Polar Graphing with *Mathematica*

Don't Forget: *Mathematica* is **case-sensitive** and you must press **Shift-Enter** to execute code.

Basic Graphing

First, you must load the graphics "package" by typing:

<<Graphics`Graphics` and press **Shift-Enter**. There is no output.

(Note: There are no spaces and the backwards apostrophe is found above the Tab key.)

Define the function $r(\ ) = 3\cos(2\ )$ and graph it by typing:

```
r[t_]=3Cos[2t]
PolarPlot[r[t],{t,0,2Pi}]
```

and press **Shift-Enter**.

(Note: Use the underscore _ when *defining* a function and we will use **t** instead of .)

Graphing Multiple Functions

Move the cursor back to the previous part and name the graph as follows:

```
r[t_]=3Cos[2t]
plot1=PolarPlot[r[t],{t,0,2Pi}]
```

and press **Shift-Enter** again.

Now, define and plot a second function $r_2(\ ) = 2\sin(8\ )$ as follows:

```
r2[t_]=2Sin[8t]
plot2=PolarPlot[r2[t],{t,0,2Pi},
     PlotStyle->{Hue[0.6],Thickness[0.02]}]
```

and press **Shift-Enter**.

(Note: You can type the entire "Plot" command on one line in *Mathematica*.)

What did the PlotStyle command do?

Go back to the code for **plot1** and insert

```
,PlotStyle->{Hue[0.3],Thickness[0.03]}
```

just inside the last **]**.

Don't forget the comma and to re-execute by pressing Shift-Enter.

To show both graphs together, type: **Show[plot1,plot2]** and press Shift-Enter.

Type and execute: **Show[plot2,plot1,Axes->False]**

What's the difference?

Type and execute the following:

```
r3[t_]=1+2Sin[2t];
r4[t_]=1+2Cos[2t];
plot3=PolarPlot[{r3[t],-r3[t],r4[t],-r4[t]},{t,0,Pi},PlotStyle->Hue[0]]
```

Show this graph along with the other two.

For a chart color and values for hue, go to: www.abbymath.com – *Mathematica* Files – Graphing – Hue

Saving Graphics

Suggestion: Right-click on your image and choose "Copy As…" then "Metafile" and paste it into a Word document. Now you can size it, save it, and print it more easily.

Partial Graphing

You don't always have to graph from 0 to 2 . For example, type and execute:

```
r[t_]=3Sin[2t];
PolarPlot[r[t],{t,0,Pi/2}];
```

To show the origin at the center of the graph, insert the "PlotRange" command as follows (*don't forget the comma*):

```
PolarPlot[r[t],{t,0,Pi/2},PlotRange->{{-3,3},{-3,3}}]
```

Now change the domain to graph as **t** goes from $\frac{\pi}{2}$ to $\frac{3\pi}{2}$.

Do you remember how to make the graph red?

Assignment

- Design your own polar picture.
- Use at least 4 different equations.
- Write on separate paper each equation that you use along with the domain for .
- Save your picture in a Word file and type your name and period at the top.
- Turn in a printout (color or black & white) and your equation list to your instructor.

Animation for Advanced Users

Type and execute the following:

```
r[t_]=3Sin[6t];
Table[PolarPlot[r[t],{t,0,n*Pi},PlotRange->{{-3,3},{-3,3}},
      PlotStyle->{Thickness[0.01],Hue[n]}],{n,0.1,2,0.1}];
```

Double-click on any image to animate it.

Use the controls at the bottom-left corner of the window to control the animation by slowing it down, speeding it up, pausing, looping, or reversing.

Double-click on the *second* blue bracket on the right-hand side of the window to "stack" the images. You can still double-click to animate, this just saves space.

Change the function to $r(t) = 1 + 3\ \text{Cos}[t]$.

Change the "PlotRange" values so that you can see the entire graph.

Change the value of "Hue" so that it is always red.

To increase the number of frames change `{n,0.1,2,0.1}` to **`{n,0.1,2,0.05}`**.
This part says to create one graph for each value of *n* as *n* goes from 0.1 to 2 counting by 0.05's.