Homework 6

Problem 1:

1. Create 3 symbolic equations exactly as they are shown below:
    a. $3x^2 + 7xy + 4 = 0$
    b. $ay + bx = z$
    c. $x - 2y + 2z = 0$
2. Solve each equation for "x" (this will give you equations in term of the other variables)
3. Solve the system of equations for x, y, and z.
4. Substitute a = 4 and b = 0.3 into the solutions using subs( ).
5. Convert the symbolic solutions into decimal numbers (i.e. not fractions) using the double( ) or eval( ) function.

Problem 2:

1. Use poly2sym( ) to create the equation $y = 0.3x^5 - 3x^2 + 4.1x - 5$. Then use diff( ) and int( ) to calculate the first derivative, second derivative, and the integral of the function.
2. Use ezplot( ) to plot the polynomial, the first derivative, the second derivative, and the integral in separate **subplots** with an x range from -4 to 4.
3. Add your name to one of the subplots using the function you created for an earlier assignment.

Problem 3:

1. Import the data from "dotData.txt" into MATLAB.
a. The file contains a large set of numbers with each set containing an x (column 1) and y
(column 2) coordinate and a size value (column 3).
2. Use the **rectangle( )** function to plot solid circles at each x,y coordinate that have a diameter equal to the size value imported from the file.
3. If done correctly, you will recognize the plotted object.
4. Hide the numbers around the perimeter of the plot
5. Add your name somewhere on the figure using your previously created function.
6. Make sure the x and y have the same scale (i.e. a circle will look round instead of like an ellipse if done correctly).
7. Don't do steps 4-6 inside a loop unless you want your code to take a LONG time to run.
8. Modify your code to make the letters in the image red and the rest black.

Problem 4:

1. Animation using **`plot( )`** and **`pause( )`** functions
    a. Create an x vector from 0 to 360 degrees with a stepsize of 10

    b. Create a y vector that is the sin of x. (Hint: use sind() to calculate sine in degrees)

    c. Initialize theta to be 0.

    d. Create a while-loop to do the following for 2 dance cycles:
        i. Create a y_plot vector that equals y times sind(theta)
        ii. Plot the x and y_plot data
        iii. Add your name to the plot
        iv. Make sure the axis doesn't resize every time
        v. Have MATLAB with for 0.1 seconds before continuing
        vi. Increment theta by 10

2. Animation using movie objects
    a. Create a movie object of the dancing sine wave for 2 dance cycles (you can create a new loop or insert commands into the loop you made above). Be sure to include your name somewhere on the plot.

    b. Create an avi using all the frame from your movie object.

    c. Create an animated gif of **ONLY THE FIRST 10 FRAMES** of your movie. If you do all the frames, your gif will be REALLY SLOW.

Problem 5

The following code generates a "kaleidoscope" animation.

```
for k = 5:35
    plot(fft(eye(k)))
    axis([-1 1 -1 1])
    pause(.05)
end
```

Create a MATLAB code to do the following:

1. Prompt the user for a value between 20 and 100. Continue to ask until the user enters a value inside this range.
2. Modify the code above so that it runs forward and then switches direction and runs backward.
    a. This animation portion **MUST** be done inside a **SINGLE** `while` loop statement. Any loop used getting the user's input doesn't count against this constraint.
    b. To accomplish this, **`k`** should increment from 5 up to the number supplied by the user in steps of 2 and then switch directions and increment back down to 5.
    c. Nesting a loop inside another loop still means you are using more than one loop and is NOT a correct solution.
    d. Use `if` statements to control whether k increments up or down.
    e. You may find it helpful to create a variable that changes when the direction changes.
3. Scale the x and y axes so the result is a true circle for all the plots/frames
4. Hide the x and y axes so they aren't visible for all plots/frames

For example, if the user enters 35, you should have about 31 or 32 total plots/frames. See "Kaleidoscope.avi" for an example of what your animation should look like.