On homework 2.3a, you are asked to validate that the filter you design meets the specifications. This can be done by visual inspection, but it can be a lot of effort to closely examine the transition regions.

Alternatively, the output of the freqz function can be used to verify the filter. The freqz function returns a (complex) magnitude response vector 'h' and the corresponding vector of frequencies 'f'. If created with sufficient resolution, these vectors can be used to verify if the filter meets a given specification.

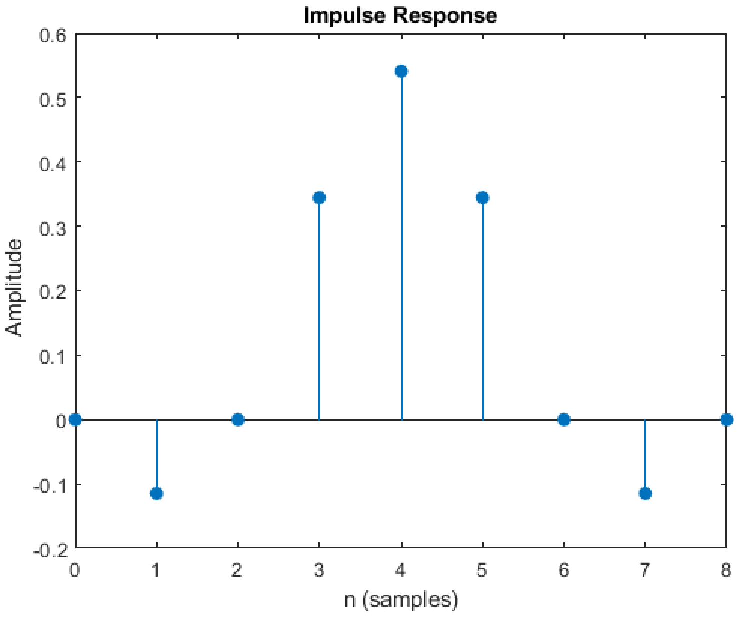
Example:

Suppose that our specification for a lowpass filter is:

- Sampling rate: 16000 Hz
- For frequencies between 0 and 3500 Hz, the passband ripple should be no greater than ± 2 dB
- For frequencies between 4500 Hz and 8000 Hz, the stopband attenuation should be at least 10 dB

We want to check if a truncated sinc meets the specification. The coefficients b for this filter can be computed using the sinc function and normalizing by the magnitude response at 0 Hz:

```
Num = sinc(-2:0.5:2); normalization = 1/freqz(Num,1,1); Num = Num.*normalization;
figure; impz(Num);
```

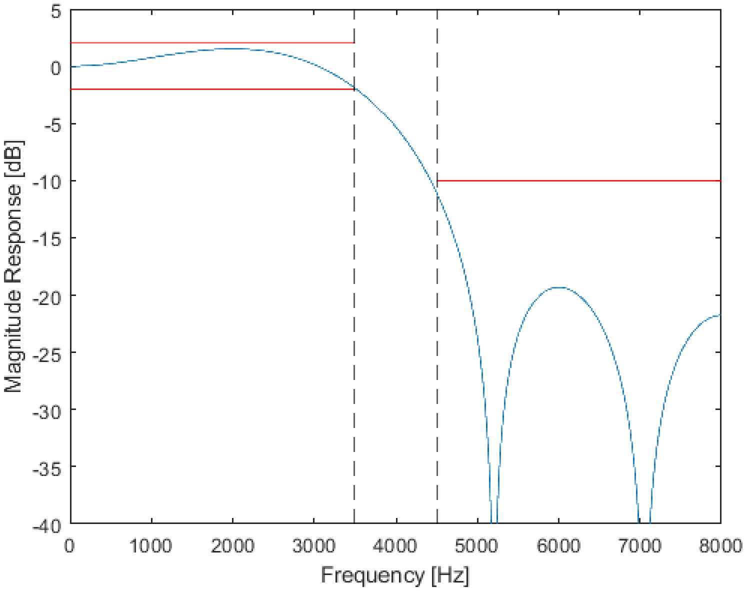


Before using freqz, we need to specify how many points 'n' to sample the DTFT in the interval [0, π). For this example, we will choose n so that we have 1 Hz resolution.

```
fs = 16000;
n = fs/2;
[h,f] = freqz(Num,1,n,fs);
```

```
 dB = 20*log10(abs(h));
 figure; plot(f,dB); xlabel('Frequency [Hz]'); ylabel('Magnitude Response [dB]');
ylim([-40,5]);

 % draw lines corresponding to the specification
 hold on; x = xlim; y = ylim;
 plot([x(1),3500],[2,2],'r'); plot([x(1),3500],[-2,-2],'r');
 plot([4500,x(2)],[-10,-10],'r');
 plot([3500,3500],y,'--k');plot([4500,4500],y,'--k');
```



Visually, it appears to meet the specification, but it appears to be close to violating the passband ripple condition (at 3500 Hz, the red line is very close to touching the blue line.) We can use the vectors h and f returned from freqz to explicitly check the condition at every 1 Hz interval:

```
passband = dB(f<=3500);
passband_spec_failed = any( abs(passband) > 2 );

stopband = dB(f>=4500);
stopband_spec_failed = any( stopband > 10 );

if passband_spec_failed || stopband_spec_failed
    disp('Filter does not meet spec')
else
    disp('Filter meets spec at current resolution')
end
```

```
Filter meets spec at current resolution
```