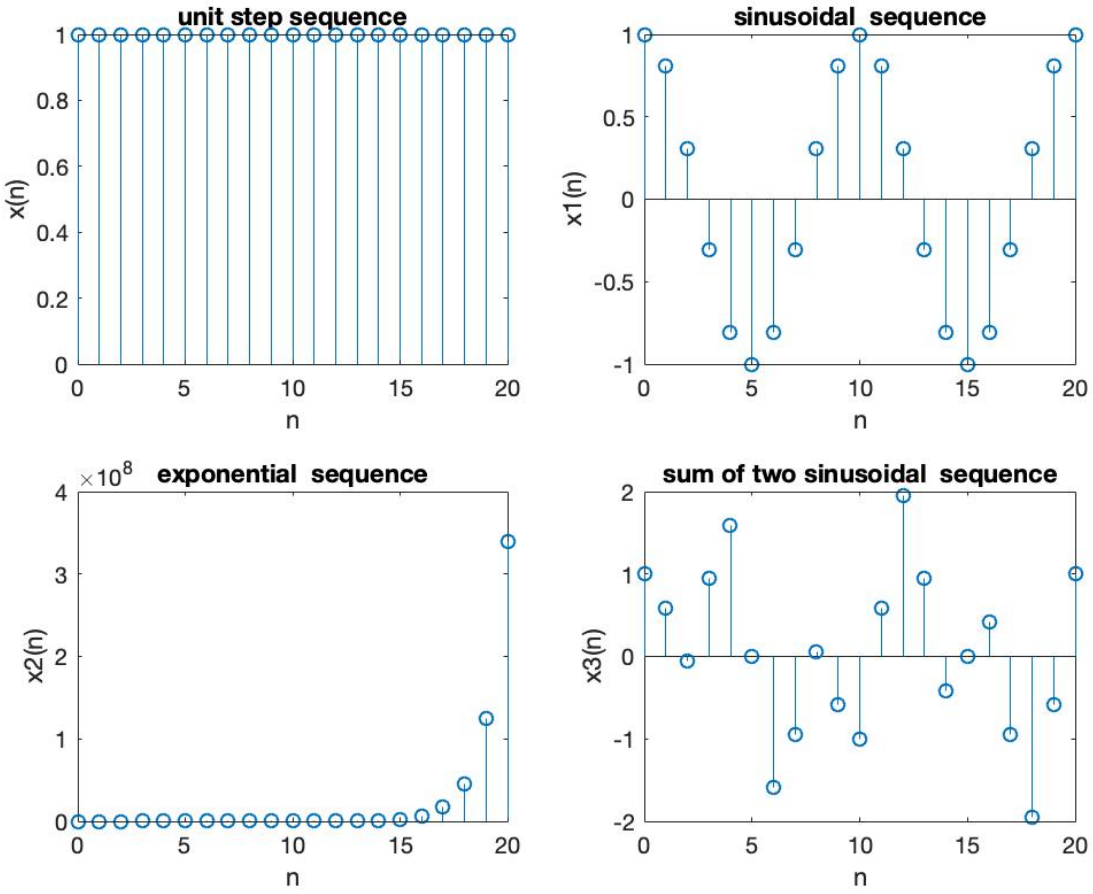
Generation of discrete time sequences:

Code:

```
N=21;
x=ones(1,N);
n=0:1:N-1;
subplot(2,2,1),stem(n,x);
xlabel('n'),ylabel('x(n)');
title('unit step sequence');
%sine
x1= cos(0.2*pi*n);
subplot(2,2,2),stem(n,x1);
xlabel('n'),ylabel('x1(n)');
title('sinusoidal  sequence');
%exponential
x2= 0.7*exp(n);
subplot(2,2,3),stem(n,x2);
xlabel('n'),ylabel('x2(n)');
title('exponential  sequence');
%sum of sine
x3= sin(0.2*pi*n)+cos(0.5*pi*n);
subplot(2,2,4),stem(n,x3);
xlabel('n'),ylabel('x3(n)');
title('sum of two sinusoidal  sequence’);
```

Output:


unit step sequence
x(n)
n
sinusoidal sequence
x1(n)
n
×10^8
exponential sequence
x2(n)
n
sum of two sinusoidal sequence
x3(n)
n

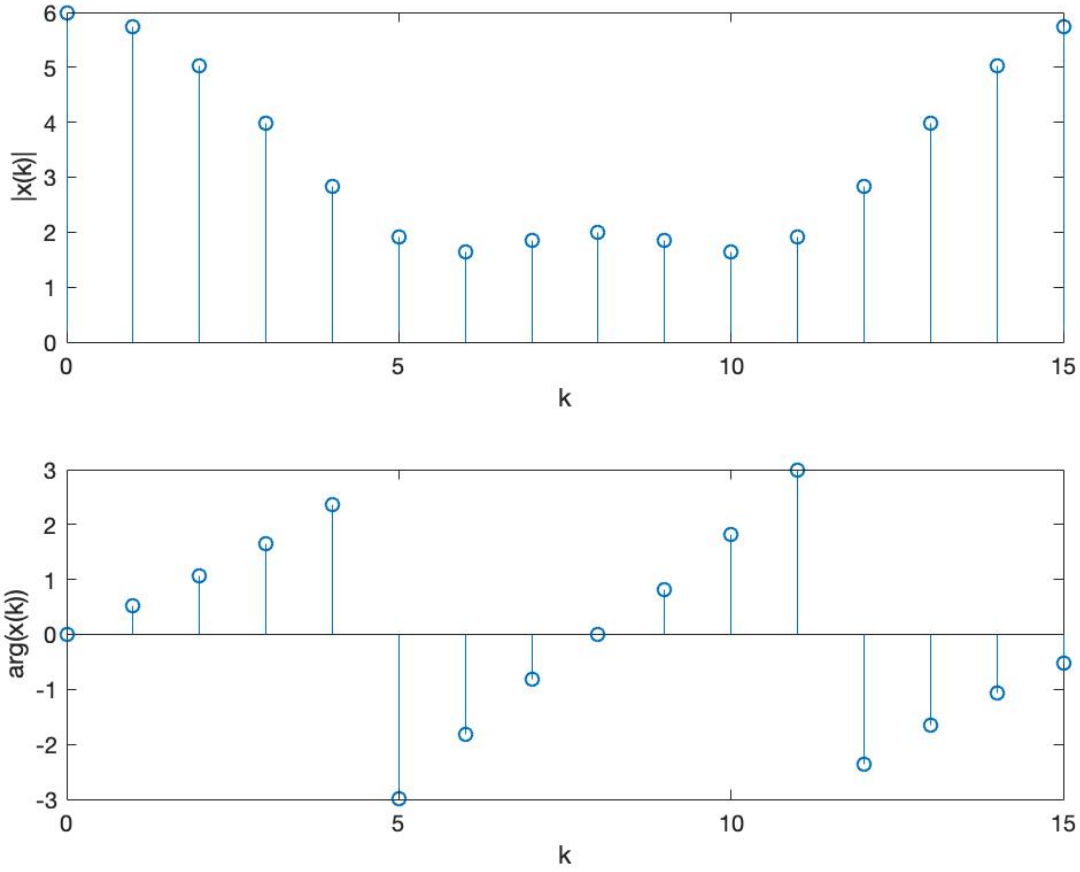
Magnitude and Phase plot

Code:

```
clear all;
xn=[1,2,3];
N=16;
Xk=dft(xn,N);
k=0:1:N-1;
subplot(2,1,1),stem(k,abs(Xk));
xlabel('k'),ylabel('|x(k)|');
subplot(2,1,2),stem(k,angle(Xk));
xlabel('k'),ylabel('arg(x(k))');


function X = dft(xn,N)
L=length(xn);
if(N<L)
    error('N must be >= L');
end
x1=[xn zeros(1,N-L)];
for k=0:1:N-1;
    for n=0:1:N-1;
        p=exp(-i*2*pi*n*k/N);
        x2(k+1,n+1)=p;
    end
end
X=x1*x2';
end
```

Output:



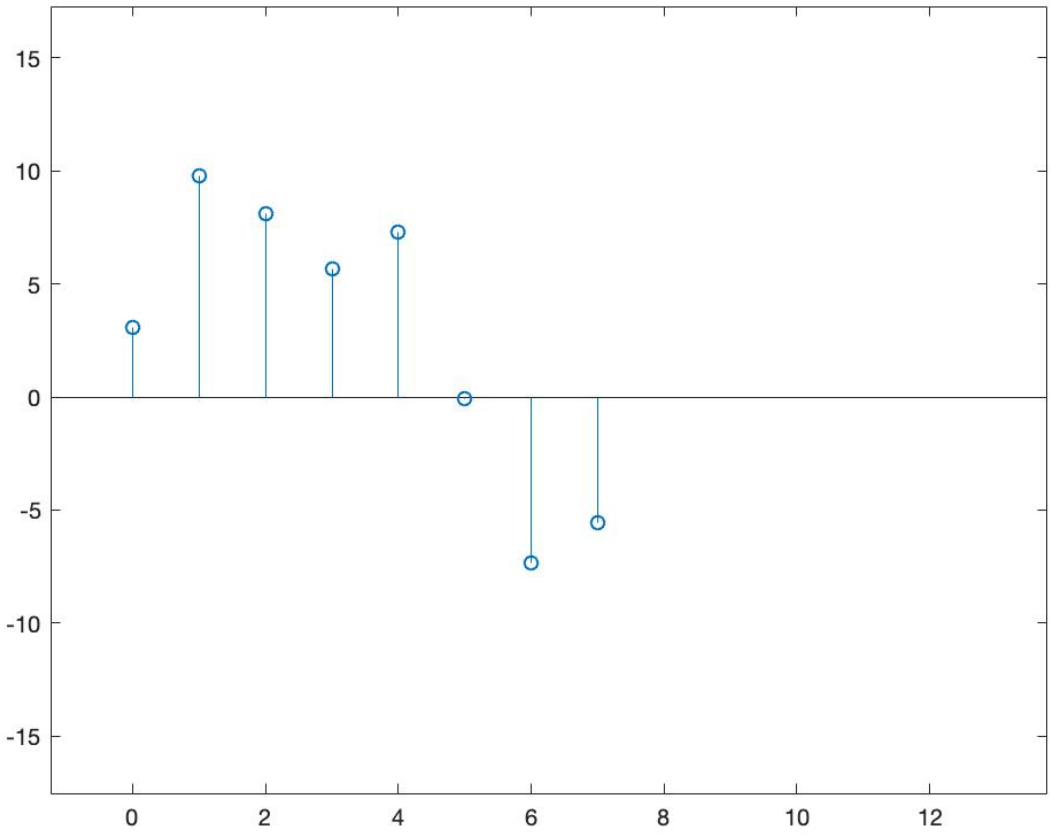
Circular Convolution

Code:

```
clear all;
n=0:7;
x=sin(3*pi*n/8);
h=[1,2,5,6];
Nx=length(x);
Nh=length(h);
N=8;
if(N<max(Nx,Nh))
    error('N must be >= max(Nx,Nh)')
end
y=circonv(x,h,N);
stem(n,y);

function [y] = circonv(x,h,N)
N2=length(x);
N3=length(h);
x=[x zeros(1,N-N2)];
h=[h zeros(1,N-N3)];
m=[0:1:N-1];
M=mod(-m,N);
h=h(M+1);
for n=1:1:N
    m=n-1;
    p=0:1:N-1;
    q=mod(p-m,N);
    hm=h(q+1);
    H(n,:)=hm;
end
y=x*H';
end
```

Output:

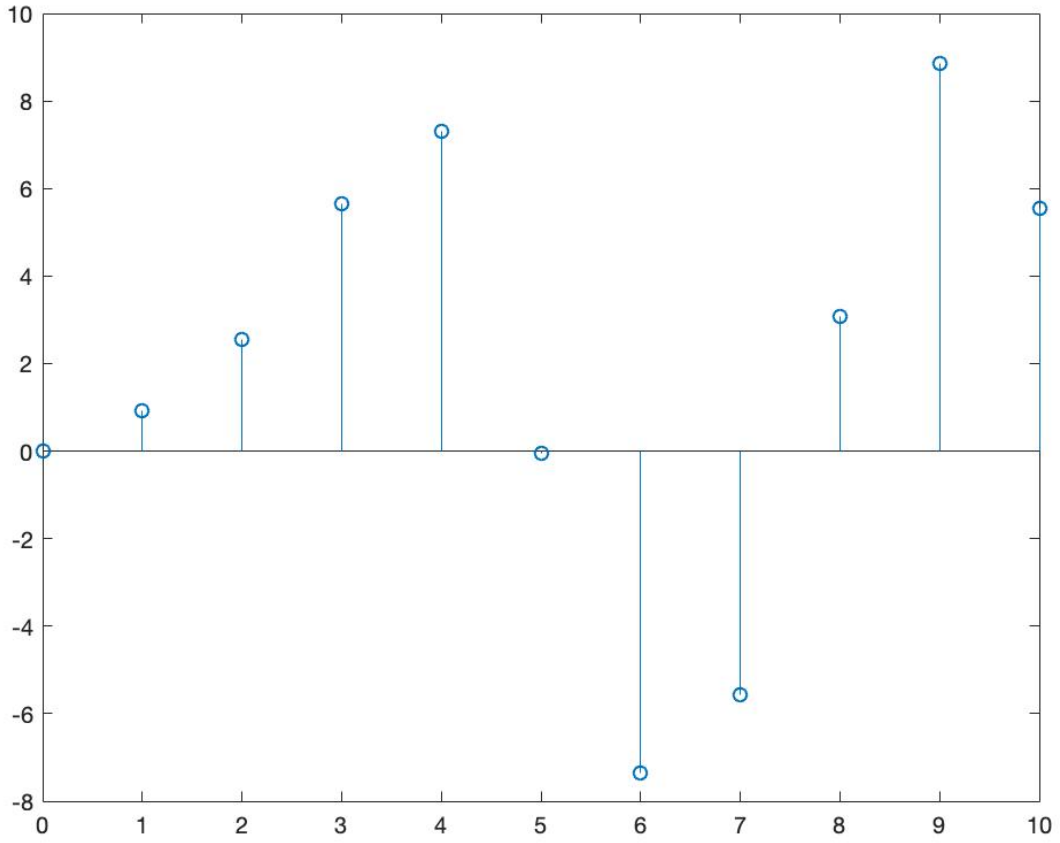


Linear Convolution

Code:

```
clear all;
n=0:7;
x=sin(3*pi*n/8);
h=[1,2,5,6];
Nx=length(x);
Nh=length(h);
y=conv(x,h);
x=0:10;
stem(x,y);
```

Output:



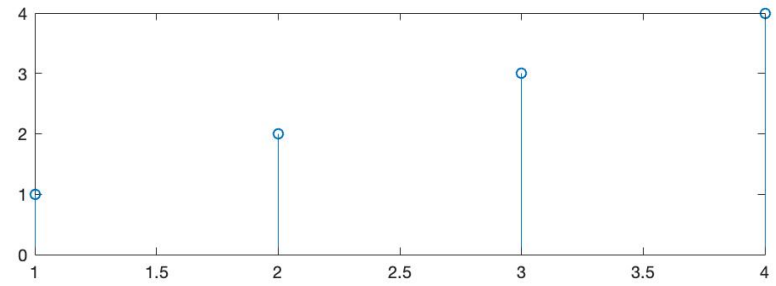
Autocorrelation and Crosscorrelation

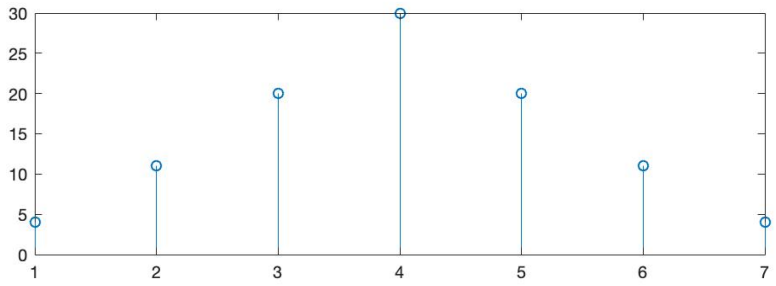
Code:

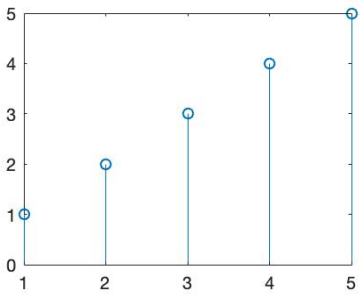
```
%autocorr
clear all;
x=input('enter');
rxx=xcorr(x);
subplot(2,1,1);
xlabel('n'),ylabel('x');
stem(x);
subplot(2,1,2);
xlabel('n'),ylabel('rxx');
stem(rxx);
%crosscorr
clear all;
x=input('enter');
y=input('enter');
rxy=xcorr(x,y);
subplot(2,2,1);
xlabel('n'),ylabel('x');
stem(x);
subplot(2,2,2);
xlabel('n'),ylabel('y');
stem(y);
subplot(2,2,3);
xlabel('n'),ylabel('rxy');
stem(rxy);
```

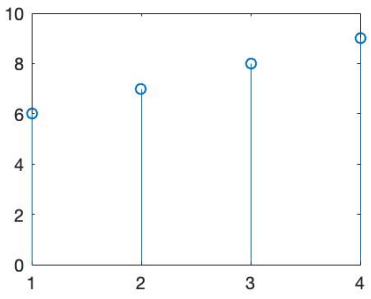
Output:

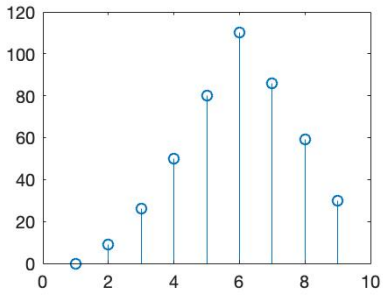
```
>> autoandcrosscorr
enter[1,2,3,4]
enter[1,2,3,4,5]
enter[6,7,8,9]
```











Butterworth IIR Filter

LPF

Code:

```
clear all;
alphap=4;
alphas=30;
fp=400;
fs=800;
F=2000;
omp=2*fp/F;oms=2*fs/F;
[n,wn]=buttord(omp,oms,alphap,alphas);
[b,a]=butter(n,wn)
w=0:0.01:pi;
[h,om]=freqz(b,a,w,'whole');
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(om/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(om/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency');
```

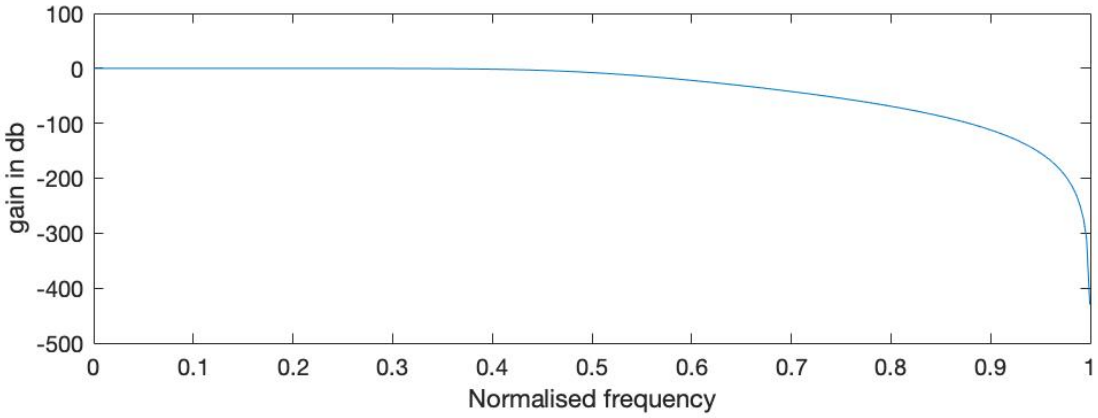
Output:

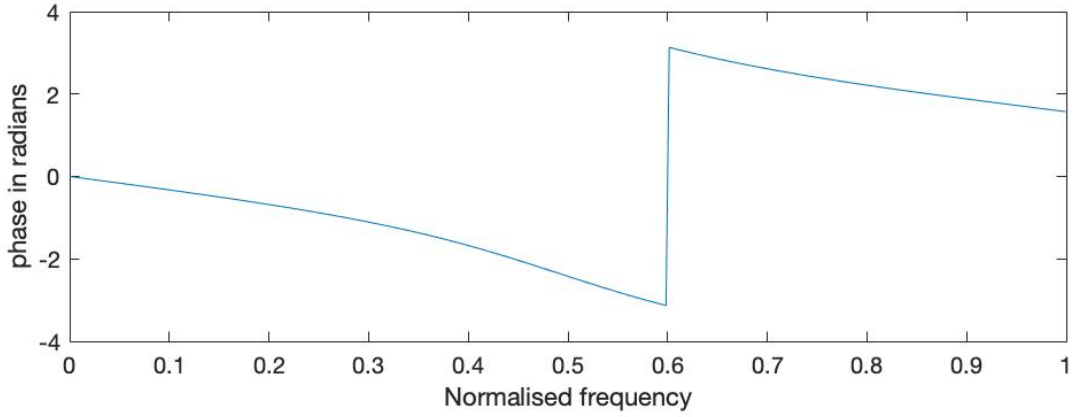
```
b =

   0.1600    0.4800    0.4800    0.1600


a =

   1.0000   -0.0494    0.3340   -0.0045
```


100
0
-100
-200
-300
-400
-500
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


4
2
0
-2
-4
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

HPF:

Code:

```
clear all;
alphap=4;
alphas=30;
fs=400;
fp=800;
F=2000;
omp=2*fp/F;oms=2*fs/F;
[n,wn]=buttord(omp,oms,alphap,alphas);
[b,a]=butter(n,wn,'high')
w=0:0.01:pi;
[h,om]=freqz(b,a,w,'whole');
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(om/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(om/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency’);
```

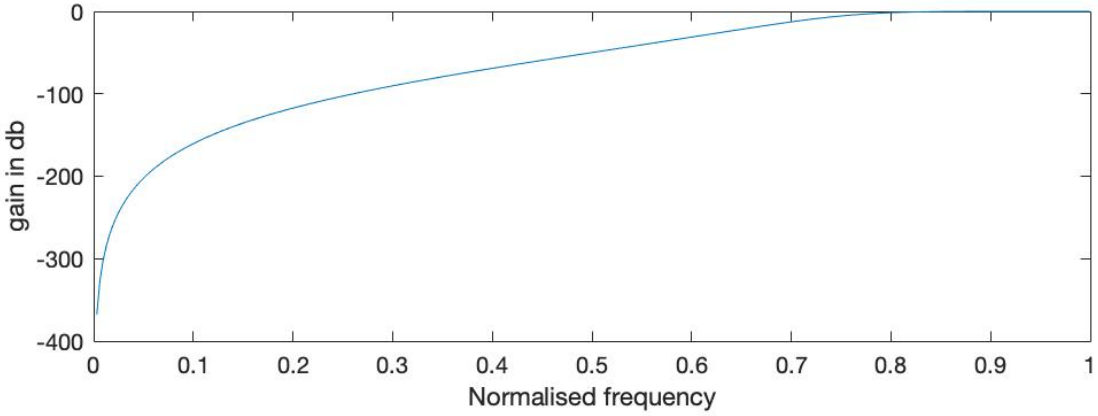
Output:

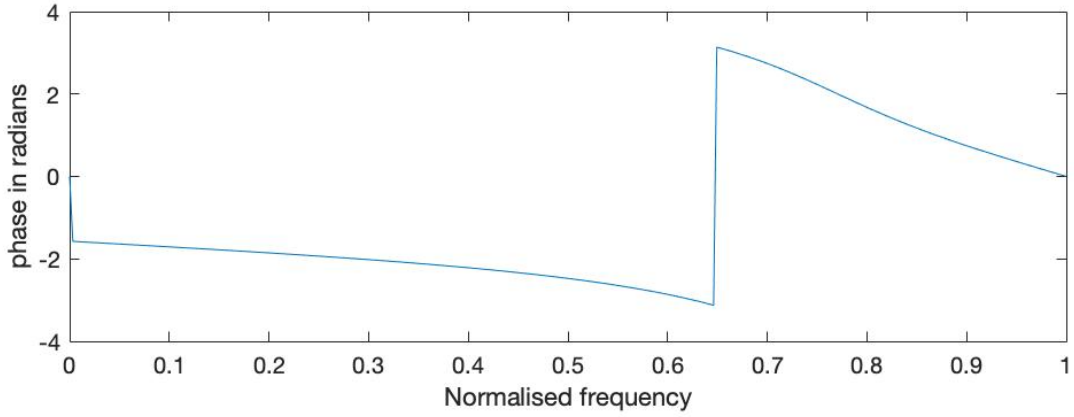
```
b =

   0.0354   -0.1061    0.1061   -0.0354


a =

   1.0000    1.3911    0.8566    0.1826
```


0
-100
-200
-300
-400
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


4
2
0
-2
-4
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

BPF:

Code:

```
clear all;
alphap=2;
alphas=20;
wp=[0.2*pi,0.4*pi];
ws=[0.1*pi,0.5*pi];
[n,wn]=buttord(wp/pi,ws/pi,alphap,alphas);
[b,a]=butter(n,wn)
w=0:0.01:pi;
[h,ph]=freqz(b,a,w);
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(ph/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(ph/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency’);
```

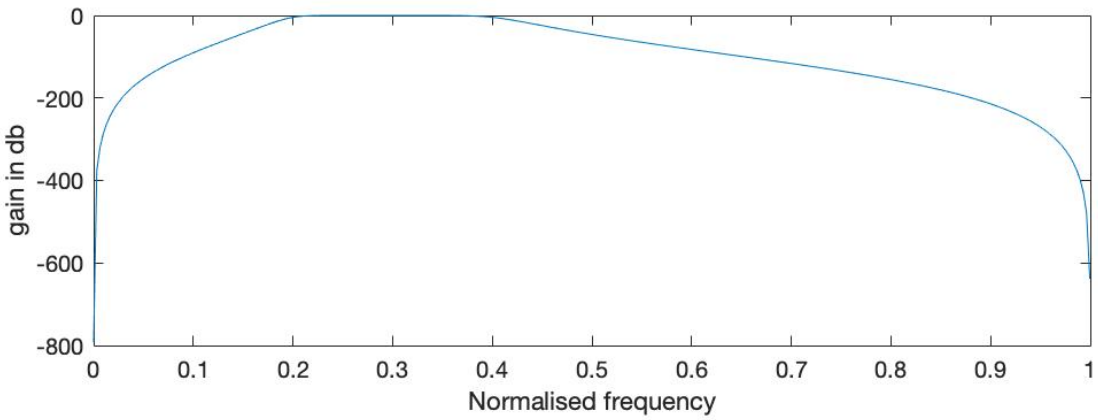
Output:

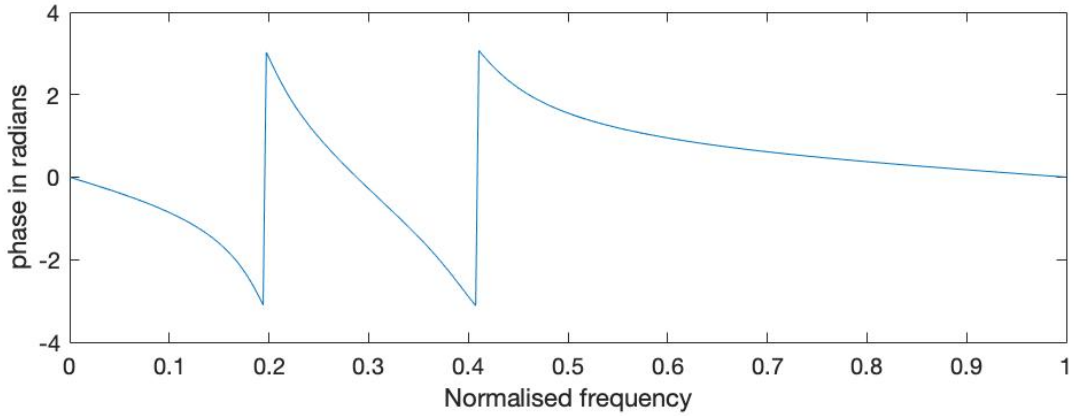
```
b =

   0.0060         0   -0.0240         0    0.0359         0   -0.0240         0    0.0060


a =

   1.0000   -3.8710    7.9699  -10.6417   10.0781   -6.8167    3.2579   -1.0044    0.1670
```


0
-200
-400
-600
-800
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


4
2
0
-2
-4
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

BSF:

Code:

```
clear all;
alphap=2;
alphas=20;
ws=[0.2*pi,0.4*pi];
wp=[0.1*pi,0.5*pi];
[n,wn]=buttord(wp/pi,ws/pi,alphap,alphas);
[b,a]=butter(n,wn,'stop')
w=0:0.01:pi;
[h,ph]=freqz(b,a,w);
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(ph/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(ph/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency');
```

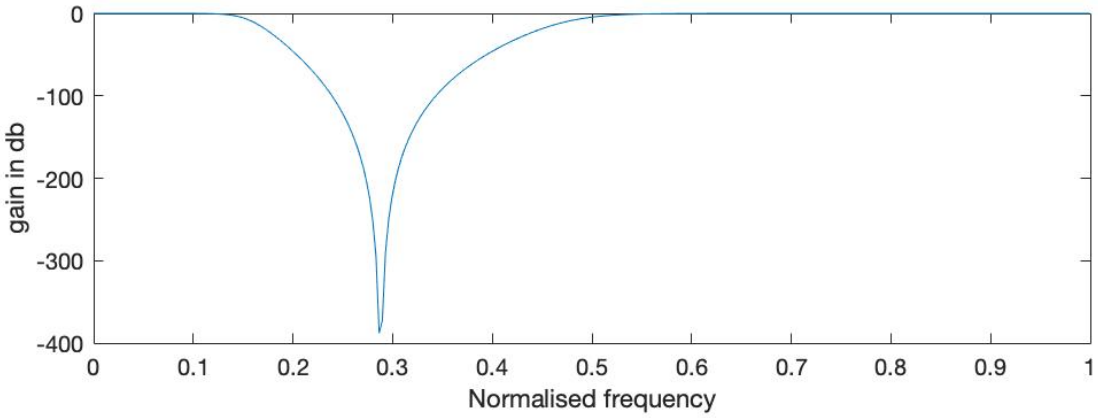
Output:

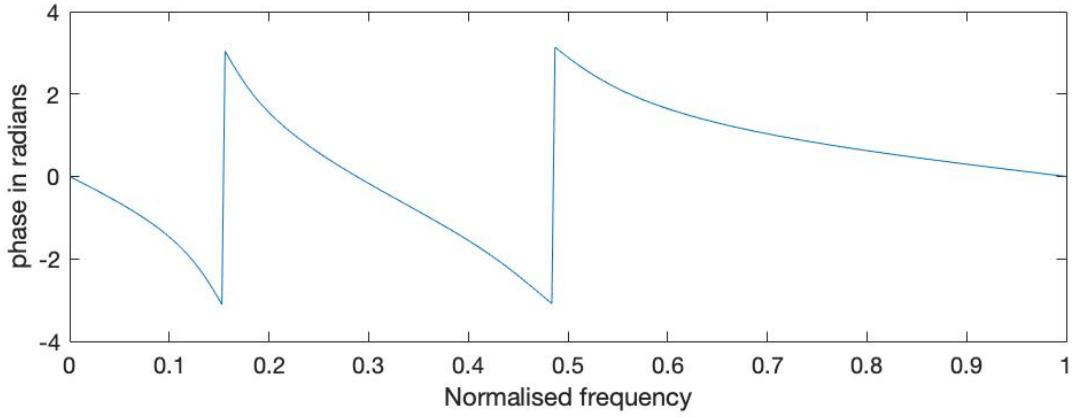
```
b =

   0.2348   -1.1611    3.0921   -5.2573    6.2629   -5.2573    3.0921   -1.1611    0.2348

a =

   1.0000   -3.2803    5.4917   -6.1419    5.0690   -3.0524    1.3002   -0.3622    0.0558
```


0
-100
-200
-300
-400
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


4
2
0
-2
-4
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

Chebyshev Filters (Type I)

LPF:

Code:

```
clear all;
alphap=1;
alphas=15;
wp=0.2*pi;
ws=0.3*pi;
[n,wn]=cheb1ord(wp/pi,ws/pi,alphap,alphas);
[b,a]=cheby1(n,alphap,wn)
w=0:0.01:pi;
[h,ph]=freqz(b,a,w);
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(ph/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(ph/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency');
```

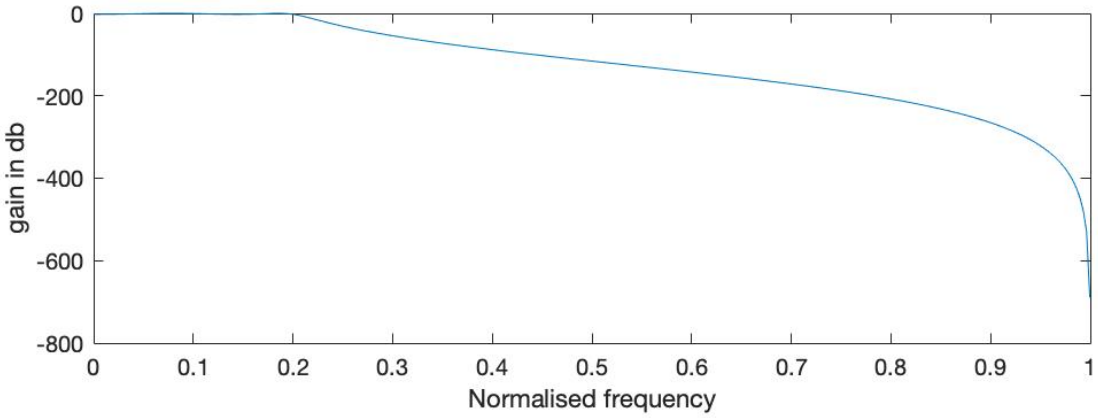
Output:

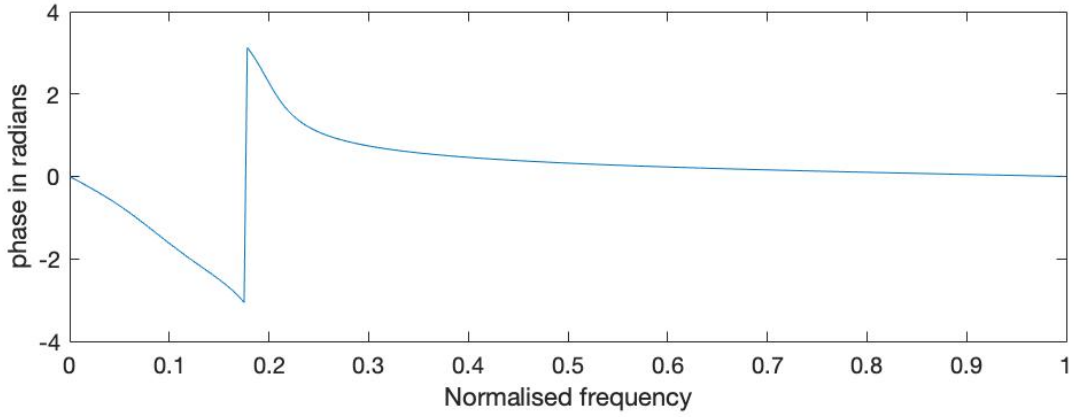
```
b =

   0.0018    0.0073    0.0110    0.0073    0.0018

a =

   1.0000   -3.0543    3.8290   -2.2925    0.5507
```


0
-200
-400
-600
-800
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


4
2
0
-2
-4
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

HPF:

Code:

```
clear all;
alphap=1;
alphas=15;
ws=0.2*pi;
wp=0.5*pi;
[n,wn]=cheb1ord(wp/pi,ws/pi,alphap,alphas);
[b,a]=cheby1(n,alphap,wn,'high')
w=0:0.01:pi;
[h,ph]=freqz(b,a,w);
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(ph/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(ph/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency’);
```

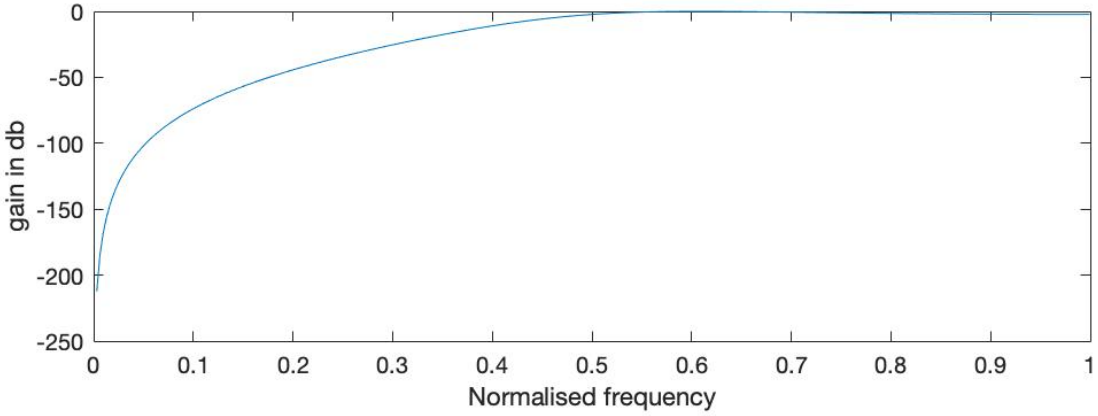
Output:

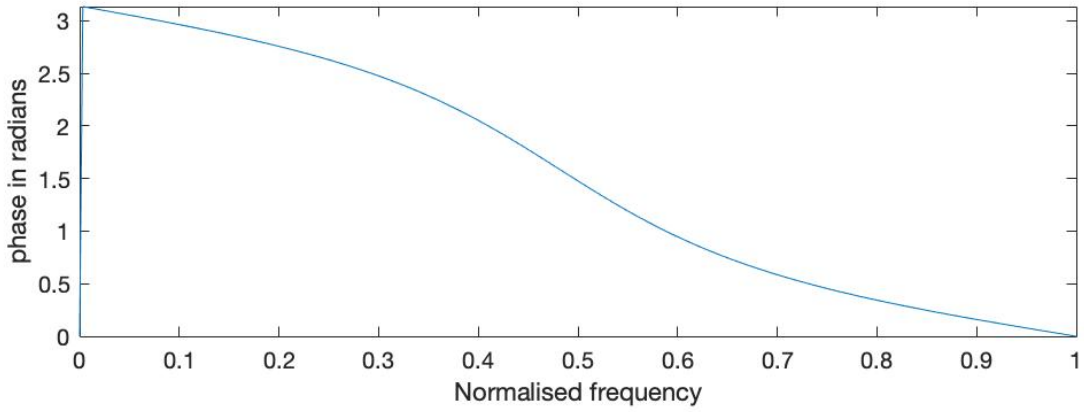
```
b =

   0.3070   -0.6141    0.3070


a =

   1.0000   -0.0641    0.3140
```


0
-50
-100
-150
-200
-250
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


3
2.5
2
1.5
1
0.5
0
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

BPF:

Code:

```
clear all;
alphap=2;
alphas=20;
wp=[0.2*pi,0.4*pi];
ws=[0.1*pi,0.5*pi];
[n,wn]=cheb1ord(wp/pi,ws/pi,alphap,alphas);
[b,a]=cheby1(n,alphap,wn)
w=0:0.01:pi;
[h,ph]=freqz(b,a,w);
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(ph/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(ph/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency’);
```

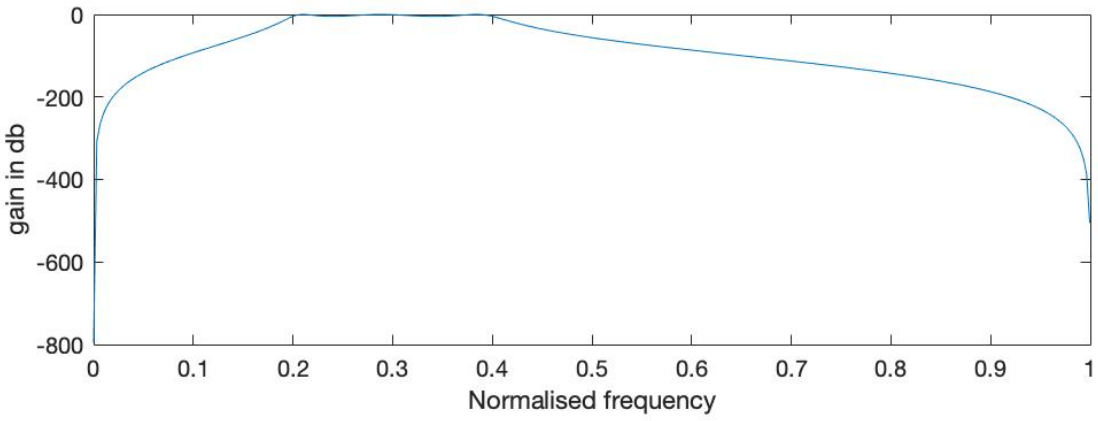
Output:

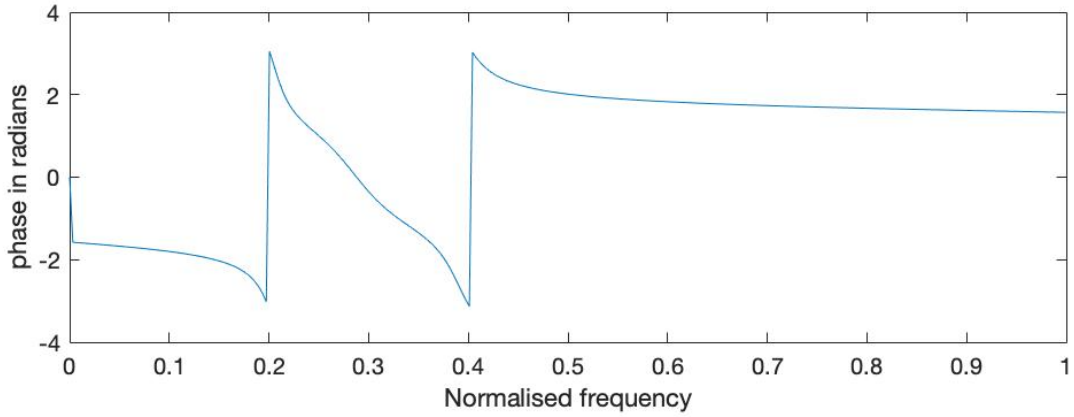
```
b =

   0.0083         0   -0.0248         0    0.0248         0   -0.0083


a =

   1.0000   -3.2632    5.9226   -6.6513    5.0802   -2.3909    0.6307
```


0
-200
-400
-600
-800
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


4
2
0
-2
-4
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

BSF:

Code:

```
clear all;
alphap=2;
alphas=20;
ws=[0.2*pi,0.4*pi];
wp=[0.1*pi,0.5*pi];
[n,wn]=cheb1ord(wp/pi,ws/pi,alphap,alphas);
[b,a]=cheby1(n,alphas,wn,'stop')
w=0:0.01:pi;
[h,ph]=freqz(b,a,w);
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(ph/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(ph/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency');
```

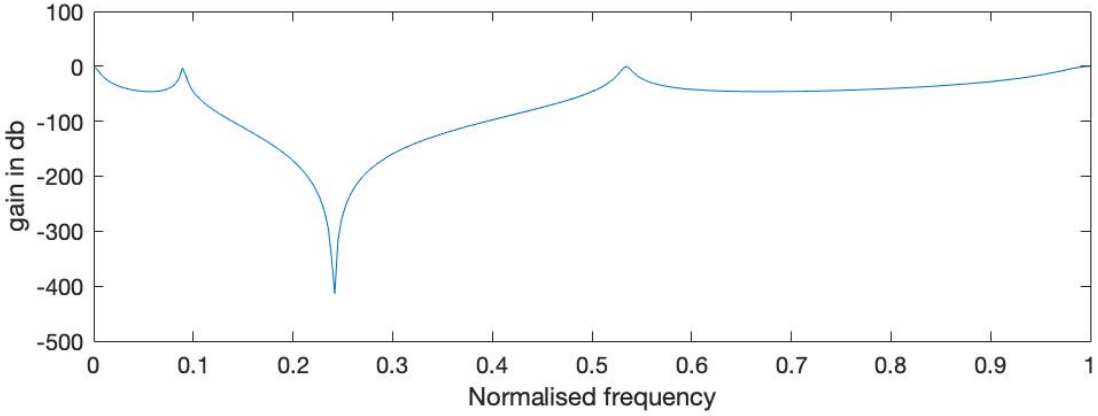
Output:

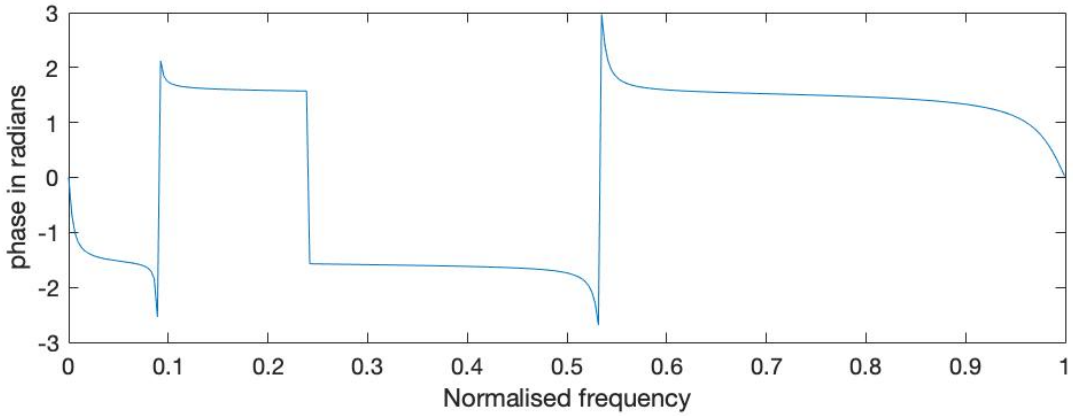
```
b =

   0.0254   -0.1106    0.2368   -0.2990    0.2368   -0.1106    0.0254


a =

   1.0000   -1.7660    0.7564   -0.1954   -0.3541    1.4412   -0.8780
```


100
0
-100
-200
-300
-400
-500
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


3
2
1
0
-1
-2
-3
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

Chebyshev Filters (Type II)

LPF:

Code:

```
clear all;
alphap=1;
alphas=15;
wp=0.2*pi;
ws=0.3*pi;
[n,wn]=cheb2ord(wp/pi,ws/pi,alphap,alphas);
[b,a]=cheby2(n,alphap,wn)
w=0:0.01:pi;
[h,ph]=freqz(b,a,w);
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(ph/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(ph/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency’);
```

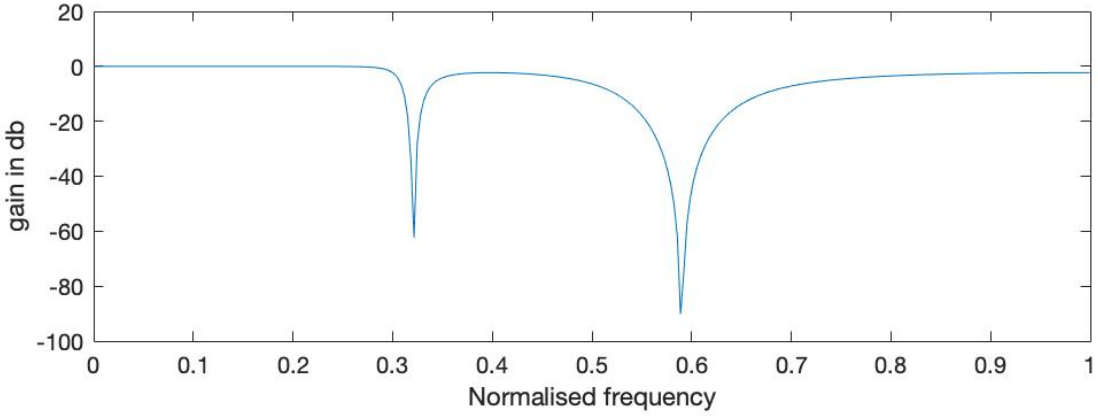
Output:

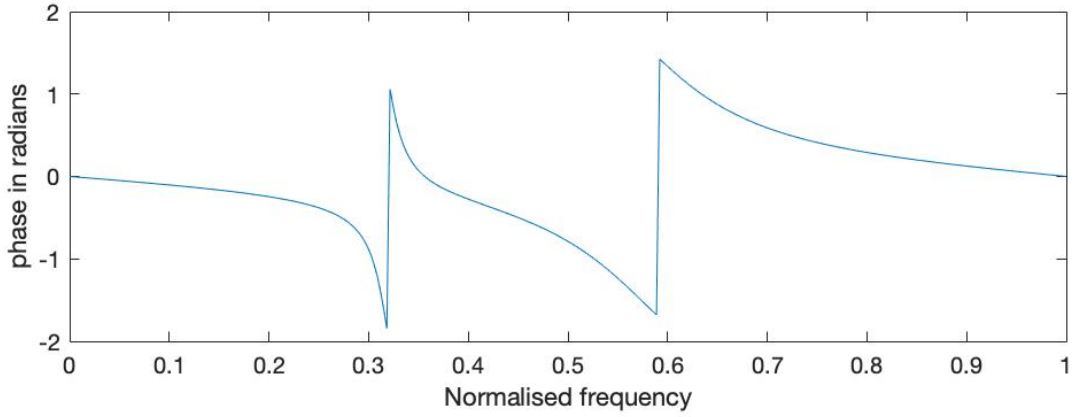
```
b =

   0.7164   -0.3652    1.0067   -0.3652    0.7164


a =

   1.0000   -0.6698    1.1096   -0.2540    0.5234
```


20
0
-20
-40
-60
-80
-100
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


2
1
0
-1
-2
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

HPF:

Code:

```
clear all;
alphap=1;
alphas=15;
ws=0.2*pi;
wp=0.5*pi;
[n,wn]=cheb2ord(wp/pi,ws/pi,alphap,alphas);
[b,a]=cheby2(n,alphap,wn,'high')
w=0:0.01:pi;
[h,ph]=freqz(b,a,w);
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(ph/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(ph/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency’);
```

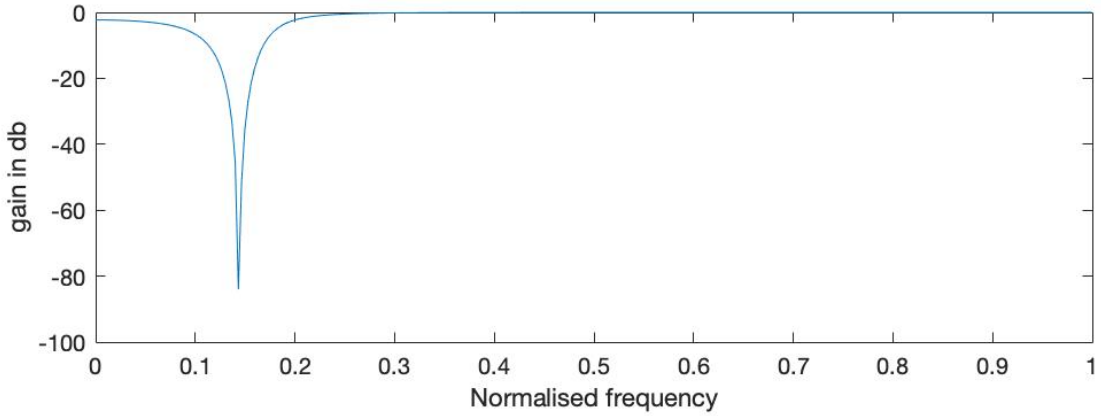
Output:

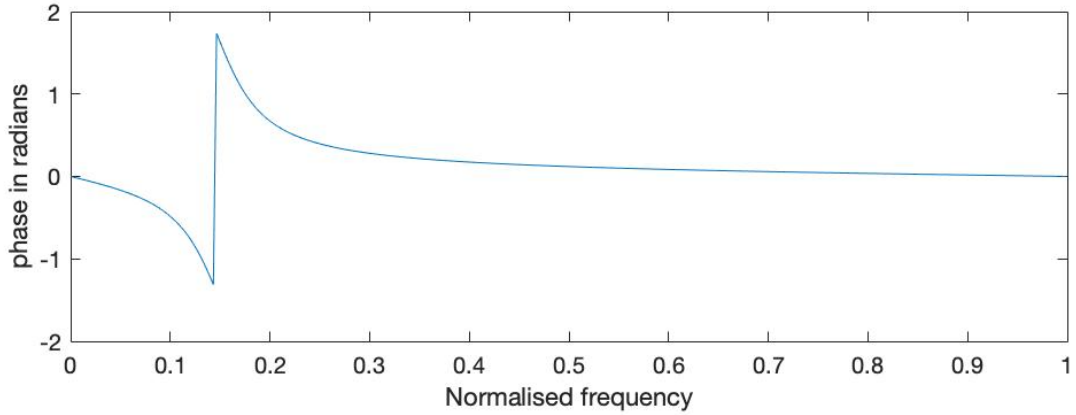
```
b =

   0.8977   -1.6154    0.8977


a =

   1.0000   -1.6044    0.8064
```


0
-20
-40
-60
-80
-100
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


2
1
0
-1
-2
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

BSF:

Code:

```
clear all;
alphap=2;
alphas=20;
ws=[0.2*pi,0.4*pi];
wp=[0.1*pi,0.5*pi];
[n,wn]=cheb2ord(wp/pi,ws/pi,alphap,alphas);
[b,a]=cheby2(n,alphas,wn,'stop')
w=0:0.01:pi;
[h,ph]=freqz(b,a,w);
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(ph/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(ph/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency');
```

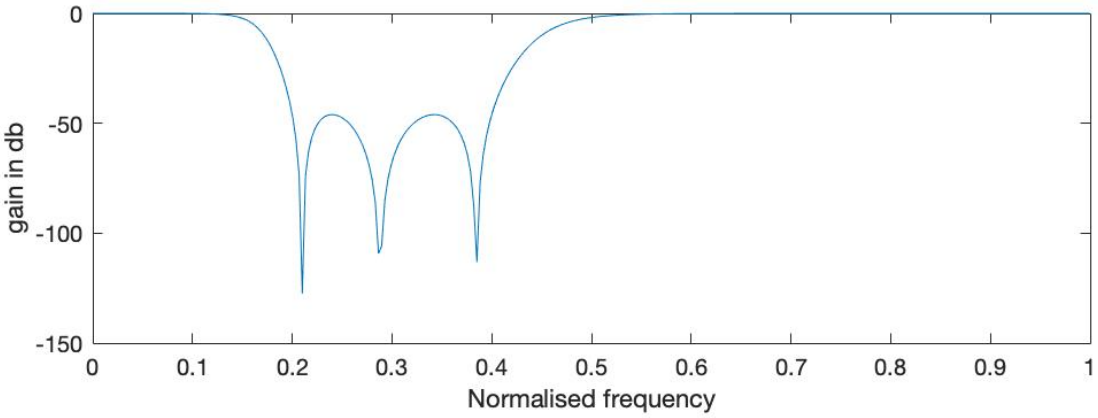
Output:

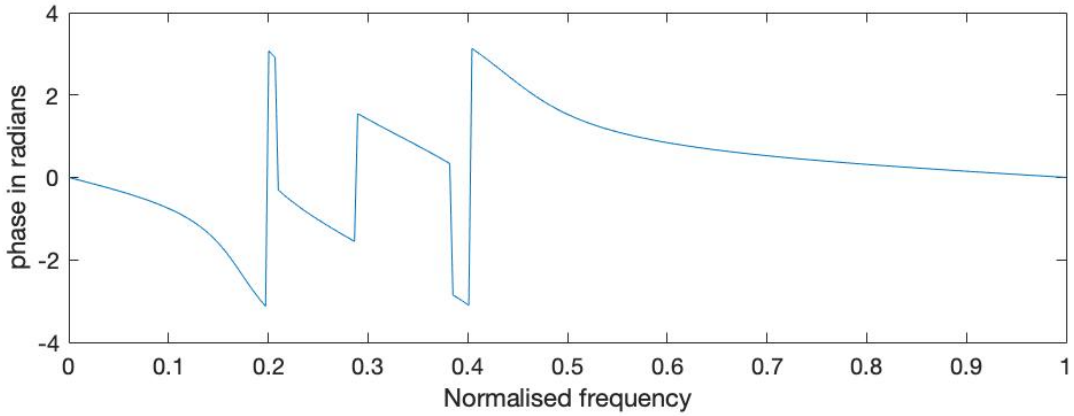
```
b =

   0.4870   -1.7177    3.3867   -4.1110    3.3867   -1.7177    0.4870

a =

   1.0000   -2.7289    4.0090   -3.7876    2.5028   -1.0299    0.2357
```


0
-50
-100
-150
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


4
2
0
-2
-4
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

BPF:

Code:

```
clear all;
alphap=2;
alphas=20;
wp=[0.2*pi,0.4*pi];
ws=[0.1*pi,0.5*pi];
[n,wn]=cheb2ord(wp/pi,ws/pi,alphap,alphas);
[b,a]=cheby2(n,alphap,wn)
w=0:0.01:pi;
[h,ph]=freqz(b,a,w);
m=abs(h);
an=angle(h);
subplot(2,1,1);plot(ph/pi,20*log(m));
ylabel('gain in db');
xlabel('Normalised frequency');
subplot(2,1,2);plot(ph/pi,an);
ylabel('phase in radians');
xlabel('Normalised frequency’);
```

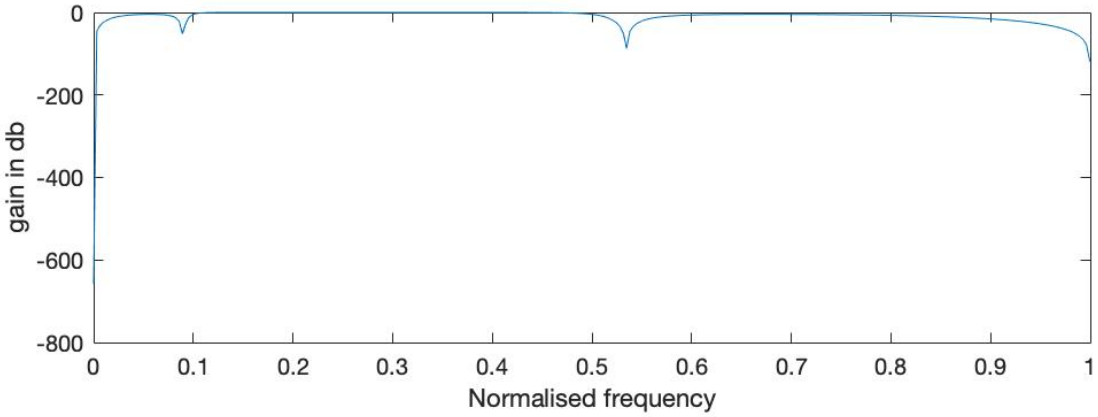
Output:

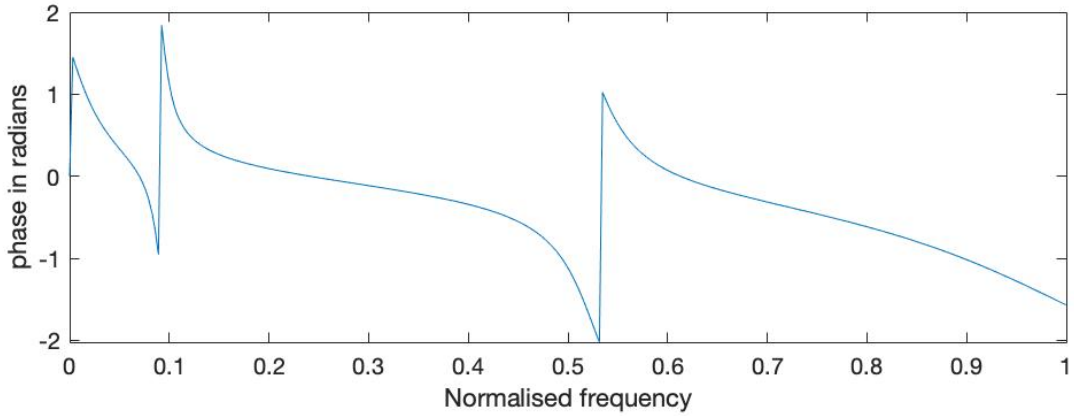
```
b =

   0.6395   -1.0909    0.3754   -0.0000   -0.3754    1.0909   -0.6395


a =

   1.0000   -2.0792    1.6074   -1.0512    0.5064    0.4297   -0.3917
```


0
-200
-400
-600
-800
gain in db
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency


2
1
0
-1
-2
phase in radians
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalised frequency

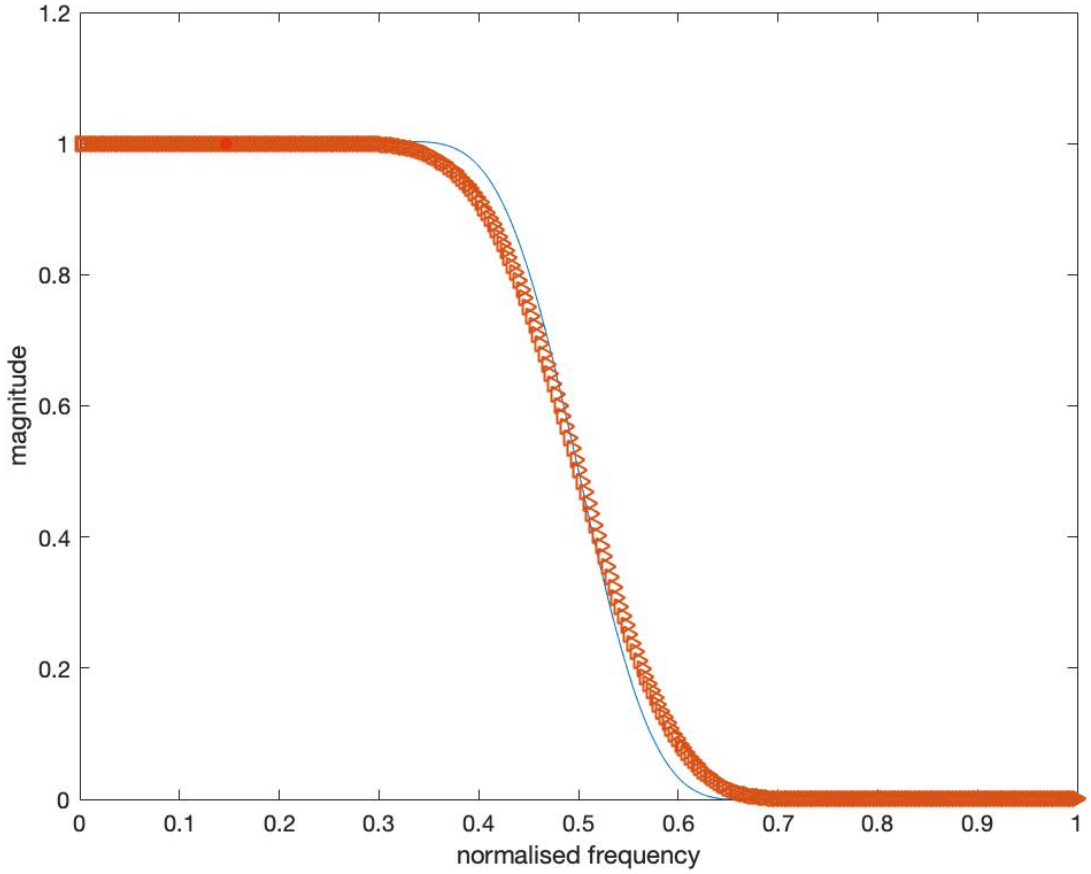
FIR Filters Using Hamming and Blackman Windows

LPF:

Code:

```
clear all;
wc=0.5*pi;
N=25;
b=fir1(N,wc/pi,hamming(N+1));
w=0:0.01:pi;
h=freqz(b,1,w);
plot(w/pi,abs(h));
hold on;
b=fir1(N,wc/pi,blackman(N+1));
w=0:0.01:pi;
h=freqz(b,1,w);
plot(w/pi,abs(h),'>');
xlabel('normalised frequency');
ylabel('magnitude');
hold off;
```

Output:

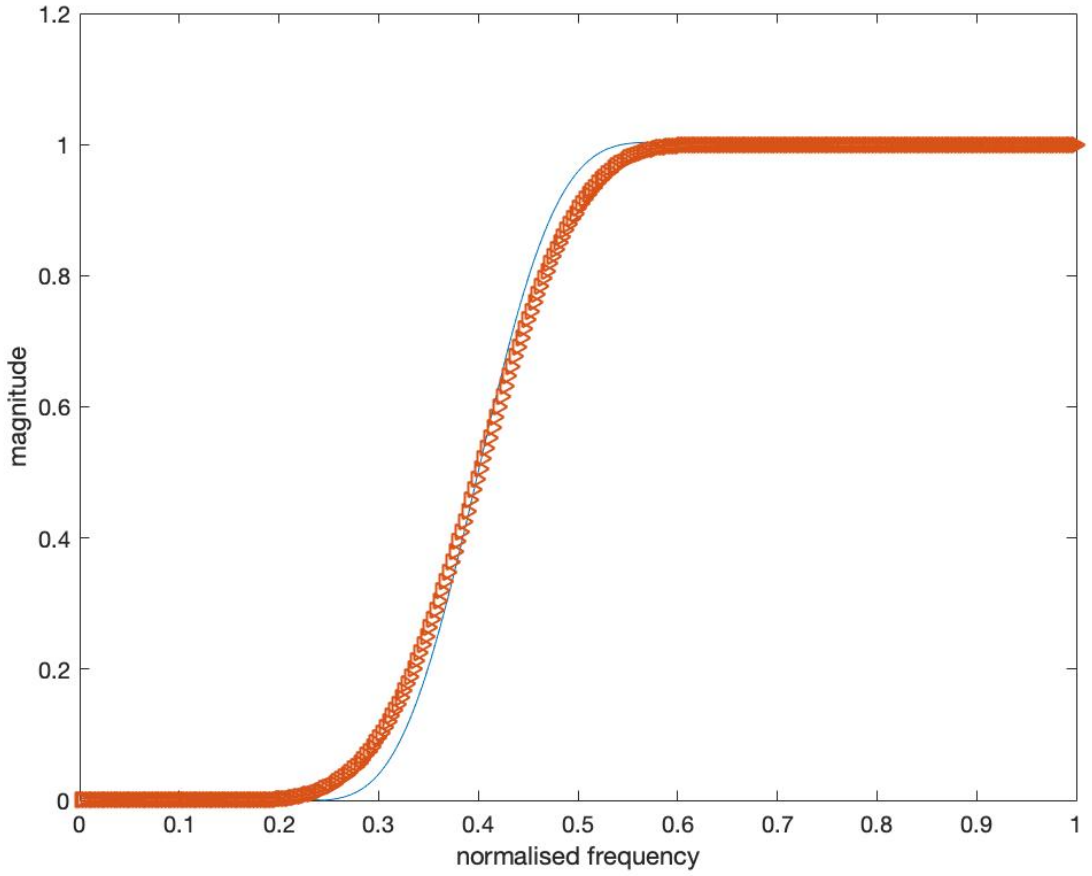

1.2
1
0.8
0.6
0.4
0.2
0
magnitude
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
normalised frequency

HPF:

Code:

```
clear all;
wc=0.4;
N=24;
b=fir1(N,wc,'high',hamming(N+1));
w=0:0.01:pi;
h=freqz(b,1,w);
plot(w/pi,abs(h));
hold on;
b=fir1(N,wc,'high',blackman(N+1));
w=0:0.01:pi;
h=freqz(b,1,w);
plot(w/pi,abs(h),'>');
xlabel('normalised frequency');
ylabel('magnitude');
hold off;
```

Output:

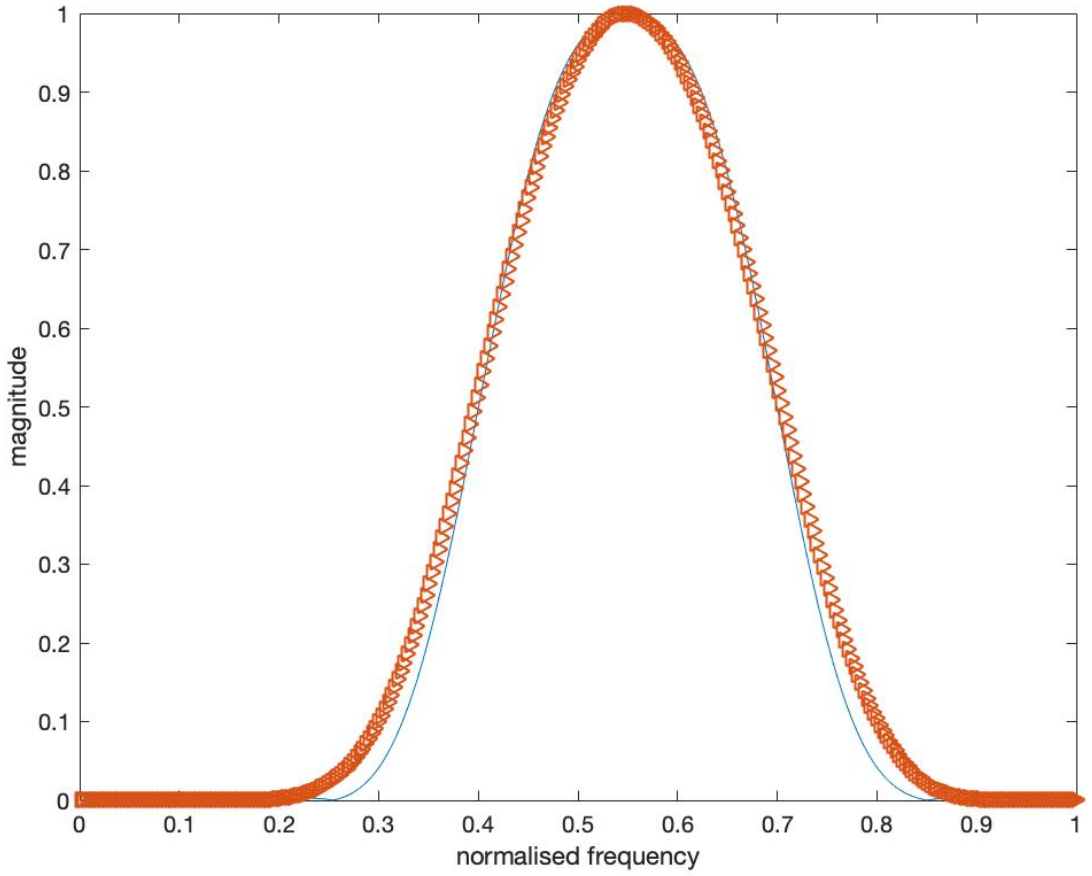

1.2
1
0.8
0.6
0.4
0.2
0
magnitude
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
normalised frequency

BPF:

Code:

```
clear all;
wc=[0.4,0.7];
N=24;
b=fir1(N,wc,hamming(N+1));
w=0:0.01:pi;
h=freqz(b,1,w);
plot(w/pi,abs(h));
hold on;
b=fir1(N,wc,blackman(N+1));
w=0:0.01:pi;
h=freqz(b,1,w);
plot(w/pi,abs(h),'>');
xlabel('normalised frequency');
ylabel('magnitude');
hold off;
```

Output:

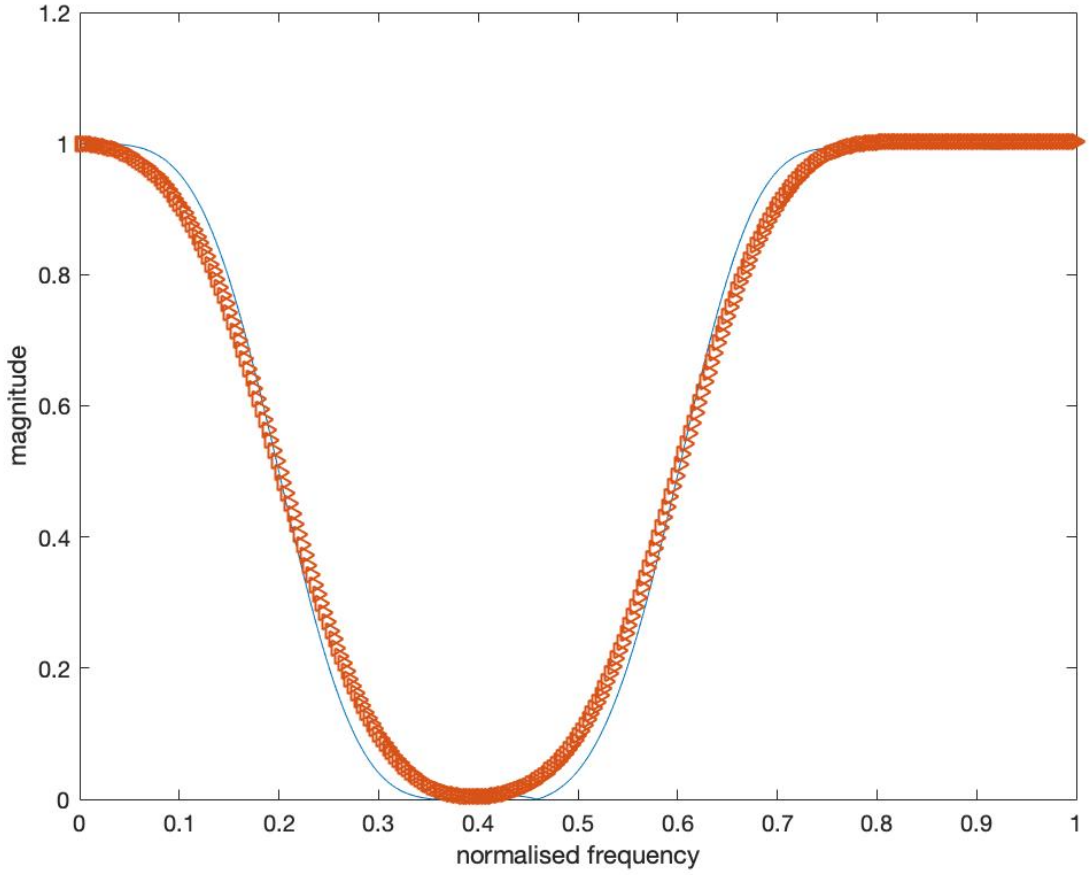


BSF:

Code:

```
clear all;
wc=[0.2,0.6];
N=24;
b=fir1(N,wc,'stop',hamming(N+1));
w=0:0.01:pi;
h=freqz(b,1,w);
plot(w/pi,abs(h));
hold on;
b=fir1(N,wc,'stop',blackman(N+1));
w=0:0.01:pi;
h=freqz(b,1,w);
plot(w/pi,abs(h),'>');
xlabel('normalised frequency');
ylabel('magnitude');
hold off;
```

Output:


1.2
1
0.8
0.6
0.4
0.2
0
magnitude
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
normalised frequency