

3

REPRESENTACIÓN DE CURVAS EN FORMA IMPLÍCITA-CURVAS DE NIVEL

3.1. Curvas definidas de forma implícita

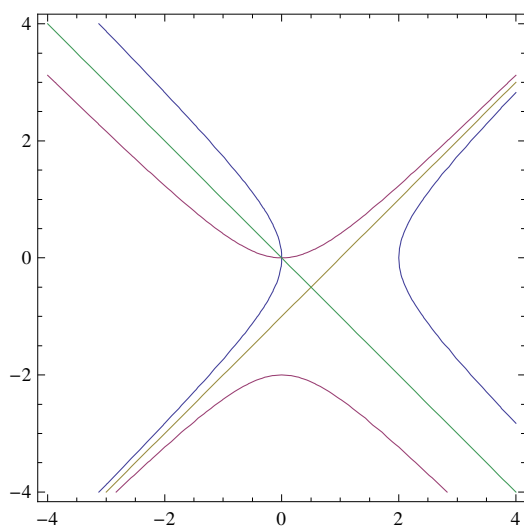
Este comando se utiliza para la representación gráfica de una curva definida de forma implícita por la ecuación $f(x,y)=0$ en un sistema de ejes coordenados rectangulares bidimensional OXY.

▼ Función `ContourPlot`

Se puede dibujar una única función implícita o varias sobre los mismos ejes

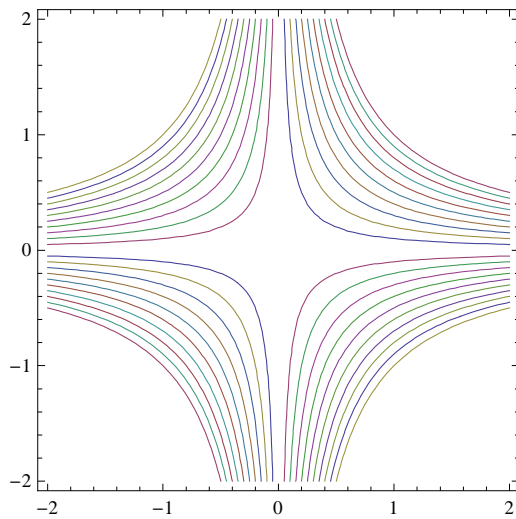
★ `ContourPlot [{ecuación1,ecuación2,...,ecuaciónn} , {x,xmin,xmáx} , {y,ymin,ymáx}]`

```
ContourPlot [ {x^2 - y^2 == 2 x, -x^2 + y^2 == -2 y, y == x - 1, y == -x} , {x, -4, 4} , {y, -4, 4} ]
```



▼ Gráfica de una familia de curvas

```
ContourPlot[Evaluate[Table[{x*y == 0.1*k, x*y == -0.1*k}, {k, 1, 10}]],
  {x, -2, 2}, {y, -2, 2}]
```

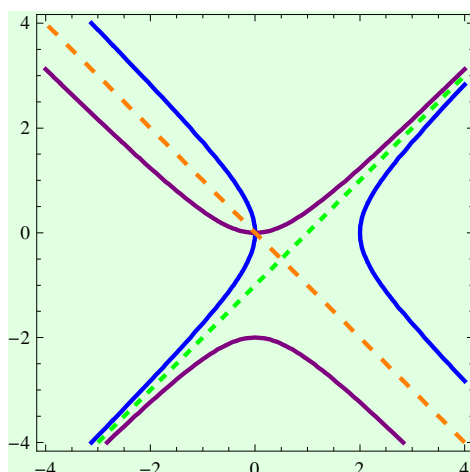


▼ Opciones a utilizar con la función ContourPlot

Algunas de las opciones que se pueden añadir a la función Plot para cambiar su especificación por defecto, también se pueden añadir a la función ContourPlot. La opción PlotStyle hay que adaptarla para la instrucción ContourPlot mediante ContourStyle

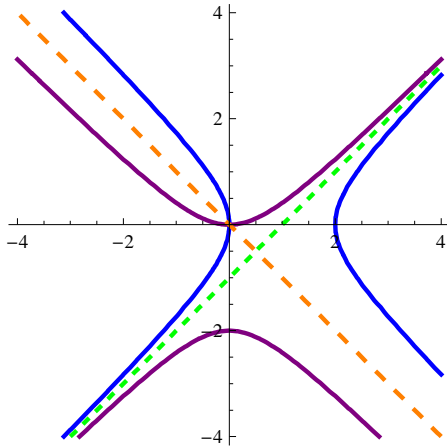
★ Estilo y color de fondo : ContourStyle→ {color, Thickness[n], Dashing[n]}, Background→Color

```
ContourPlot[{x^2 - y^2 == 2 x, -x^2 + y^2 == -2 y, y == x - 1, y == -x},
  {x, -4, 4}, {y, -4, 4}, ContourStyle → {{Blue, Thickness[0.01]},
  {Purple, Thickness[0.01]}, {Green, Thickness[0.01], Dashing[0.02]},
  {Orange, Thickness[0.01], Dashing[0.03]}}, Background → LightGreen]
```



★ Marco y Ejes

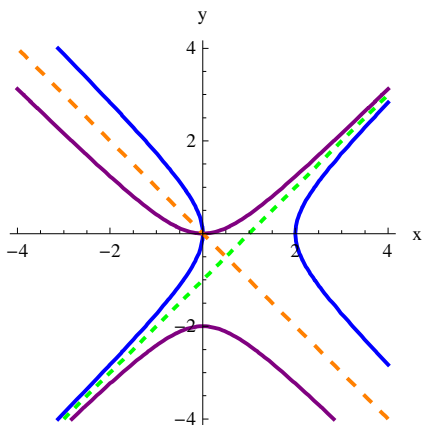
```
ContourPlot[{x^2 - y^2 == 2 x, -x^2 + y^2 == -2 y, y == x - 1, y == -x}, {x, -4, 4},
  {y, -4, 4}, ContourStyle -> {{Blue, Thickness[0.01]}, {Purple, Thickness[0.01]},
  {Green, Thickness[0.01], Dashing[0.02]}, {Orange, Thickness[0.01], Dashing[0.03]}},
  Axes -> True, Frame -> False, AspectRatio -> Automatic]
```



★ Etiquetas

```
ContourPlot[{x^2 - y^2 == 2 x, -x^2 + y^2 == -2 y, y == x - 1, y == -x}, {x, -4, 4},
  {y, -4, 4}, ContourStyle -> {{Blue, Thickness[0.01]}, {Purple, Thickness[0.01]},
  {Green, Thickness[0.01], Dashing[0.02]}, {Orange, Thickness[0.01], Dashing[0.03]}},
  Axes -> True, Frame -> False, AxesLabel -> {"x", "y"}, PlotLabel ->
  Style["FUNCIONES IMPLICITAS", 16, Bold, RGBColor[0.2, 0.5, 0.1]], AspectRatio -> Automatic]
```

FUNCIONES IMPLICITAS

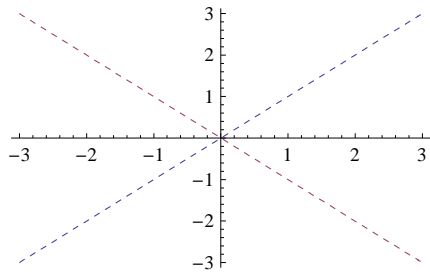


▼ Combinación de gráficos

Se puede mostrar varios gráficos combinados, si esos gráficos están previamente dibujados

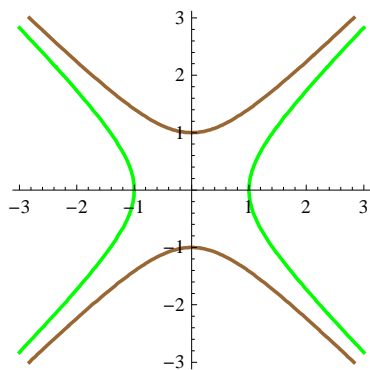
★ Show [gráfico1, gráfico2,...]

```
grafico1 = Plot[{x, -x}, {x, -3, 3}, PlotStyle -> Dashing[{0.015}]
```

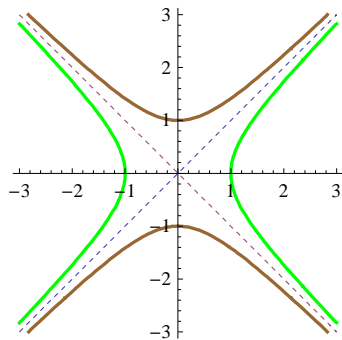


★ Algunas opciones para cambiar el aspecto de los gráficos

```
grafico2 = ContourPlot[{x^2 - y^2 == 1, -x^2 + y^2 == 1}, {x, -3, 3}, {y, -3, 3}, Axes -> True,
  Frame -> False, ContourStyle -> {{Green, Thickness[0.01]}, {Brown, Thickness[0.01]}}
```



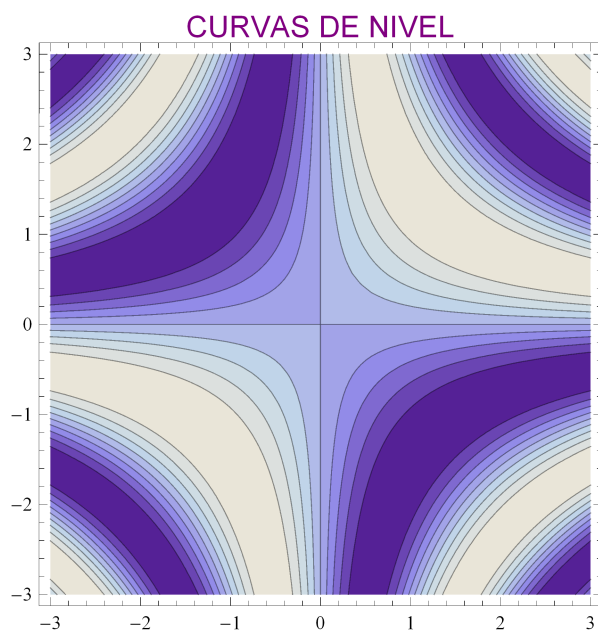
```
Show[grafico1, grafico2, AspectRatio -> Automatic]
```



3.2. Curvas de nivel de una función de dos variables

▼ Curvas de nivel

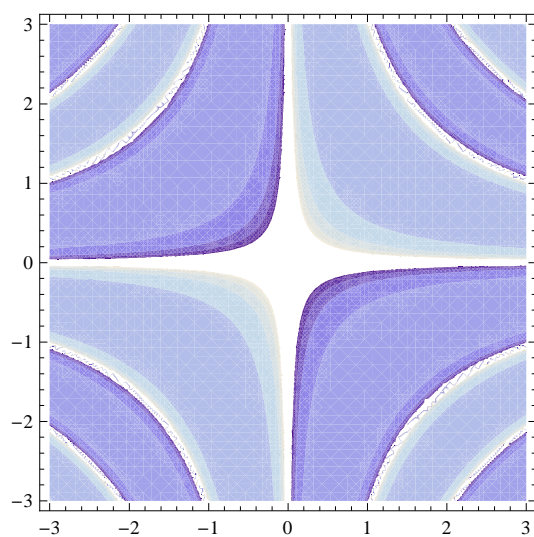
```
ContourPlot[Sin[x * y], {x, -3, 3}, {y, -3, 3}, PlotLabel -> "CURVAS DE NIVEL"]
```



▼ Otras opciones de Contourstyle

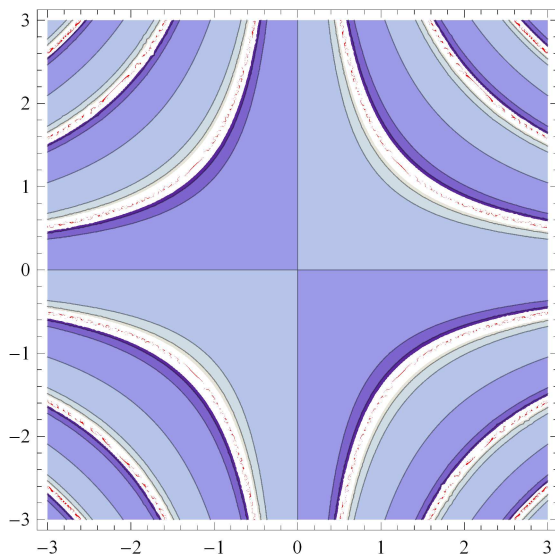
★ ContourStyle -> None

```
ContourPlot[1 / Sin[x * y], {x, -3, 3}, {y, -3, 3}, ContourStyle -> None]
```



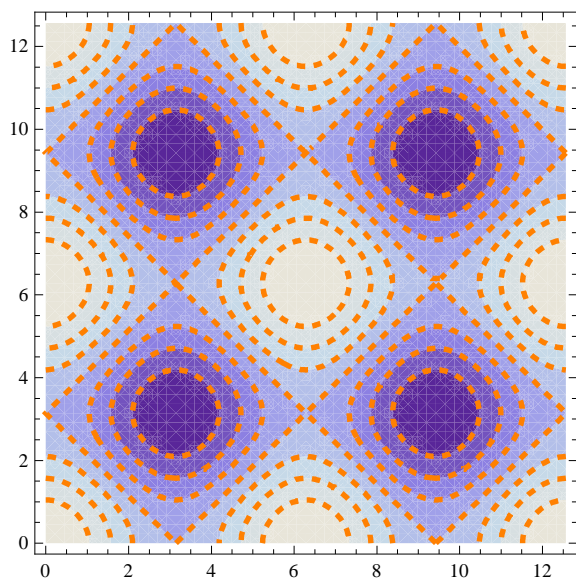
★ Exclusions

```
ContourPlot[Tan[x * y], {x, -3, 3}, {y, -3, 3},  
Exclusions -> Cos[x * y] == 0, ExclusionsStyle -> Red]
```



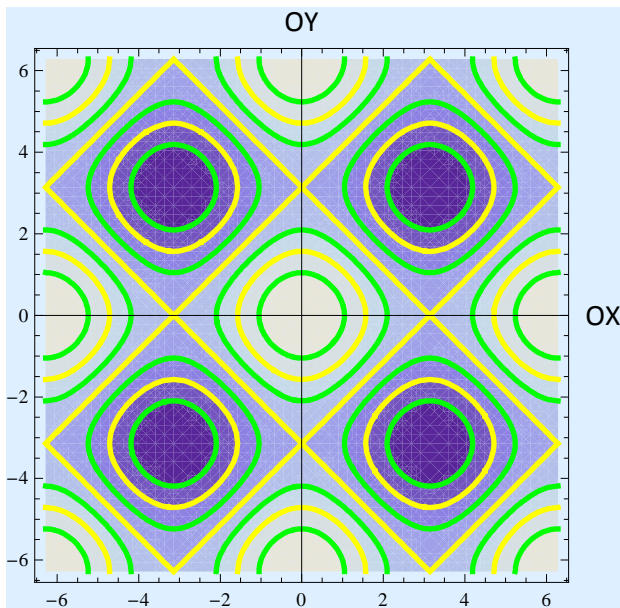
★ ContourStyle -> Directive[Color, Grosor, Trazo]

```
ContourPlot[Cos[x] + Cos[y], {x, 0, 4 Pi}, {y, 0, 4 Pi},  
ContourStyle -> Directive[Orange, Thickness[0.01], Dashed]]
```



★ Alternando colores con las curvas de nivel

```
ContourPlot[Cos[x] + Cos[y], {x, -2 π, 2 π}, {y, -2 π, 2 π},
  ContourStyle -> {{Thickness[0.01], Yellow}, {Thickness[0.01], Green}},
  Axes -> True, AxesLabel -> {"OX", "OY"}, Background -> LightBlue]
```



★ PlotRange

```
g1 = ContourPlot[x^2 - y^2, {x, -4, 4}, {y, -4, 4},
  ContourStyle -> {{Thickness[0.01], Red}, {Thickness[0.01], Blue}}, Axes -> True];

g2 = ContourPlot[x^2 - y^2, {x, -4, 4}, {y, -4, 4},
  ContourStyle -> {{Thickness[0.01], Red}, {Thickness[0.01], Blue}},
  Axes -> True, PlotRange -> {-2, 2}];
```

```
GraphicsGrid[{{g1}, {g2}}]
```

