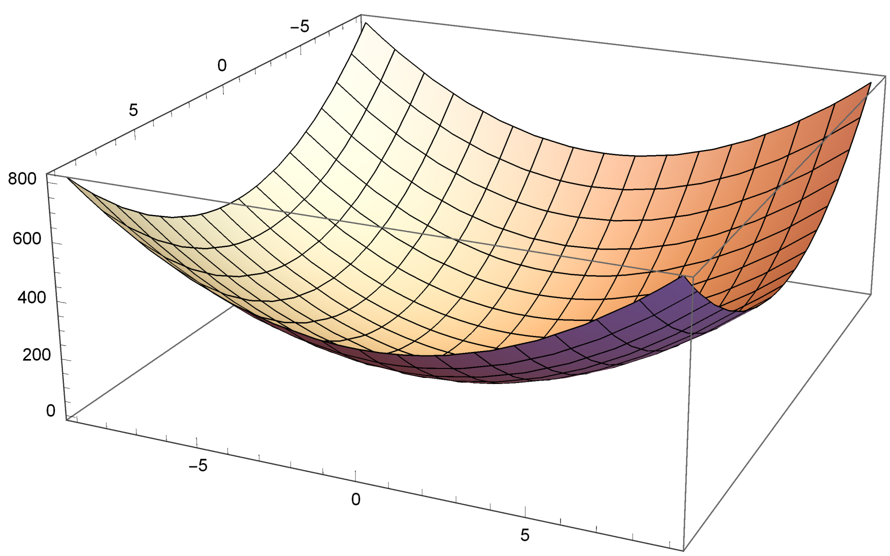
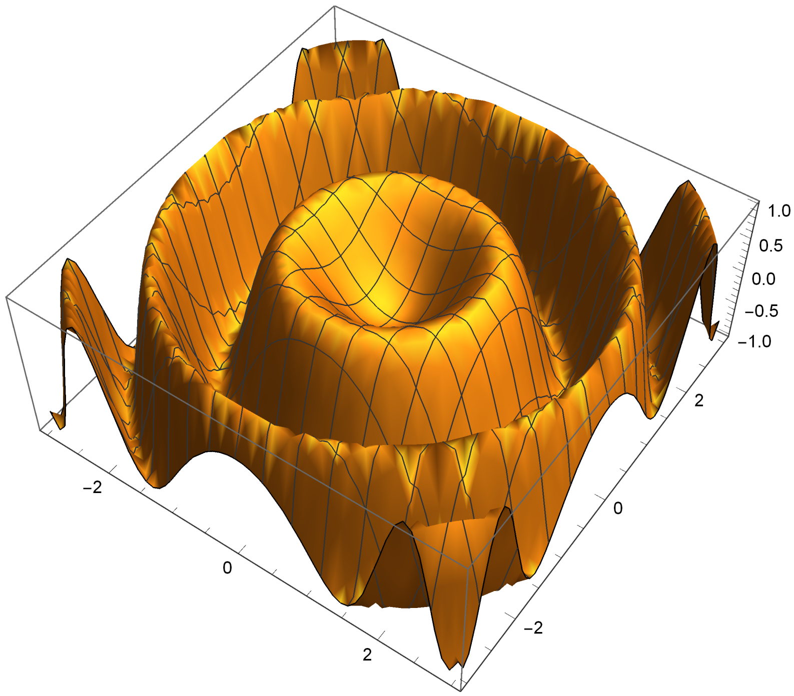
Graphing Curves and Surfaces in Mathematica

Graphing functions of the form z = f (x, y)

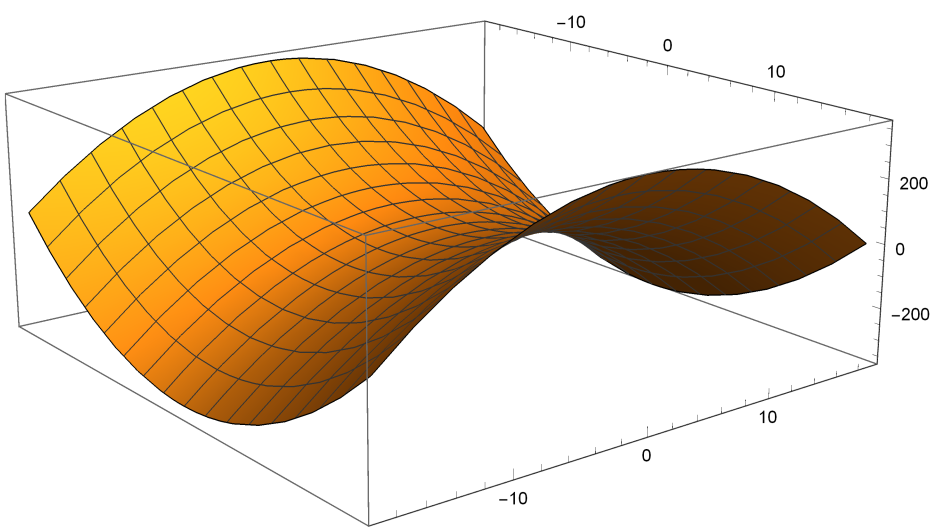
```
Plot3D[5 x^2 + 5 y^2 + 1, {x, -9, 9}, {y, -9, 9}]
```



```
Plot3D[Sin[x^2 + y^2], {x, -3, 3}, {y, -3, 3}]
```

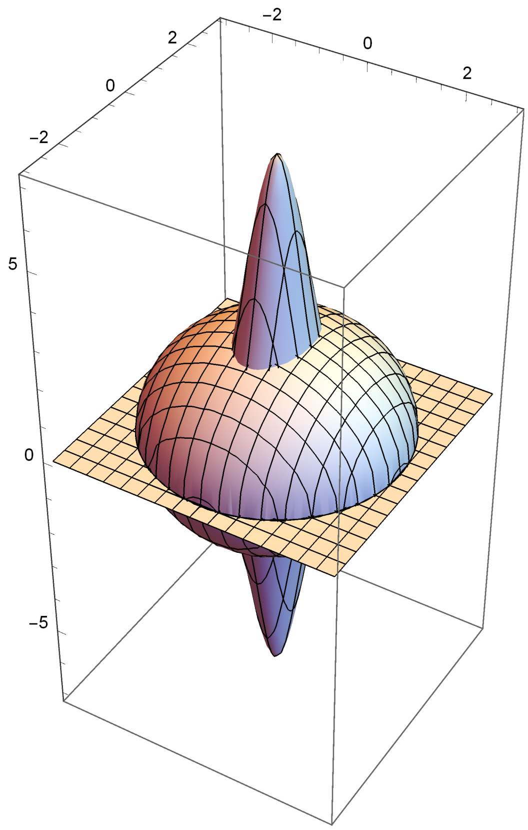



```
Plot3D[x^2 - y^2 + 1, {x, -19, 19}, {y, -19, 19}]
(This is called a saddle.)
```

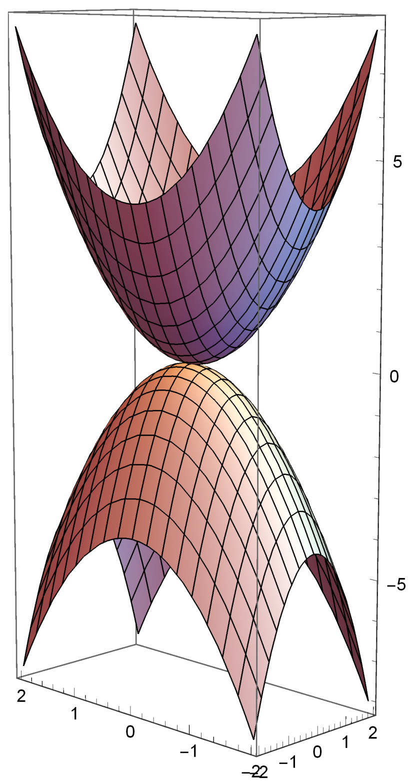


Plotting several surfaces on the same set of axes.

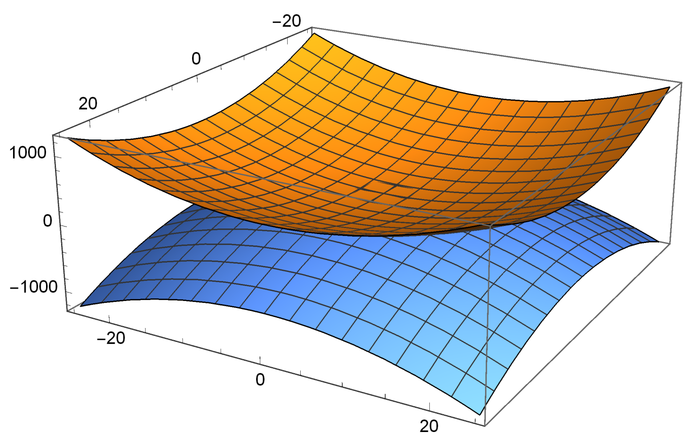
```
Plot3D[{Sqrt[7 - x^2 - y^2], -Sqrt[7 - x^2 - y^2], 7 Exp[-2 x^2 - y^2],
  -7 Exp[-2 x^2 - y^2]}, {x, -3, 3}, {y, -3, 3}, BoxRatios → {1, 1, 2}]
```



```
Plot3D[{x^2 + y^2, -x^2 - y^2}, {x, -2, 2}, {y, -2, 2}, BoxRatios → {1, 2, 4}]
```



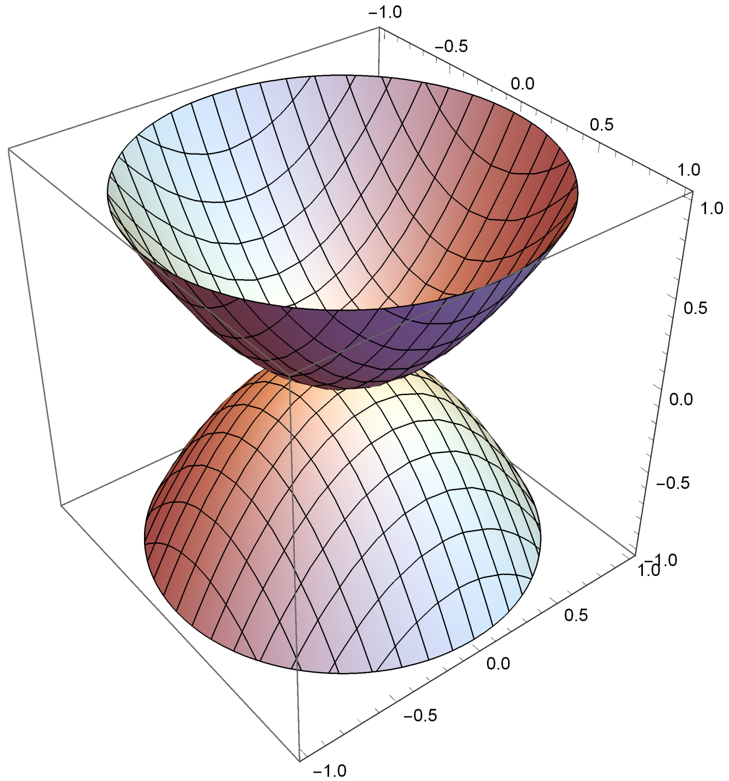

```
Plot3D[{x^2 + y^2, -x^2 - y^2}, {x, -25, 25}, {y, -25, 25}]
```



restricting the domain of the surface

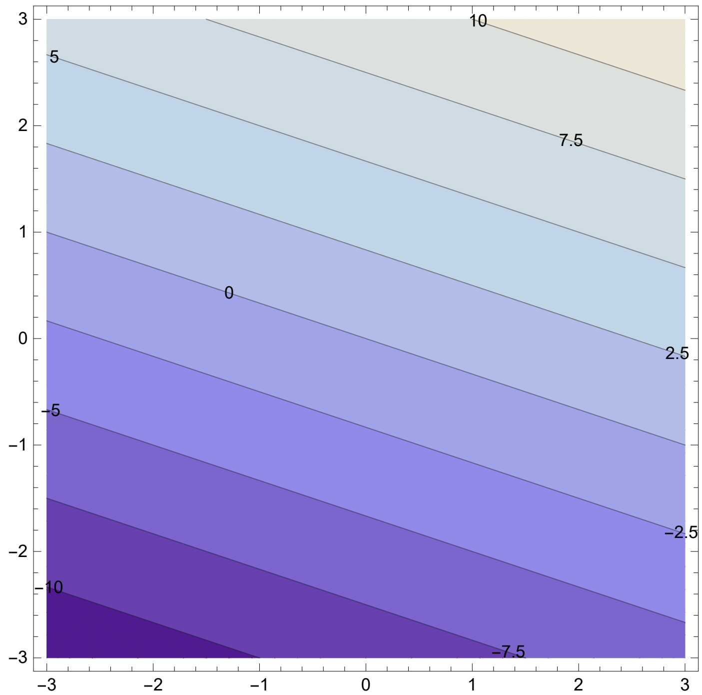
domain of restricting surface is a disk of radius 1 centered at the origin

```
Plot3D[{x^2 + y^2, -x^2 - y^2}, {x, -1, 1}, {y, -1, 1},
 BoxRatios → Automatic, RegionFunction → Function[{x, y, z}, x^2 + y^2 ≤ 1]]
```

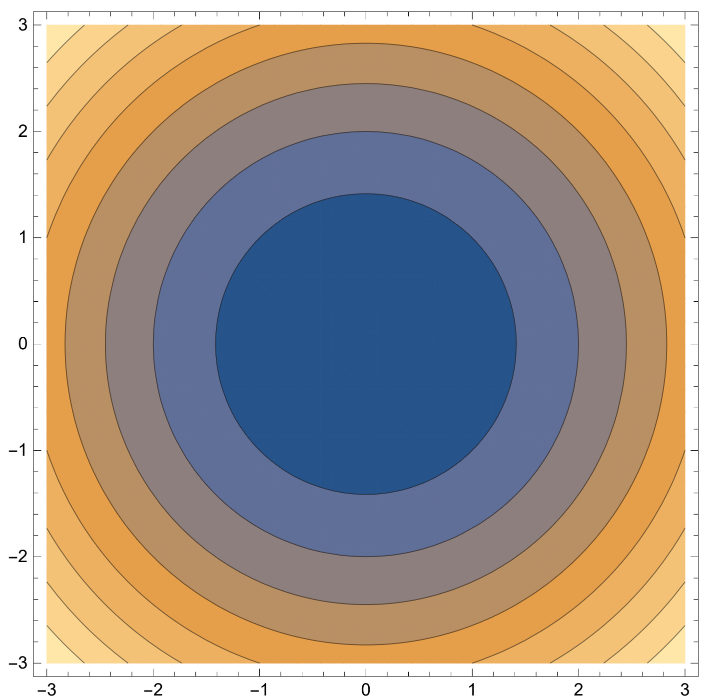


Level Curves

```
ContourPlot[x + 3 y, {x, -3, 3}, {y, -3, 3}, ContourLabels → True]
```



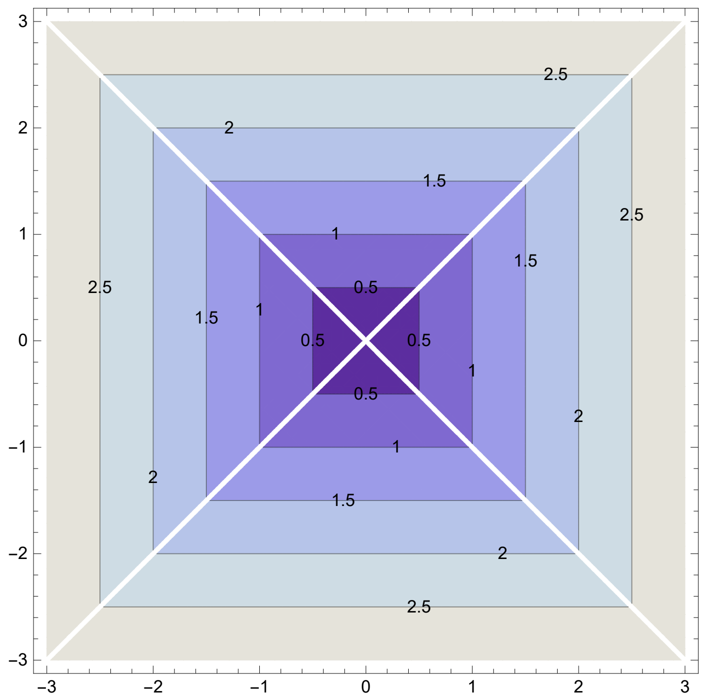

```
ContourPlot[x^2 + y^2, {x, -3, 3}, {y, -3, 3}]
```



Notice that as you move the cursor over a level curve (in Mathematica, but perhaps not on this webpage), you are given the c – value of the level curve.

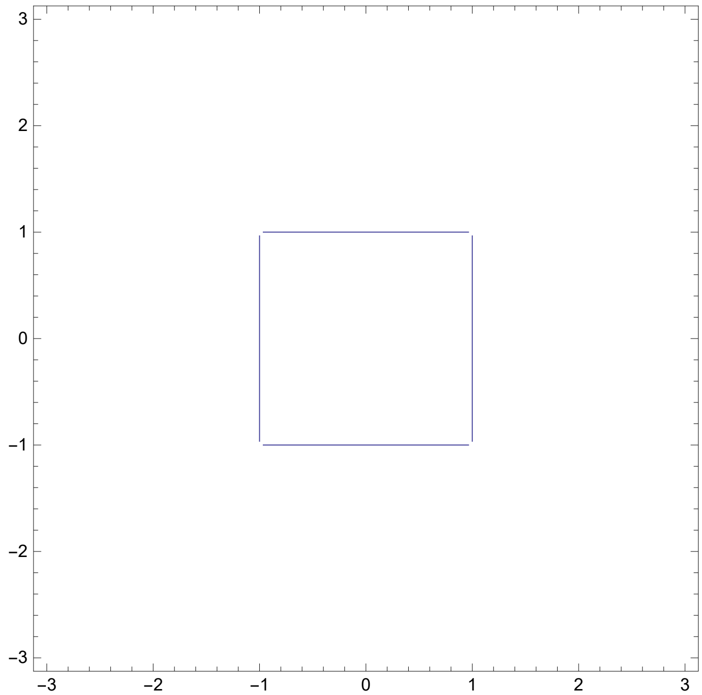
Here is a more unusual example :

```
ContourPlot[Max[Abs[x], Abs[y]], {x, -3, 3}, {y, -3, 3}, ContourLabels → True]
```



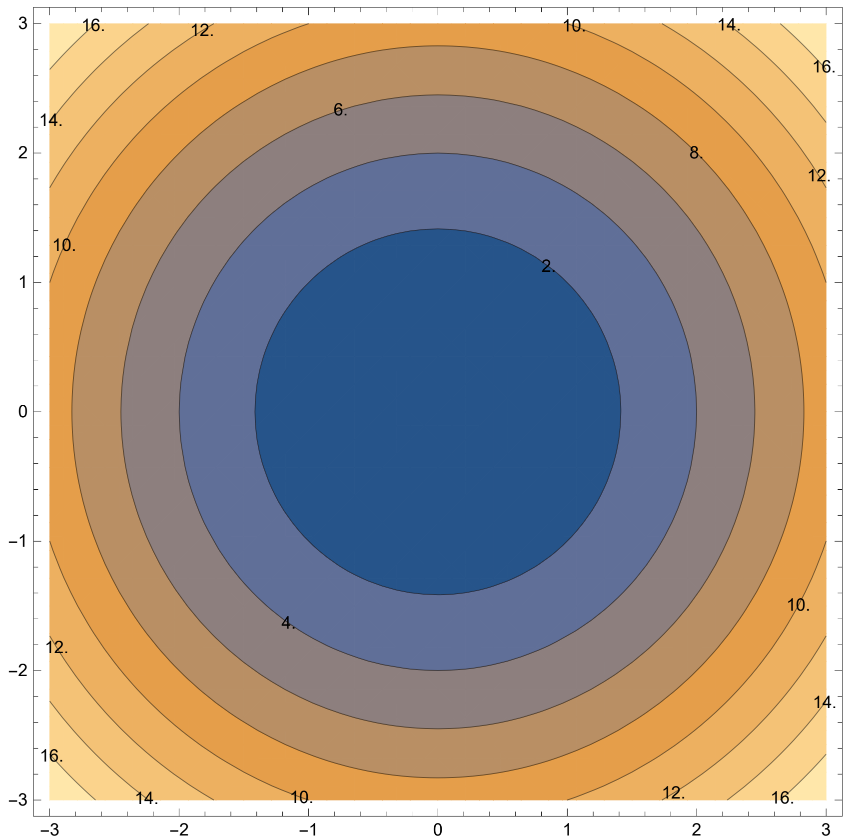
Next, to plot a particular level curve,
we need to use the == symbol. (Note == is double equal sign.)

```
ContourPlot[Max[Abs[x], Abs[y]] == 1, {x, -3, 3}, {y, -3, 3}]
```



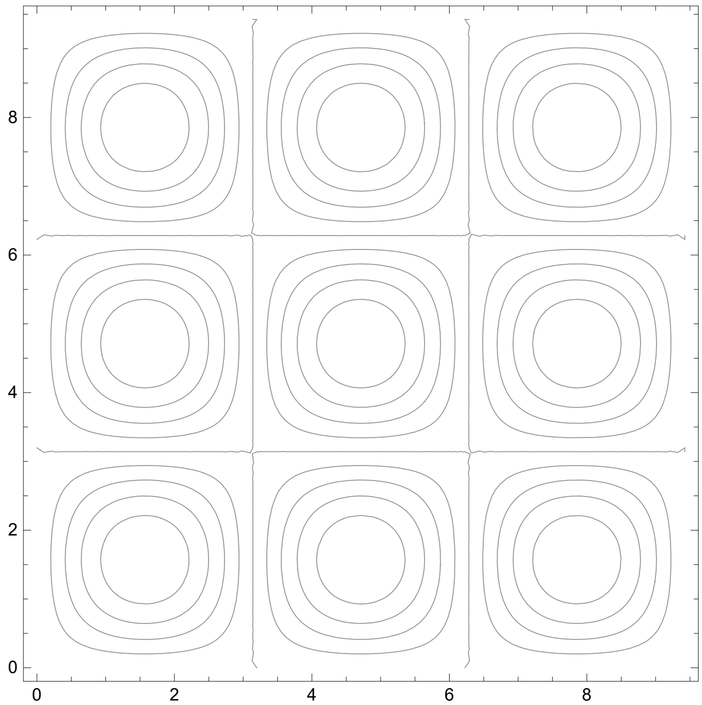
To have the contour values printed on the contour diagram, we may use the ContourLabels command :

```
ContourPlot[x^2 + y^2, {x, -3, 3}, {y, -3, 3}, ContourLabels → True]
```



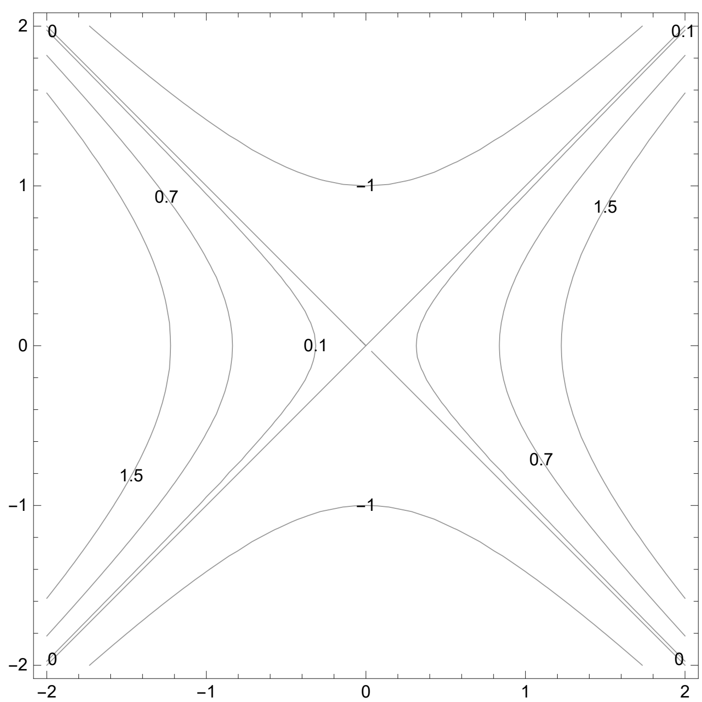
To plot the curves without the default shading, use the ContourShading parameter :

```
ContourPlot[Sin[x] Sin[y], {x, 0, 3 Pi}, {y, 0, 3 Pi}, ContourShading → None]
```

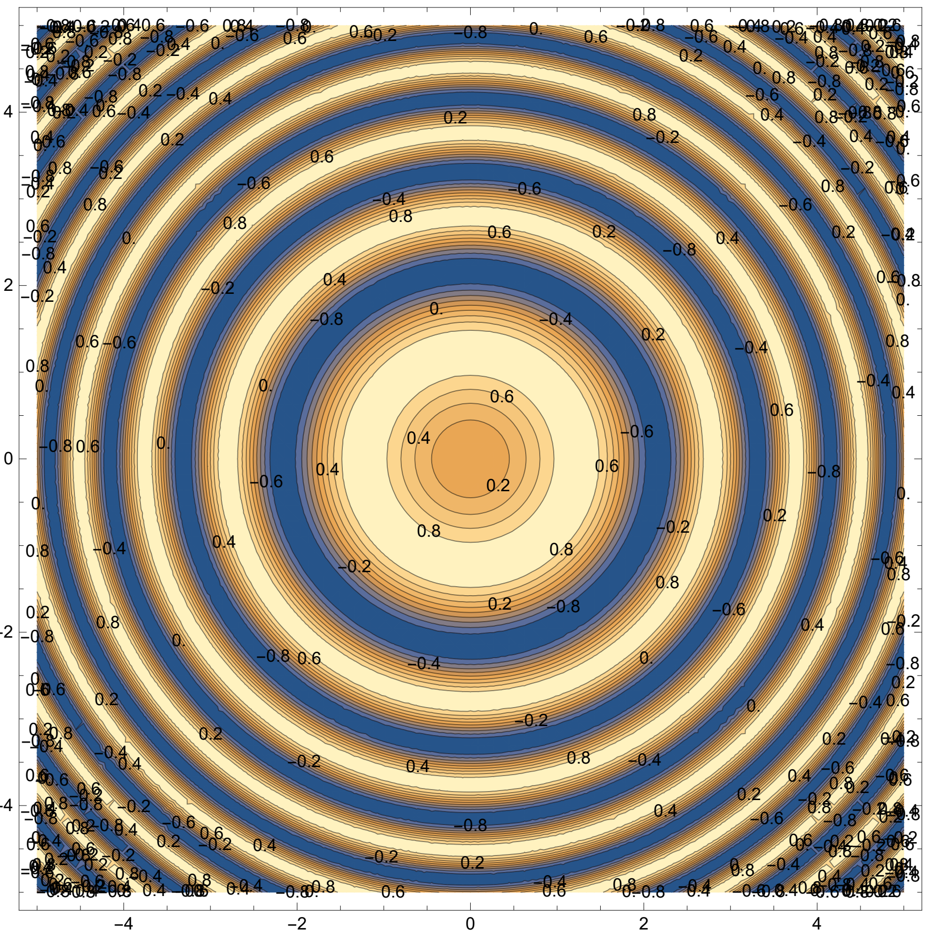


To specify an explicit set of contours :

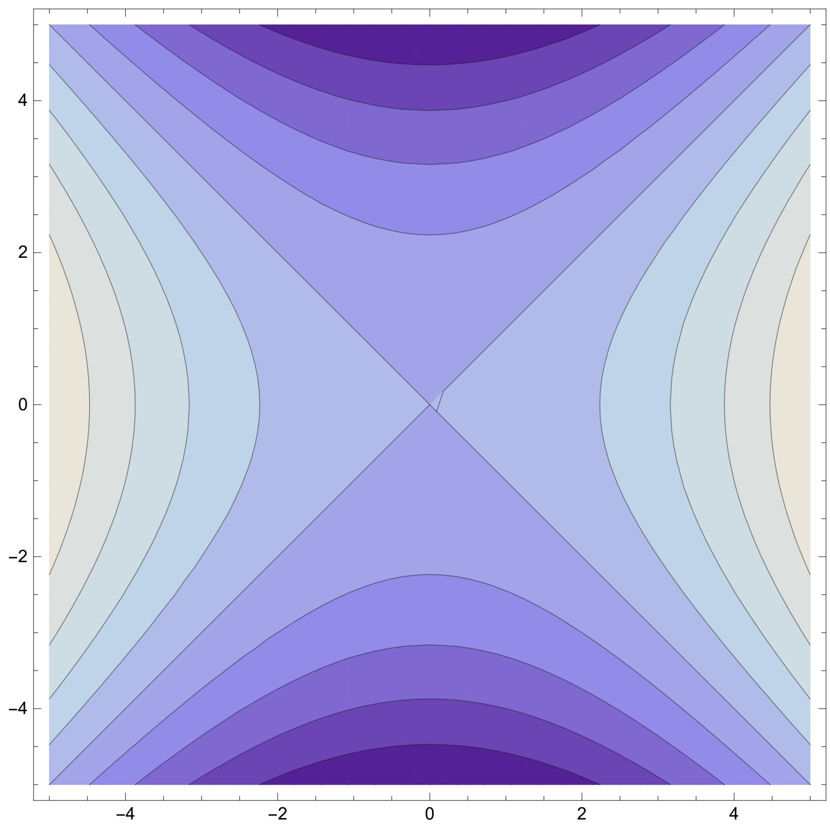
```
ContourPlot[x^2 - y^2, {x, -2, 2}, {y, -2, 2},
 Contours → {-1, 0, 0.1, 0.7, 1.5}, ContourShading → None, ContourLabels → True]
```



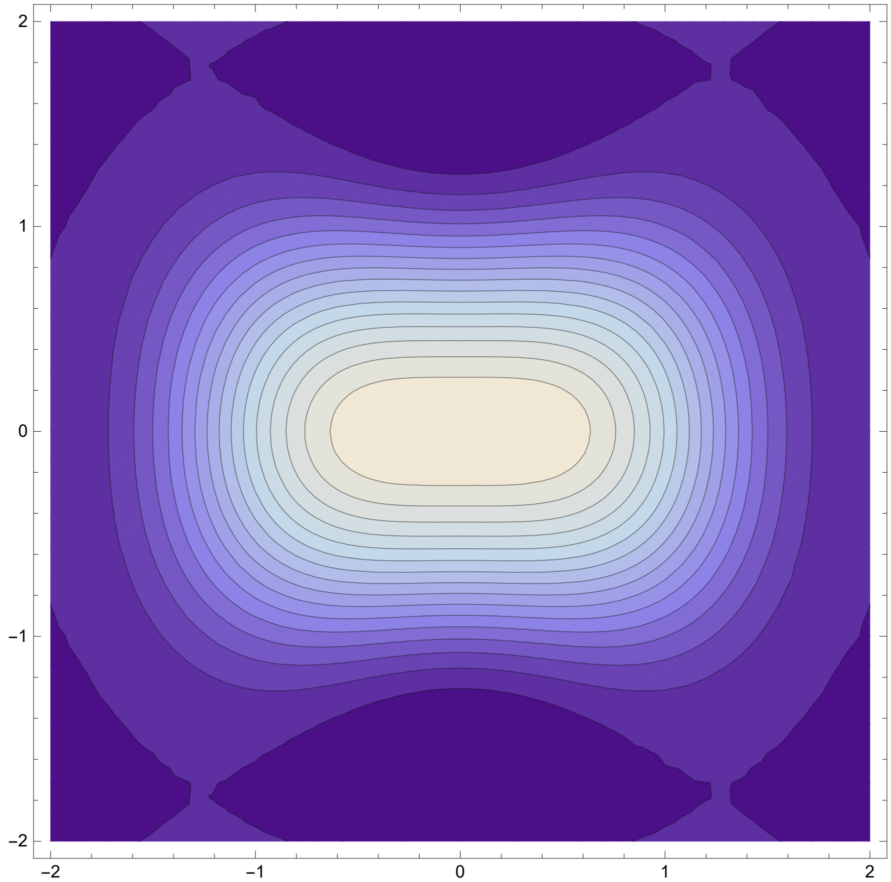
```
ContourPlot[Sin[x^2 + y^2], {x, -5, 5}, {y, -5, 5}, ContourLabels → True]
```



```
ContourPlot[x^2 - y^2, {x, -5, 5}, {y, -5, 5}]
```



```
ContourPlot[Exp[-x^2 - y^2] (Sin[x^2] + Cos[y^2]), {x, -2, 2}, {y, -2, 2}, Contours → 16]
```

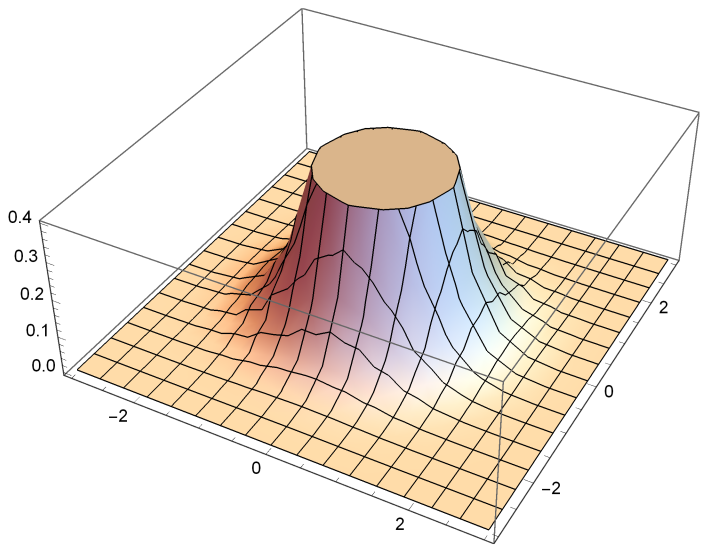


Avoiding Truncation

Avoiding Truncation

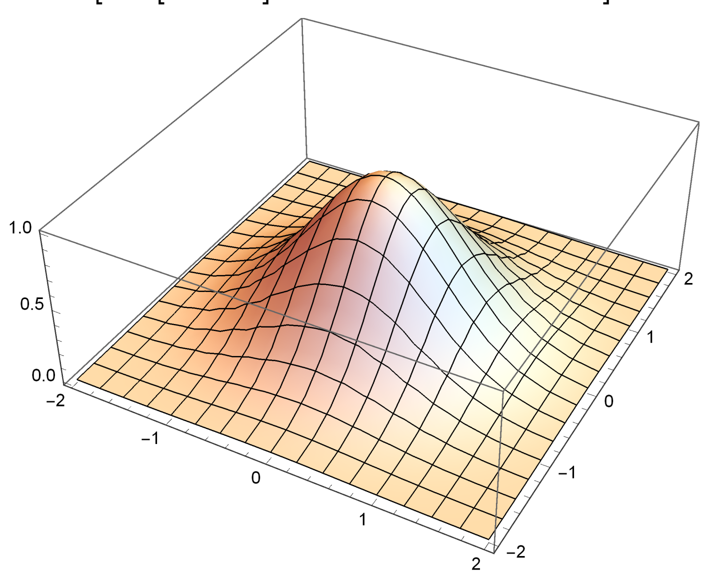
Consider the following attempt at graphing :

```
Plot3D[ Exp[-x^2 - y^2], {x, -3, 3}, {y, -3, 3}]
```



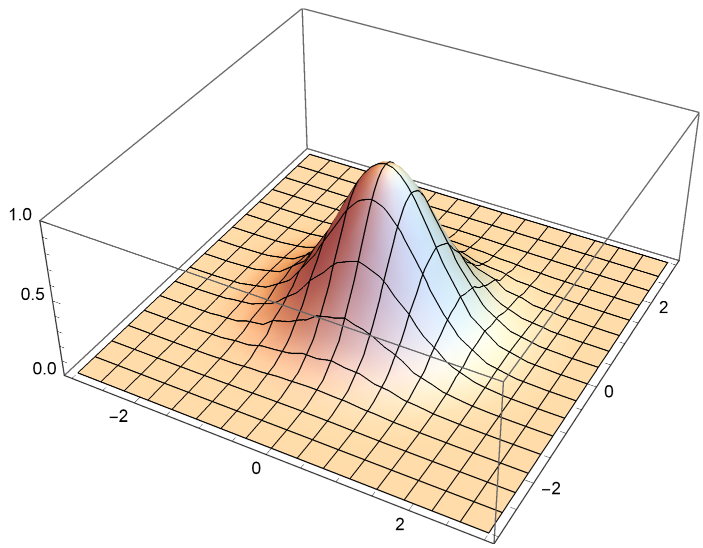
Notice that, although the surface peaks at the origin with a value of 1, the surface is truncated. There are two ways to address this problem. The first is to change the x and y domains.

```
Plot3D[ Exp[-x^2 - y^2], {x, -2, 2}, {y, -2, 2}]
```



The second is to use PlotRange :

```
Plot3D[Exp[-x^2 - y^2], {x, -3, 3}, {y, -3, 3}, PlotRange → {0, 1}]
```

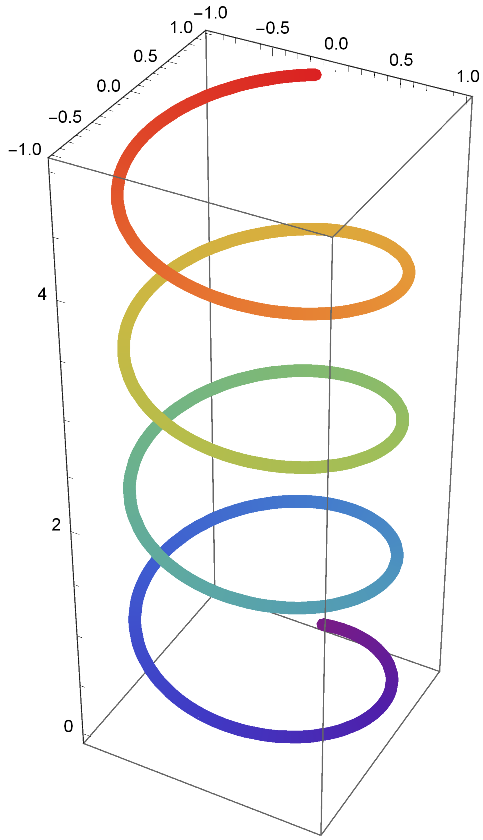


Graphing Parameterized Curves in two or three dimensions

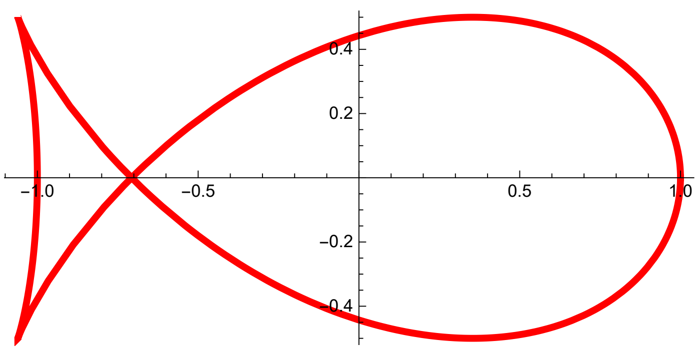
Curves dimensions Graphing in or Parameterized three two

Study the following three examples of parameterized curves :

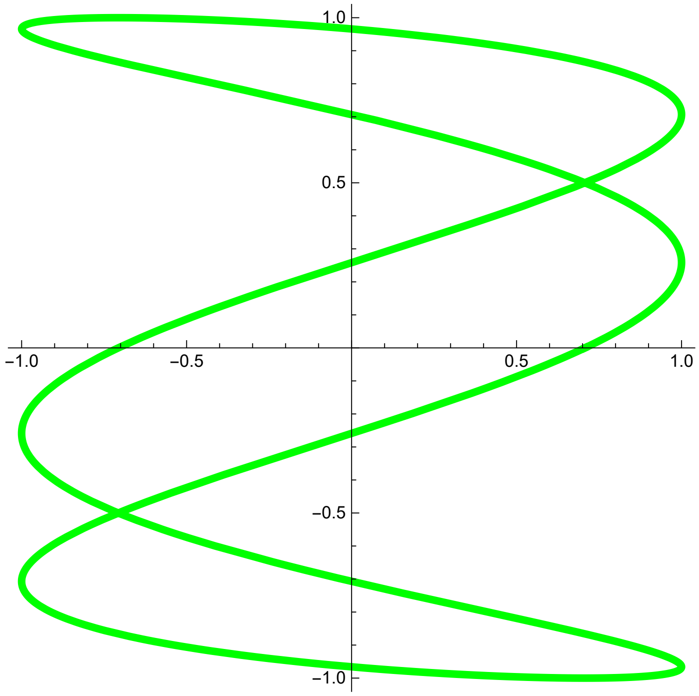
```
ParametricPlot3D[{Sin[5 u], Cos[5 u], u}, {u, 0, 5},
 ColorFunction → "Rainbow", PlotStyle → Thickness[0.03]]
```




```
ParametricPlot[{Cos[t] - Sin[t]^2/√2, Cos[t] Sin[t]},
 {t, 0, 2 π}, PlotStyle → {Red, Thickness[0.01]}]
```



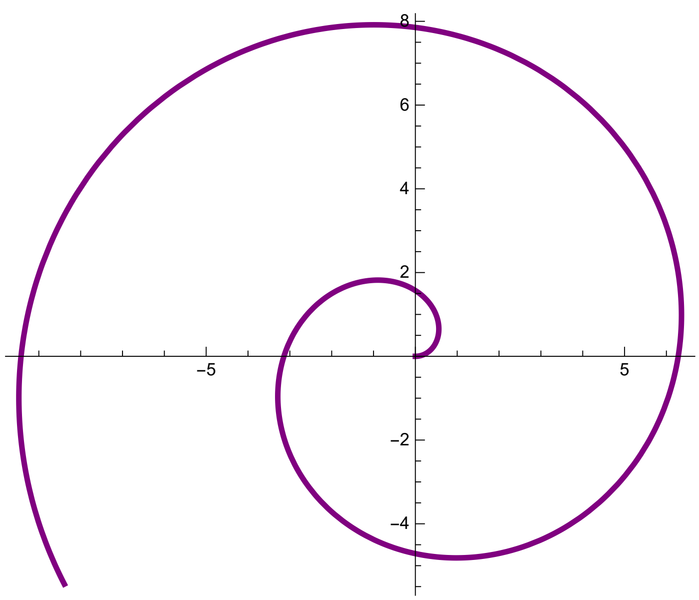
```
ParametricPlot[{Sin[3 t + Pi / 4], Sin[t]},
 {t, 0, 2 Pi}, PlotStyle → {Green, Thickness[0.011]}]
```



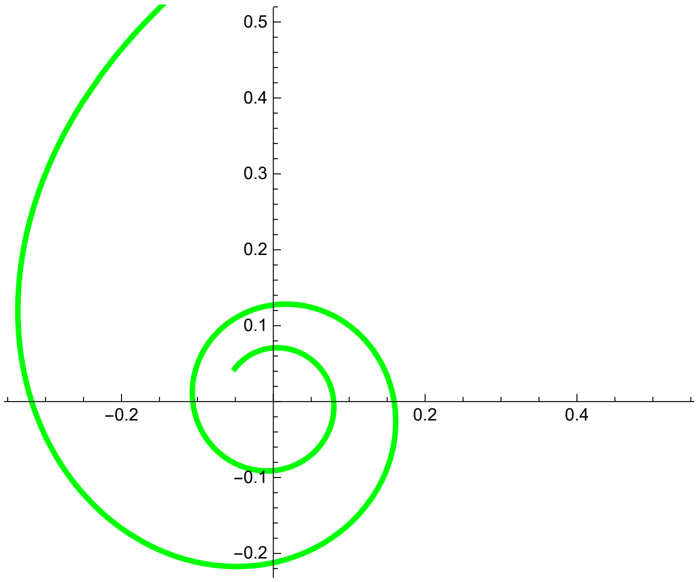
Polar Coordinate Graphs

Coordinate Graphs Polar

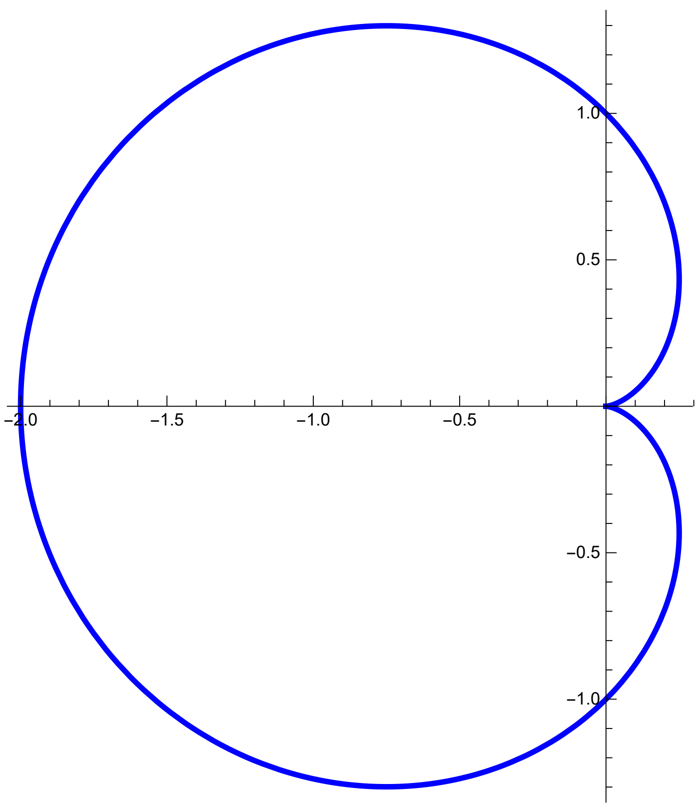
```
PolarPlot[θ, {θ, 0, 10}, PlotStyle → {Purple, Thickness[0.008]}]
```



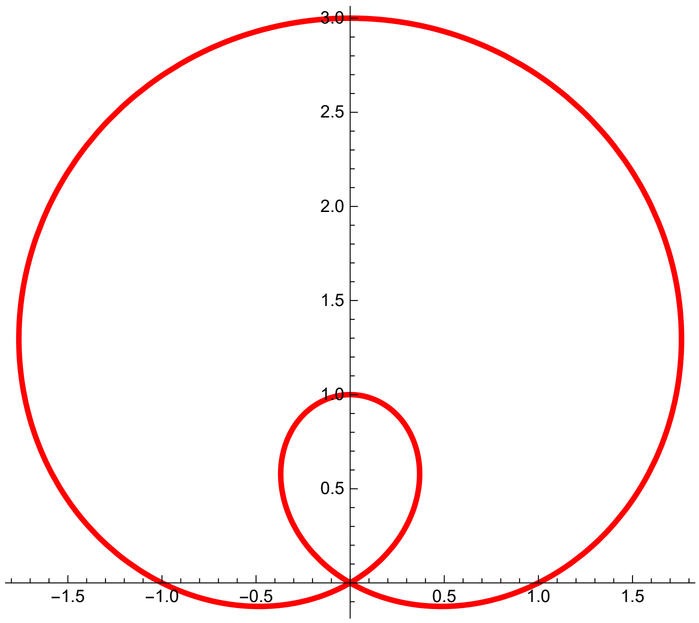
```
PolarPlot[1/θ, {θ, 0, 15}, PlotStyle → {Green, Thickness[0.008]}]
```



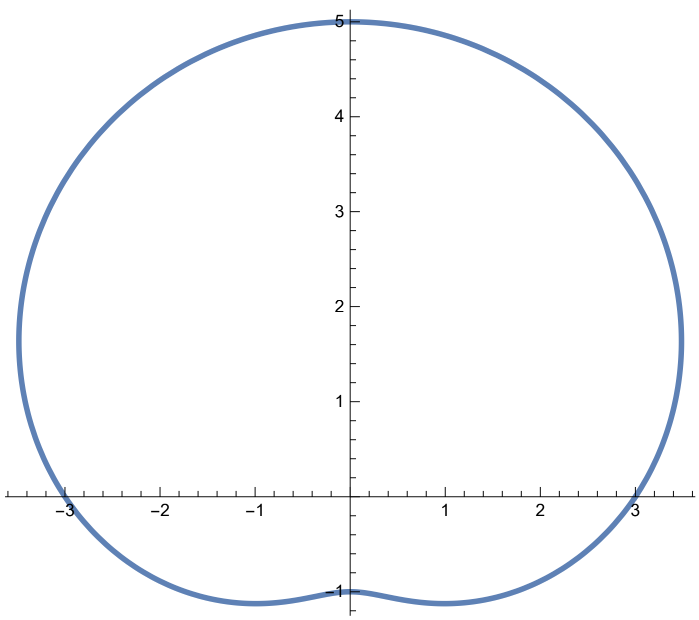

```
PolarPlot[1 - Cos[θ], {θ, 0, 2 π}, PlotStyle → {Blue, Thickness[0.008]}]
```



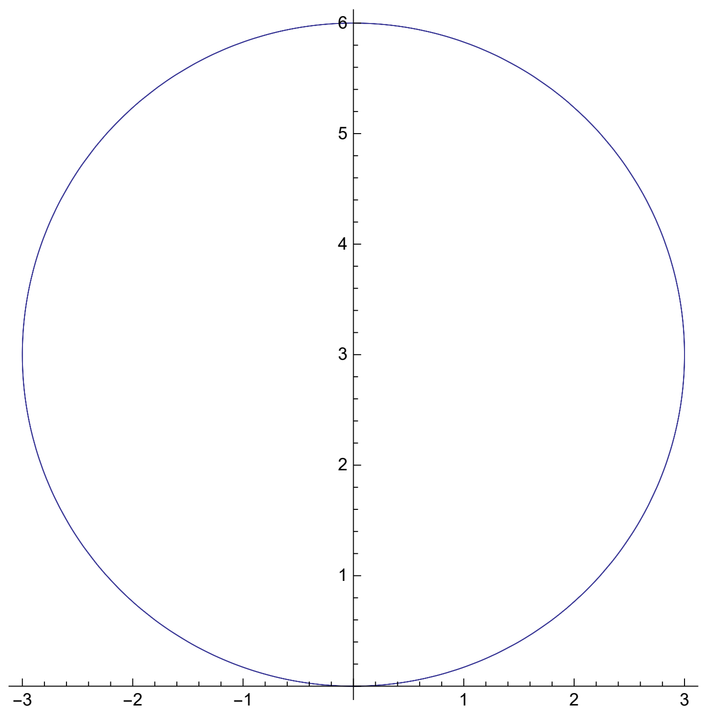
```
PolarPlot[1 + 2 Sin[θ], {θ, 0, 2 π}, PlotStyle → {Red, Thickness[0.008]}]
```



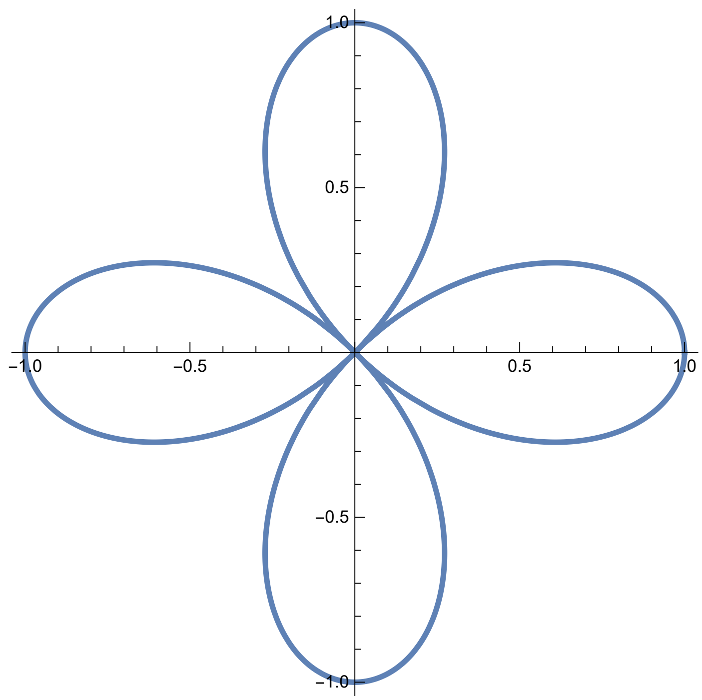

```
PolarPlot[3 + 2 Sin[θ], {θ, 0, 2 π}, PlotStyle → Thickness[0.008]]
```



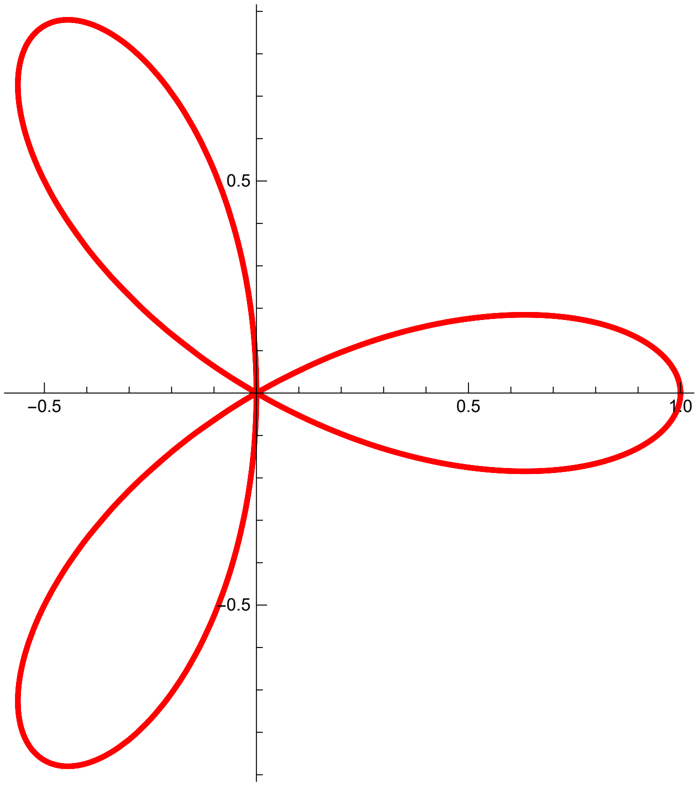
```
PolarPlot[6 Sin[θ], {θ, 0, 2 π}]
```



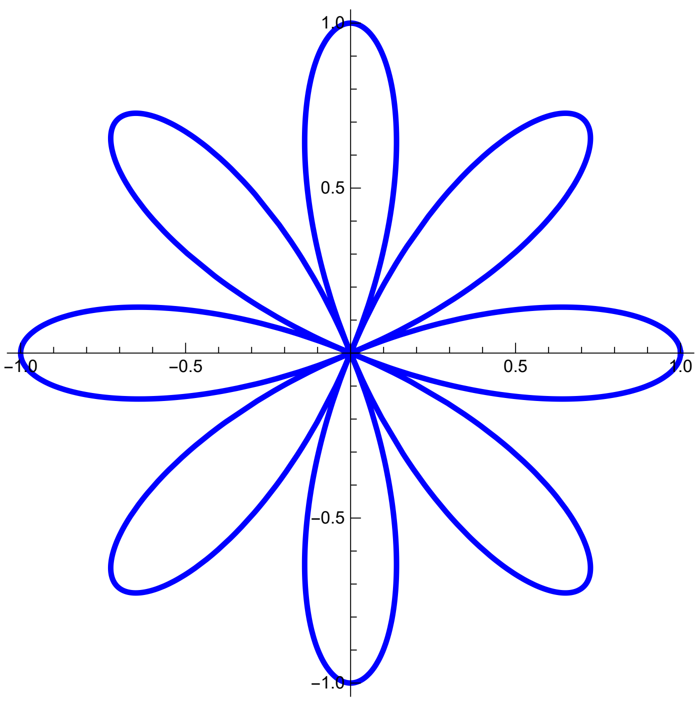

```
PolarPlot[ Cos[2 θ], {θ, 0, 2 π}, PlotStyle → Thickness[0.008]]
```



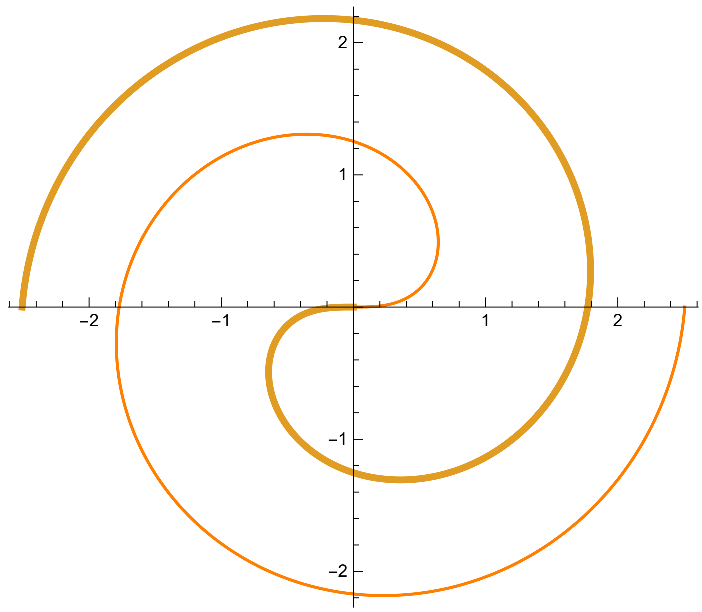
```
PolarPlot[Cos[3 θ], {θ, 0, 2 π}, PlotStyle → {Red, Thickness[0.008]}]
```



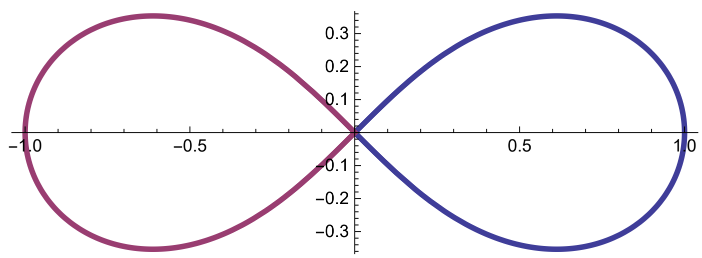

```
PolarPlot[Cos[4 θ], {θ, 0, 2 π}, PlotStyle → {Blue, Thickness[0.008]}]
```



```
PolarPlot[ {Sqrt[θ], -Sqrt[ θ]}, {θ, 0, 2 π}, PlotStyle → {Orange, Thickness[0.01]}]
```




```
PolarPlot[ {Sqrt[Cos[2 θ]], -Sqrt[Cos[2 θ]]},
 {θ, -π/4, π/4}, PlotStyle → { Thickness[0.008]}]
```



```
ParametricPlot3D[{{Sin[t], Cos[t], t/8}, {.7, .7, Pi/32} + t {.7, -.7, 1/8}},
 {t, -4, 5}, PlotStyle → {AbsoluteThickness[5.]}, AspectRatio → 1]
```