**Tune-Up Tuesday for September 9, 2025**

% Copy this file into a Matlab script window, add your code and answers to the  
% questions as Matlab comments, hit "Publish", and upload the resulting PDF file  
% to this page for the tune-up assignment.  Please do not submit a link to a file  
% but instead upload the file itself.   **Late penalty:** 2 points per minute late.

% (a) Copy, paste and run the Matlab code from lecture slide 1-18 to generate a  
% cosine signal *x*(*t*) = cos(2 pi *f*0 *t*) with *f*0 = 440 Hz and play it as an audio signal  
% for 3s at a sampling rate of *f*s = 8000 Hz. 440 Hz is an 'A' note on the fourth  
% octave on the Western scale. See [Piano key frequencies](https://en.wikipedia.org/wiki/Piano_key_frequencies).

**% (a) ‘A’ note on** [**Western scale**](https://en.wikipedia.org/wiki/Piano_key_frequencies) **in 4th octave (A4) at 440 Hz**

**f0 = 440; % note ‘A4’**

**fs = 8000; % sampling rate**

**Ts = 1/fs; % sampling time**

**t = 0 : Ts : 3; % 3 seconds**

**x = cos(2\*pi\*f0\*t);**

**sound(x, fs);  
pause(3);**

% (b) Add to the code in (a) to generate a new signal  *y*(*t*) = cos(2 pi *f*0 *t*) cos(2 pi *f*1 *t*)  
% with *f*1 = 110 Hz by using the same sampling rate of *f*s = 8000 Hz.  The code on  
% the left side of lecture slide 3-3 might be helpful.  Please remember to use the .\*  
% operator for pointwise multiplication of two vectors.

**% (b) Multiply cosine at 440 Hz and cosine at 110 Hz**

**% Modified code from left side of lecture slide 3-3.**

**f1 = 110; % ‘A’ in the second octave**

**x1 = cos(2\*pi\*f1\*t);**

**y = x .\* x1;**

% (c) Add to the code in (b) to play *y*(*t*) as an audio signal.  
% Describe what you hear.  
% Express y(t) as a sum of two sinusoids.

**% (c) Play y(t) at a sampling (playback) rate of 8000 Hz.**

**% I hear two notes/tones at a lower pitch than 440 Hz.**

**% The product can be written as a sum of two cosines.**

**% Using lecture slide 3-2,**

**% y(t) = 0.5 cos(2 pi 330 t) + 0.5 cos(2 pi 550 t)**

**% Tones at 330 Hz (E 4th octave) and 550 Hz (C# 5th octave)**

**% are harmonics of 110 Hz. See** [**Piano frequencies**](https://en.wikipedia.org/wiki/Piano_key_frequencies)**.**

**sound(y, fs); % soundsc(y, fs) could also have been used.**

**% Note: The product cos(2\*pi\*f0\*t) cos(2\*pi\*f1\*t) will have**

**% amplitudes on [-1, 1] because each cosine has amplitudes**

**% on [-1, 1]. Hence, the sound command can be used without**

**% clipping amplitude values.**

**% Note: Playing a tone at 110 Hz may not be audible when**

**% played back on a laptop. Many laptop playback systems**

**% have very low volume outputs for frequencies below 200 Hz.**

% (d) Plot one period of y(t).  We'll first need to find the period of y(t).  
% The product of two sinusoids with frequencies f0 and f1 produces  
% frequencies at f0+f1 and f0-f1.  You could modify the code from the  
% bottom right side of lecture slide 3-3.

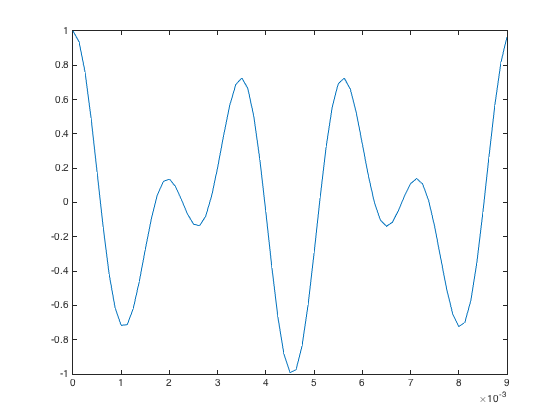
**ffund = gcd(f0+f1, f0-f1); % 440 Hz +/- 110 Hz**

**Tfund = 1/ffund;**

**n = round(Tfund / Ts); % Tfund / Ts isn’t an integer.**

**plot( t(1:n), y(1:n) );**

% See below for the plot. The plot has several artifacts as noted below.



Non-smooth troughs