Coding Assignment - III

This coding assignment is designed to teach you to think in terms of matrices and vectors because this is how MATLAB organizes data. The other purpose of this homework is to make you comfortable with using **help** to learn about new functions. The names of the functions you'll need to look up are provided in **bold** where needed. Also, in case, recall we cannot use space in the script's name. **Homework must be submitted here: https://mega.nz/filerequest/u6M20A3fei4.**

What to turn in: Your .m file should be renamed to the format firstname_indexnumber. Example justice_181211.m Failure to use the correct format in naming your file will lead to your code not being graded.

a. **Plotting multiple lines and colors.** In class we saw how to plot a single line in the default blue color on a plot. You may have noticed that subsequent plot commands simply replace the existing line. Here, we'll write a script to plot two lines on the same axes.

b. Open a script and name it `twoLinePlot.m`. Write the following commands in this script.

c. Make a new figure using **figure**

d. We'll plot a sine wave and a cosine wave over one period

   i. Make a time vector t from 0 to 2π with enough samples to get smooth lines

   ii. Plot $\sin(t)$

   iii. Type **hold on** to turn on the 'hold' property of the figure. This tells the figure not to discard lines that are already plotted when plotting new ones. Similarly, you can use **hold off** to turn off the hold property.

   iv. Plot $\cos(t)$ using a red dashed line. To specify line color and style, simply add a third argument to your plot command (see the third paragraph of the **plot** help).

   This argument is a string specifying the line properties as described in the help file. For example, the string 'k:' specifies a black dotted line.

d. Now, we'll add labels to the plot

   i. Label the x axis using **xlabel**

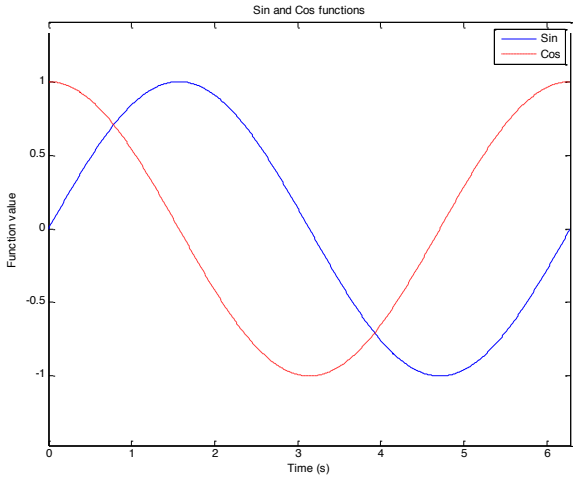
   ii. Label the y axis using **ylabel**

   iii. Give the figure a title using **title**

   iv. Create a legend to describe the two lines you have plotted by using **legend** and passing to it the two strings 'Sin' and 'Cos'.

e. If you run the script now, you'll see that the x axis goes from 0 to 7 and y goes from -1 to 1. To make this look nicer, we'll manually specify the x and y limits. Use **xlim** to set the x axis to be from 0 to 2π and use **ylim** to set the y axis to be from -1.4 to 1.4.

f. Run the script to verify that everything runs right. You should see something like this:


Sin and Cos functions
Sin
Cos
Function value
Time (s)
1
0.5
0
-0.5
-1
0
1
2
3
4
5
6