = abs(h1);
rad2deg = 180/pi;
ang_h1 = wrapTo180(unwrap(angle(h1))*rad2deg);

plot(w1,mag1)
grid on
title('3.b.The magnitude of Hamming of N=100', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('\omega', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xticks([0 pi/8 pi/4 3*pi/8 pi/2 5*pi/8 3*pi/4 7*pi/8 pi])
xticklabels({'0' '\pi/8' '\pi/4' '3\pi/8' '\pi/2' '5\pi/8' '3\pi/4' '7\pi/8' '\pi'})
```

Plot:

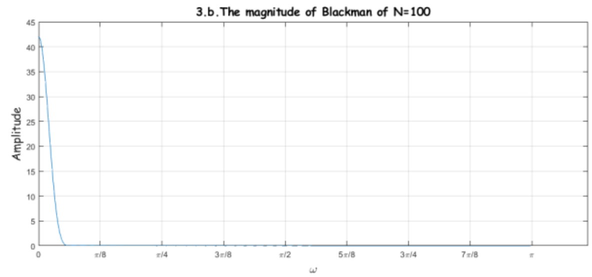


Blackman

Code:

```
N = 100;
modulus = mod(N, 2);
if modulus ==0
   N = N+1;
end
n=1;
for i = -(N-1)/2:(N-1)/2
   temp1(n)=0.42+0.5*cos(2*pi*i/(N-1))+0.08*cos(4*pi*i/(N-1));
   n = n+1;
end
n = [-(N-1)/2: (N-1)/2];


[h1, w1] = freqz(temp1, 1, 512);
mag1 = abs(h1);
rad2deg = 180/pi;
ang_h1 = wrapTo180(unwrap(angle(h1))*rad2deg);

plot(w1,mag1)
grid on
title('3.b.The magnitude of Blackman of N=100', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('\omega', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xticks([0 pi/8 pi/4 3*pi/8 pi/2 5*pi/8 3*pi/4 7*pi/8 pi])
xticklabels({'0' '\pi/8' '\pi/4' '3\pi/8' '\pi/2' '5\pi/8' '3\pi/4' '7\pi/8' '\pi'})
```

Plot:



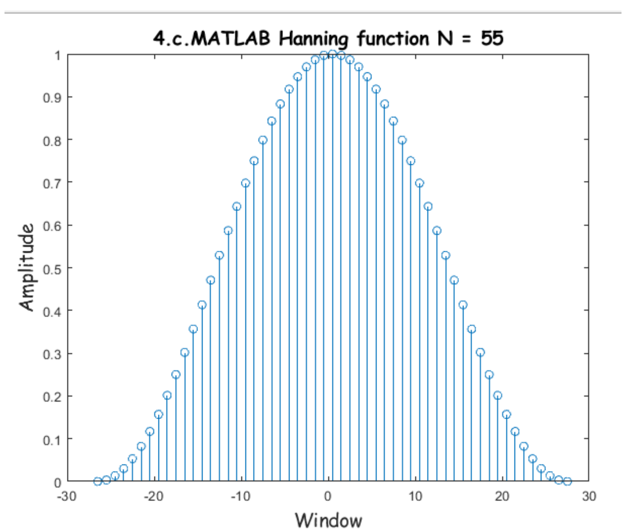
c. Compare your hanning2 and hamming2 functions to Matlab's built in functions, hanning and hamming.

MATLAB Hanning

Code:

```
N = 55;
x = hann(N);
nx = [-length(x)/2+1:length(x)/2];
stem(nx,x)
title('4.c.MATLAB Hanning function N = 55', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('Window', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
```

Plot:



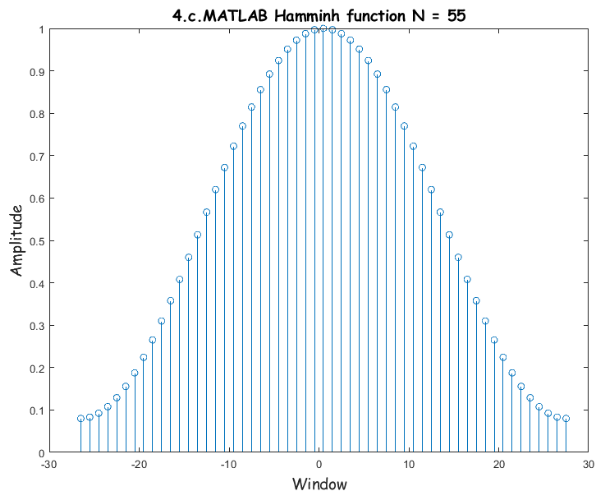
MATLAB Hamming

Code:

```
N = 55;
x = hamming(N);
nx = [-length(x)/2+1:length(x)/2];
```

```
stem(nx,x)
title('4.c.MATLAB Hamminh function N = 55', 'fontname', 'Comic Sans MS', 'fontsize', 14)
xlabel('Window', 'fontname', 'Comic Sans MS', 'fontsize', 14)
ylabel('Amplitude', 'fontname', 'Comic Sans MS', 'fontsize', 14)
```

Plot:



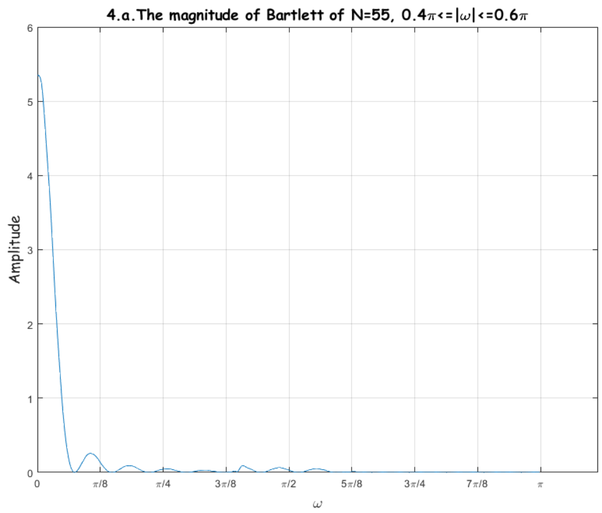
4. Using each of the window functions your wrote in Exercise 3, design a series of causal linear phase FIR filters of lengths N = 55 and N = 100 to a approximate the desired frequency response, $H_d(e^{jw})$, where $|H_d(e^{jw})| = 1$ for $0.4\pi \leq |w| \leq 0.6\pi$ and 0 otherwise.
   a. Plot the magnitude of the filter's frequency response.

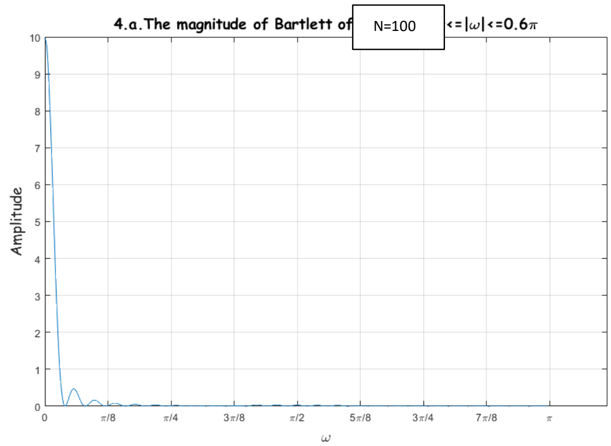
Bartlett

Code:

Skipped because it's redundant.

plots:



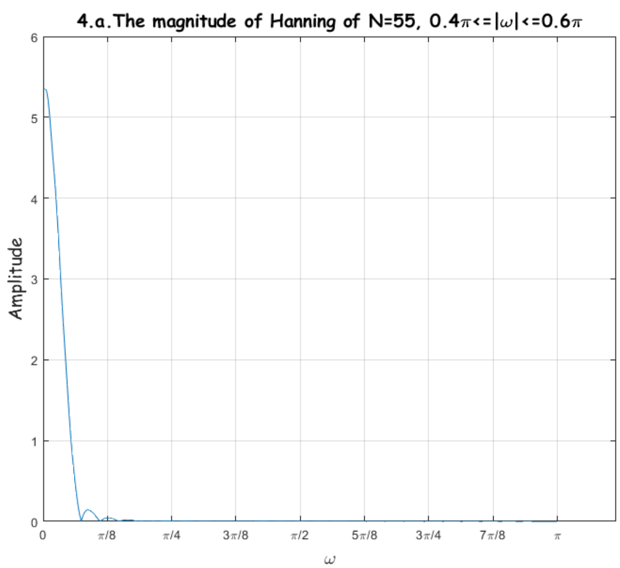


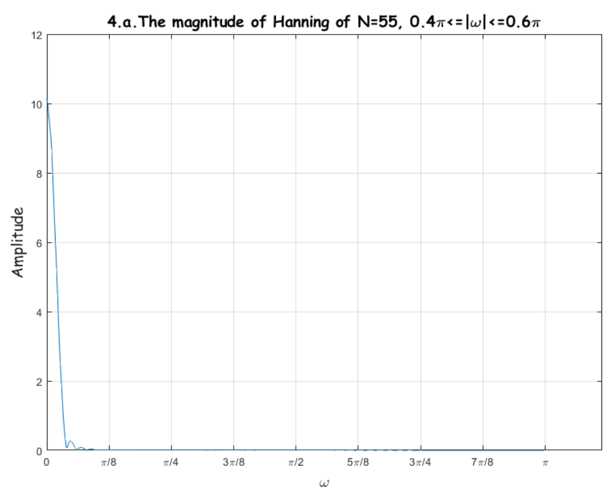
Hanning2

Code:

Again, redundant.

Plots:



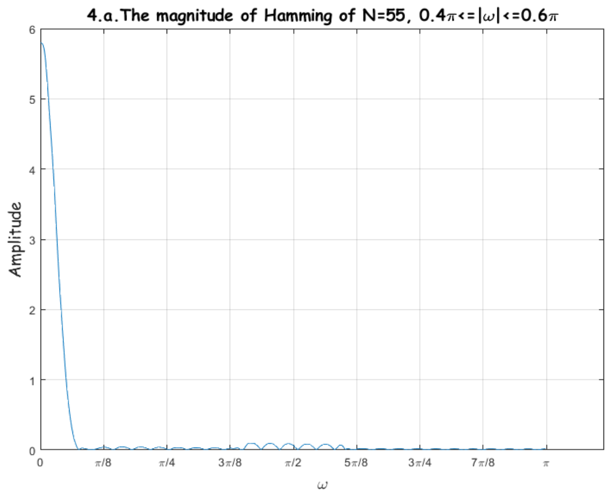


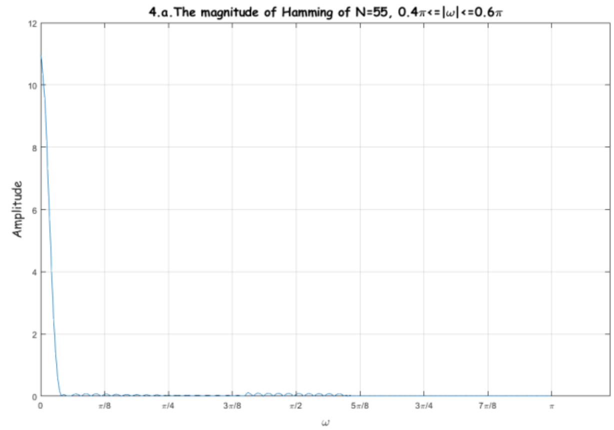
Hamming2

Code:

Redundant

Plots:



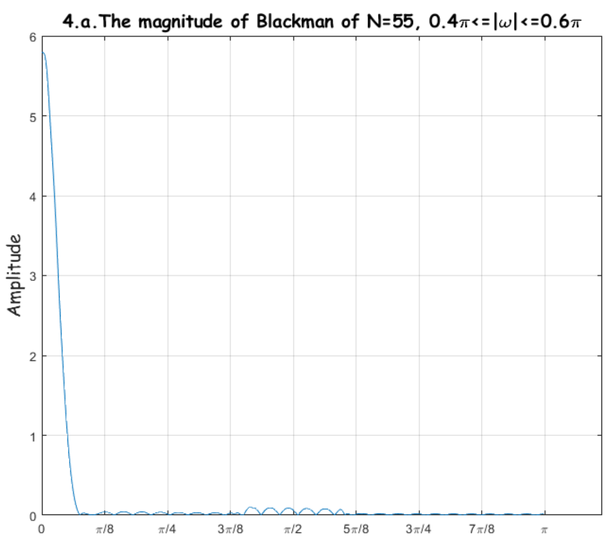


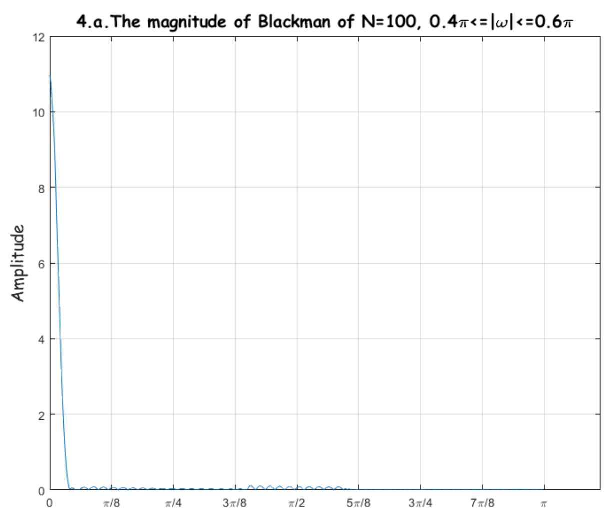
Blackman

Code:

Redundant

Plots:





b. Give an expression as a function of w for filter's phase response.

Arg{H(e^jw)}

5. Perform a pole zero plot of one of your filter designs where N = 55.

Blackman

Code:

```
N = 55;
l=0.4*pi;
u=0.6*pi;
[y, ny] = blakcman(l, u, N);

pzmap(y, ny)
grid on
mark = findobj(gca, 'type', 'line');
for i = 1:length(mark)
   set(mark(i), 'markersize', 14);
   set(mark(i), 'linewidth',2);
end
```

Plot: