CIS 402: Problem Set 1

Directions for Submission

E-mail your answers to me at ajp9@cornell.edu. The subject of your message should be “CIS402 PS1,” and the body of the message should contain your answers. Some mailers can format messages using HTML or RTF. Please turn this feature off and send your message using plain text. If for some reason you cannot send your message as text, you may attach your answers as a text file.

Essential Knowledge–Please give a brief answer (1-2 sentences) for each

1. Type `plot([1,2;1,2;1,2]',[3,4;3,2;2,4]','ro');` at the command line. How many graphics objects does Matlab create? What are they?

   The call to plot creates 3 line objects, one for each column. It also causes a figure and an axes to be created if needed.

2. You just finished making a lovely plot of your data. The figure consists of a single axes where you’ve plotted several data sets, each in a different color, using `plot`. The plot is simple, but because each data set is quite large, it takes a long time to render. When you step back to admire your work, you notice that the points don’t stand out, and you wish you had placed an x at every point. If you had saved the handles to the lines, you could have used set to change the markers. You’re hungry and want to go to lunch and don’t want to replot the whole thing. How could you get the handles and set the markers without replotting? (Hint–it involves a property of the axes).

   The handles for the lines are stored as an array in the axes’ “children” field. To get them, type

   ```
   >> h=get(gca,'children');
   ```

   Note: if you have other objects (lines, patches, text, etc.) in the figure, you will have to find some way to separate their handles from the lines you want to change. The children field acts lack a stack, so the first object you created will be the last one in h.
3. What are the RGB values for the following Matlab colors: 'y', 'm', 'w', and 'k'? (Hint—You can always plot a line using a color and look at its properties)

<table>
<thead>
<tr>
<th>String color</th>
<th>R</th>
<th>G</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>m</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>w</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>k</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Programming: New Markers**

Matlab has several built-in markers for line objects—circles, triangles, squares, and pentagrams to name a few. Unfortunately, there is no way that I can think of to define new marker types. One way around this is to create a function to place an arbitrary polygon at all of the x-y positions in your data. We will do this in stages.

4. An equilateral \( n \)-gon is a polygon with \( n \) vertices and where all sides have the same length and all angles are equal. Write a function called “ngon” that accepts an integer \( n \) and returns the x and y coordinates for the vertices of an equilateral \( n \)-gon. The \( n \)-gon should be centered at the origin and the vertices should fall on a circle of radius 1. Hint: it may be easier to lay out the vertices in polar coordinates and convert to Cartesian.

A sample ngon.m is on the web.

5. Write a function called “polymark” that accepts vectors x and y, an integer \( n \), and a radius \( r \). The function should plot an \( n \)-gon with radius \( r \) at each of the (x,y) points. Your function must return handles to any graphic objects it creates. Note: for the polygons to be symmetric, the x and y axes should use the same units, and you can force this by typing “axis equal.” If you’d like, your function could intelligently scale the \( n \)-gons so that they are symmetric.

A sample polymark.m is on the web. This version creates the polygons as a single object and uses NaN’s to avoid lines linking the polygons.

Bonus. My function returns a separate handle for each of the \( n \)-gons. For a large data set, this is a lot of handles. I will give a cookie (a big,
CTB-sized cookie) the person whose function generates the minimum number of handles.