MATH 151 Lab 6

Carson Kjar, Ashton Isaac Hull, Jax Colby Lanie, Hudson Hurtig

section 525

In [22]:
```
from sympy import *
from sympy import Symbol, N
from sympy.plotting import (plot,plot_parametric)
from sympy.plotting import (plot,plot_parametric)
```

Question 1

1a

In [9]:
```
x = symbols('x')
r = symbols('r')

y = exp(r*x)

yPrime = diff(y,x)

yDoublePrime = diff(y,x,2)

a = 2*yDoublePrime + yPrime - y


print("solutions to the equation 2y''+y'- y = 0 are", solve(a,r))
```

```
solutions to the equation 2y''+y'- y = 0 are [-1, 1/2]  thus there are no solutions t
o the problem
```

1b

In [11]:
```
b = yDoublePrime + 6*yPrime + 10*y

print("solutions to the equation y''+6y'+ 10y = 0 are", solve(b,r))
```

```
solutions to the equation y''+6y'+ 10y = 0 are [-3 - I, -3 + I]
```

1c

In [13]:
```
y = exp(-3*x)*(cos(x)+sin(x))

yPrime = diff(y,x)

yDoublePrime = diff(y,x,2)

b = yDoublePrime + 6*yPrime + 10*y

print("solutions to the equation y''+6y'+ 10y = 0 are", solve(b,x), " thus there are r
```

```
solutions to the equation y''+6y'+ 10y = 0 are []
```

Question 2

2a

In [88]:
```
t = symbols('t')

vx = exp(2*sin(t))
vy = exp(cos(t))

vxPrime = diff(vx,t)
vyPrime = diff(vy,t)

slope = vyPrime/vxPrime


print("the equation in exact form tangent to the point of the equation at t = pi/6 is

print("the equation in decimal form tangent to the point of the equation at t = pi/6 i
```

```
the equation in exact form tangent to the point of the equation at t = pi/6 is y = -s
qrt(3)*(x - E)*exp(-1)*exp(sqrt(3)/2)/6 - exp(sqrt(3)/2)
the equation in decimal form tangent to the point of the equation at t = pi/6 is y =
-0.252478818450665*x - 1.69113409097091
```

2b

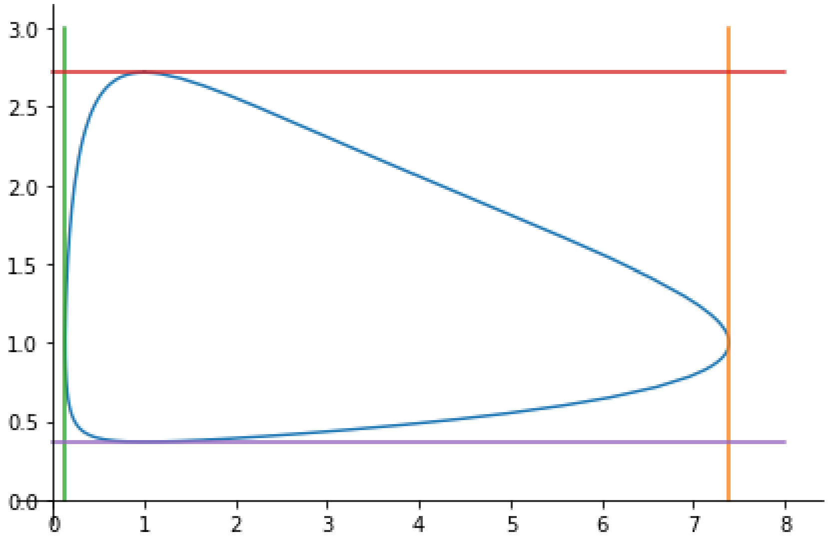
In [89]:
```
print("the slope line is vertical when the slope approaches anyrealnumber/0")
print("thus points where the slope line is vertical are where t =", solve(vxPrime,t))
print("")
print("the slope line is horizontal when the slope is 0/anyrealnumber")
print("thus the points where the slope line is horizontal are where t =", solve(vyPrim
```

```
the slope line is vertical when the slope approaches anyrealnumber/0
thus points where the slope line is vertical are where t = [pi/2, 3*pi/2]

the slope line is horizontal when the slope is 0/anyrealnumber
thus the points where the slope line is horizontal are where t = [0, pi]
```

2c

In [101…
```
plot_parametric((vx, vy, (t,0,2*pi)), (exp(2),t,(t,0,3)), (exp(-2),t,(t,0,3)),(t,exp(c
```



Out[101]: <sympy.plotting.plot.Plot at 0x2b06a46f100>

Question 3

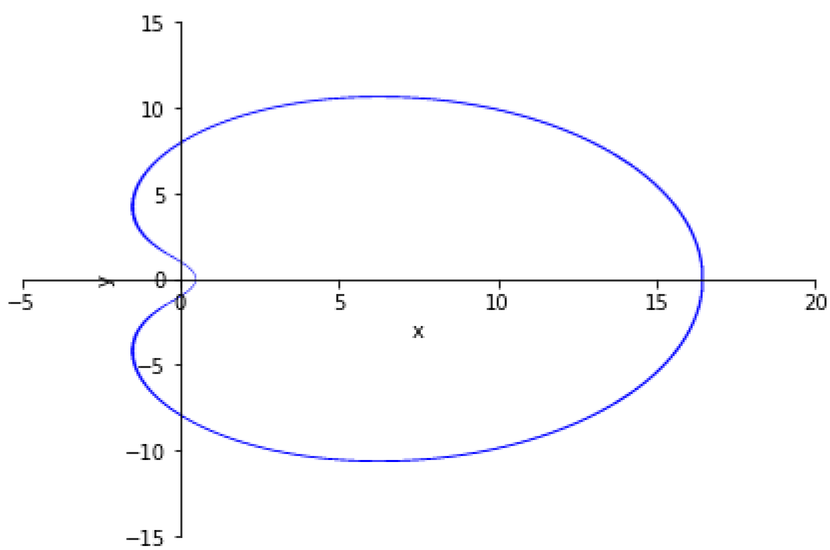
3a

In [4]:
```
y = symbols('y', real=True)

x = symbols('x', real=True)

eqn = (-((x**2 + y**2)/4) + (2*x)-2)**2  - 5*(x**2 + y**2)

pcurve=plot_implicit(eqn,(x,-5,20),(y,-15,15))
```



3b

In [5]:
```
eqnPrime = idiff(eqn,y,x)

eqnPrime = simplify(eqnPrime)

print("dy/dx of the equation graphed above is:", eqnPrime)
```

```
dy/dx of the equation graphed above is: (-x**3 + 12*x**2 - x*y**2 + 4*y**2 + 32)/(y*
(x**2 - 8*x + y**2 - 32))
```

3c

In [11]:

```
num = numer(eqnPrime)
den = denom(eqnPrime)



vert = solve([den.evalf(),eqn],[x,y])

hori = solve([num.evalf(),eqn],[y,x])


print("the slope line is vertical when the slope approaches anyrealnumber/0")
print("thus points where the slope line is vertical are where x =", [i for i in vert])
print("")
print("the slope line is horizontal when the slope is 0/anyrealnumber")
print("thus the points where the slope line is horizontal are where y =", [i for i in 
```

```
the slope line is vertical when the slope approaches anyrealnumber/0
thus points where the slope line is vertical are where x = [(-1.50000000000000, -4.21
307488658818), (-1.50000000000000, 4.21307488658818), (0.486080130000819, 0.0), (16.4
581917799983, 0.0)]

the slope line is horizontal when the slope is 0/anyrealnumber
thus the points where the slope line is horizontal are where y = [(-10.6473087586551,
6.26905898780519), (10.6473087586551, 6.26905898780519)]
```

3d

In [18]:

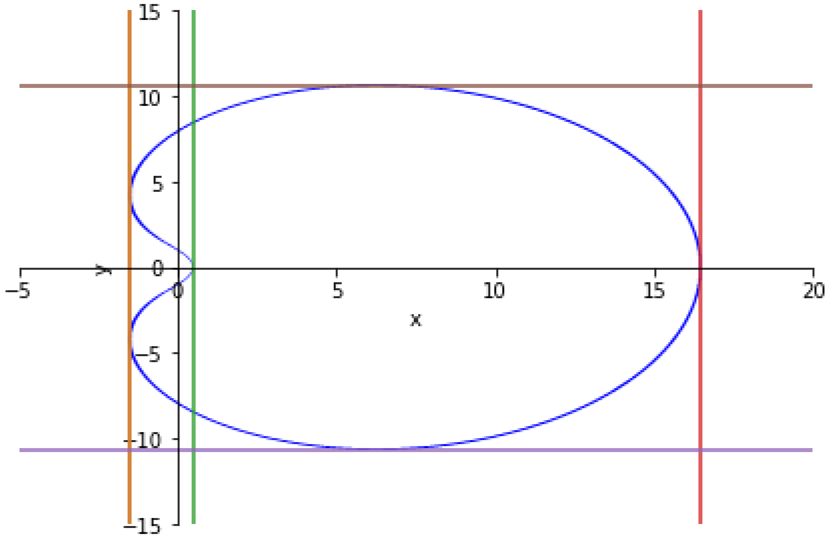
```
pcurve=plot_implicit(eqn,(x,-5,20),(y,-15,15),show=False)
# Using parametric equations to plot horizontal and vertical lines: x=#, y=t␣,→NEED TO
t=symbols('t')
phoriz=plot_parametric(

    (vert[0][0],t,(t,-15,15)),
    (vert[1][0],t,(t,-15,15)),
    (vert[2][0],t,(t,-15,15)),
    (vert[3][0],t,(t,-15,15)),

    show=False)
# Both of these CAN be combined into one plot_parametric command if you want
pvert=plot_parametric(

    (t,hori[0][0],(t,-5,20)),
    (t,hori[1][0],(t,-5,20)),

    show=False)
pcurve.extend(phoriz)
pcurve.extend(pvert)
pcurve.show()
```



Question 4

4a

In [48]:
```
y = symbols('y', real=True)
x = symbols('x', real=True)

y = (x**1.5*sqrt(x**3+1))/(2-7*x)**4

expy = expand_log(log(y),force=True)

logder = y * diff(expy)
der = diff(y)

print("logarithmic differentiation  finds the derrivative to be:", logder)
```

```
logarithmic differentiation  finds the derrivative to be: x**1.5*sqrt(x**3 + 1)*(3*x*
*2/(2*(x**3 + 1)) + 28/(2 - 7*x) + 1.5/x)/(2 - 7*x)**4
-0.00536*sqrt(2)

 -0.00536*sqrt(2)
```

4b

In [47]:
```
print("direct differentiation finds the derrivative to be:", der)
```

```
direct differentiation finds the derrivative to be 1.5*x**0.5*sqrt(x**3 + 1)/(2 - 7*
x)**4 + 28*x**1.5*sqrt(x**3 + 1)/(2 - 7*x)**5 + 3*x**3.5/(2*(2 - 7*x)**4*sqrt(x**3 +
1))
```

4c

In [55]:
```
print("although they appear different, when we substitute for x = 1 we find that they

print("\n\nalternatively we see that when we subtract the equations from eachother the
```

although they appear different, when we substitute for x = 1 we find that they output the same thing:

-0.00536*sqrt(2)

-0.00536*sqrt(2)

alternatively we see that when we subtract the equations from eachother they equal zero 0

```
In [ ]:
```