Extra Credit Points: 20
For Matlab Assignment #4

Instead of having the program print out pairs of values for true anomaly (*T*) and distance (*r*), have the program display a plot of the planet's location at regular time intervals.

To do this, place the following code after your assignment of parameters:

```
set(gca,'DataAspectRatioMode','manual')
hold on
plot(0,0,'*')
```

The first line prevents Matlab from rescaling the plot which would make all ellipses look like circles – a bad thing if you want to see what an orbit looks like. The second line keeps a single plot so that multiple points can be put on it. The third line plots a * at the origin to represent the sun.

Then replace the "disp" command with the following:

```
plot(r*cos(T),r*sin(T),'.g')
```

This plots the planet at the appropriate *x* and *y* coordinates from the sun. The period in '.g' makes the dots big enough to see, and the g makes it green – use '.r' for red dots, '.b' for blue dots, etc.

By running the program again before closing the figure window, you can plot orbits for multiple planets at the same time. Below are some additional data. You can go online to find data for other planets and orbiting objects.

Planet	Eccentricity	Semimajor Axis (Astronomical Units)	Heliocentric Period (Terrestrial days / yrs)
Earth	0.0167175	1.000	365.256 days
Mars	0.0933865	1.489	686.980 days
2003 UB_{313} 10^{th} planet?	0.4378	67.89	557 years
Comet Halley	0.967	17.2	76 years

For the extra points, print out at least three plots that show two orbiting objects each. Label the objects and the time intervals between locations for each object. Then, for each plot, write a sentence or two about something interesting that the plot tells you about the motion of these objects.

For 5 additional points: If you use 2003 UB_{313}, you should notice something funny about the orbit. Identify and fix the problem in your program.