Math 451 PLOTTING ACTIVITY **Not to turn in**

1. Plot $y = \cos x$ from $x = -5$ to $x = 5$. Create a script called something like *cosx* and keep modifying in the way described.

   (a) Create a vector containing x values from -5 to 5 by halves. Don't suppress the output (don't end with a semicolon), and run your script to make sure it creates the correct vector. Then change it so that it goes from -5 to 5 by tenths, and suppress the output.

   (b) Create another vector y that is $\cos x$. Then plot x versus y. Use `help plot` as needed.

   (c) Note that the top and bottom of the graph are "crowded" by the border. Use the the *axis* command to make the vertical scale go from -2 to 2 to fix this.

   (d) Try labeling the axes and giving your plot a title. Use the *See also* items at the bottom of the `help plot` readout to assist you with this.

   (e) Add a grid to your plot. Then create a black horizontal line for the x-axis. The easiest way I found online (I didn't look too hard for easier ways) is to use the commands

   ```
   x=[xmin,xmax];
   y=[0,0];
   plot(x,y,'k')
   ```

   The `'k'` makes the line black. Add other similar commands to create a black vertical line for the y-axis. Recall that you will need the `hold on` command to get all things you are plotting to show at the same time.

   (f) Add a plot of the sine function, in red.

   (g) Try to make the plots of the functions dashed, dotted or other style lines. Try any other fancy stuff you want - you might wish to create a `legend` that indicates which plot is which function.

2. Sometimes you might want to display two separate graphs at the same time. Basically the way that you do this is to create a "matrix" of graphs. Alter the script from Exercise 1 (maybe save under a new name first) to get the cosine and sine plots on separate grids, one above the other. Use `help subplot` to see how to do this. The instructions aren't super clear, but start experimenting a little and see what happens. Don't fight it too long, though - ask a classmate or me for help if you don't figure it out in a few minutes. Try for side-by-side after you get the one above the other to work.

3. In this exercise you will plot the function $f(x) = 6x - x^2$ and the tangent line to it at $x = 2$. Make a new script file called something like *parabola*.

   (a) Let's plot for x from -1 to 7, so create an appropriate x vector.

   (b) Create a $y = f(x)$ vector - there is an issue we discussed in lecture yesterday that you will have to attend to.

   (c) Plot the function, add some labels, and use the *axis* command to scale the picture so that it is "pleasing" (the parabola should not be to "sharp" or too "flat").

   (d) Write an anonymous function (remember, that's the `f= @(x)` *function* bit) for y, then define $y = f(x)$. This may seem a bit more complicated than necessary, but the function will be useful for the next thing you'll do as well.

   (e) I made the equation we found for a tangent line a little too complicated yesterday. The tangent line to a function f at a point x_0 has equation $y = f(x_0) + f'(x_0)(x - x_0)$. Plot the tangent line to the parabola at $x_0 = 2$.

   (f) If you didn't already, write a function *df* for the derivative of f and use it in your command for creating the tangent line.

4. If you have time, you can now try plotting a three dimensional plot. In this case, let's plot $f(x, y) = x \sin y$ from -4 to 4 for both x and y.

   (a) Instead of creating vectors for x and y we must create a "meshgrid." I don't fully understand all of the subtleties of this, but begin your code with just the line `[X Y]=meshgrid(-4:1:4,-6:1:6)` with no semicolon, because we want to see the output. Run the script to see what it creates.

   (b) Now change the spacing on your meshgrid to tenths and add a semicolon. Enter the function we wish to plot as an anonymous function, and then use it to create a Z (yes, uppercase - I'm not sure it needs to be, but do it for now). Note that the X and Y from the meshgrid are upper case as well.

   (c) Use the command `plot3` to plot X, Y, Z in that order. (Try a different order if you want - I did so by accident and was totally confused!)

   (d) I know pretty much nothing about making this look better. I didn't like the way it looked, and settled on adding the line `mesh(Z);`, which magically made it look better to me.

   (e) From here on out perhaps you can show *ME* something.