1. Download the code `Echo.m` from the class website. Note: There is audio involved in this assignment, so make sure to turn down the volume, and please put on headphones if you are in the computer lab!

   (a) Run `Echo.m` in MATLAB. You should hear George Frideric Handel's "Hallelujah" chorus.

   (b) Suppose we want to give the listener the impression that the performance took place in a large hall with an echo. Say we decide that the echo should have a delay of $T = 0.5$ seconds and its amplitude should be $\alpha = 0.7$ of the original sound. In the code, make the following changes

   ```
   T = 0.5;
   alpha = 0.7;
   ```

   and run the code again. You should be able to hear the echo. Feel free to experiment with different values of the delay and the amplitude of the echo.

   Here is an explanation of what the code is doing. The code convolves Handel's music with the function

   $$f(t) \;=\; \delta(t) \,+\, \alpha\,\delta(t-T)$$

   where T is the echo's delay and α is the echo's amplitude. Suppose we call Handel's piece $g(t)$. Then we have

   $$\begin{aligned} g(t) * f(t) \;&=\; g(t) * [\delta(t) \,+\, \alpha\,\delta(t-T)] \\ &=\; g(t) * \delta(t) \,+\, \alpha\, g(t) * \delta(t-T) \\ &=\; g(t) \,+\, \alpha\, g(t-T) \end{aligned}$$

   [Recall that if we convolve a function with a shifted impulse, it results in the function getting shifted by the same amount as the impulse, i.e., $g(t) * \delta(t-T) = g(t-T)$.] The code then plots $f(t)$, $g(t)$, and $y(t) = g(t) * f(t)$, in this order, on a 3-by-1 figure.

   Let us elaborate on the role of each of the MATLAB commands in our code.

   - The lines

     ```
     load handel
     g = y';
     ```

     load Handel's piece into MATLAB and store it in `g`. So $g(t)$ is Handel's "Hallelujah".
   - The lines

     ```
     Delta = 1/8000;
     Pulse = (1/Delta) * ones(1,Delta/t_inc);
     ```

     create a pulse function of width Δ and amplitude $1/\Delta$. Since we want our pulse function to "imitate" an impulse, we take Δ very small. (Due to some numerical issues, we can not take Δ smaller than 1/8000.)
   - The lines

     ```
     ooo = zeros(1,T/t_inc);
     f = [Pulse, ooo, alpha * Pulse];
     ```

     create the function $f(t) = \delta(t) + \alpha\,\delta(t-T)$. Note that `ooo` is a long string of zeros added in between the two pulse functions to create a delay of T seconds.
   - The line

     ```
     y = conv(f,g) * t_inc;
     ```

     convolves $f(t)$ and $g(t)$.

   As we learn more techniques in the weeks to come, we will be able to carry out much more sophisticated manipulations on music and audio files.

2. For the previous problem, run the command

   ```
   sound(g,2*Fs)
   ```

   What you just played was the function $g(2t)$! Now try playing $g(t/2)$ by inputing the command

   ```
   sound(g,Fs/2)
   ```

   You can experiment with other sounds by using

```
load chirp
load gong
load laughter
load splat
load train
```

3. Download the code `Convolution.m` from the class website.

   (a) Run the code. The code convolves the unit width (`Delta = 1`) pulse function $f(t)$ with the exponential function $g(t) = e^{-t}$. Now convolve these same two functions on the JOC website, and compare the result with the result of your code.

   (b) Now set `Delta = 2` and run the code again. Repeat for `Delta = 0.5`.

   (c) What happens as you decrease the value of `Delta`? (Due to numerical issues, you will not be able to decrease `Delta` beyond 0.001.)