The purpose of Computer Assignment 3 is to learn about amplitude modulation (AM). Amplitude modulation is achieved by taking an audio signal and multiplying it by a high frequency cosine (the *carrier*). Different signals can each be modulated at unique carrier frequencies and then summed together without interference.

You can learn more about amplitude modulation by reading Section 6-12 of the textbook. Your goal will be (a) to combine two signals by amplitude modulating them, and then (b) use amplitude *demodulation* to pull them back apart. Follow the following steps in Matlab:

1. Create/import/load two audio signals of your choosing. The signals must be between five and ten seconds long and should have *exactly* the same number of samples. The sampling rate for both should be 44,100 samples/second. Call these signals `x` and `y`.
2. Upsample `x` and `y` to a new sampling rate of 384,000 samples per second using these commands:
   ```
   fs = 44100;
   fs2 = 384000;
   x_upsample = resample(x,fs2,fs);
   y_upsample = resample(y,fs2,fs);
   ```
   Upsampling to a higher sampling rate is a necessary step to make all the math work out.
3. Select two *carrier frequencies*. The first one should be in the range of $30 - 50$ kHz and the second one should be somewhere between $70 - 100$ kHz. Call these `f_modx` and `f_mody`.
4. Create a *carrier cosine wave* with frequency `f_modx` and sampling frequency 384,000 samples/sec. This should have exactly the same number of samples as `x`. Call this signal `carrier_x`.
5. Repeat the previous step using `f_mody` to create `carrier_y`.
6. Multiply `x_upsample` with `carrier_x` to create `x_modulated` and `y_upsample` with `carrier_y` to create `y_modulated`
7. Use `myFFT` to compare the Fourier Transform of `x` with that of `x_modulated`. Do the same thing with `y` and `y_modulated`. How has modulating the signal affected its Fourier Transform? Based on your reading in the book, is this what you expect? Why or why not?
8. Add `x_modulated` and `y_modulated` to create your final modulated signal `modulated_signal`. Examine its Fourier Transform. Is it what you expect? If you have done this correctly, you will have effectively combined `x` and `y` into one signal in such a way that we can extract one or the other later on.

The second part of this assignment is to *demodulate* the sounds. You'll start with `modulated_signal` and extract either `x` or `y` and listen to it. This demodulation is made a little easier because you already know the values of `f_modx` and `f_mody`, and you already have the signals `carrier_x` and `carrier_y`.

1. Multiply `carrier_x` with `modulated_signal`. This should have the effect of *frequency shifting* your desired signal down from the carrier frequency into the audio range. Use `myFFT` to convince yourself you've done this properly.

2. Lowpass filter your signal from the previous step so that you just keep the desired demodulated signal and get rid of everything else. You can use `butter` to define filter coefficients for the numerator and denominator and `filter` to apply the filter to the signal. The filter design step can be accomplished as follows:

   ```
   [n, wn] = buttord(f1/(fs2/2),f2/(fs/2),3,60); [b,a] = butter(n,wn);
   ```

   where `f1` and `f2` are the left and right edges of the "transition" band; that is, the filter magnitude response transitions from passband to stopband over the frequency range `f1` to `f2`.

3. Finally, you should downsample your signal from 384,000 samples per second back down to 44,100 (because most computers can't play audio at 384,000). Use the `resample` command to do this. Finally, use `soundsc( xx ,44.1e3)` to listen to your demodulated audio (assuming `xx` is the name of the demodulated signal)

4. Repeat these steps to see if you can extract `y` from `modulated_signal`.

Finally, challenge yourself by seeing if you can use your demodulation steps to extract the three hidden audio clips inside `ca3_AM_Data.mat`. The steps are exactly the same except you'll have to figure out for yourself what the carrier frequencies are.

Write a one-page report in IEEE format explaining what you did, what your results were, and how your various steps compare to what is expected (based on the explanation in the book).

Honors Students should also attempt to demodulate the frequency modulated signal stored in `ca3_FM_Data.mat`. There is a link on Canvas showing where you can read about FM modulation and demodulation. The good news is that only two steps are involved: (1) take the derivative of the FM signal (use the `diff` function in Matlab); and (2) apply an envelope detector (use the `envelope` command).

Work in groups of two, if possible. You may consult with other groups for ideas. However, your methods, analysis, and write-up should be uniquely your groups' own. Place your paper (MSWord format only) plus your Matlab code into a single zip file. Submit the zip file through Canvas. All assignments must be uploaded by 11:59 pm on Tuesday, November 15.