MATH 151 Lab 6

Jaeheon Kim, Vishwa Kumarvel, Alex Dao, Nico Bui

In [59]:
```
from sympy import *
from sympy.plotting import (plot,plot_parametric)
```

Question 1

1a

In [60]:
```
import sympy as sp
import sympy as sp
import matplotlib.pyplot as plt
import numpy as np
import math
t = sp.symbols('t')
vector_function = sp.Matrix([sp.exp(2 * sp.sin(t)), sp.exp(sp.cos(t))])
# Part (a): Find the equation for the tangent line at t = /6
t_value = np.pi/6 # Value of t at which we want the tangent line
tangent_vector = vector_function.subs(t, t_value)
tangent_slope = sp.diff(vector_function[1], t) / sp.diff(vector_function[0], t)
tangent_line = tangent_slope * (t - t_value) + tangent_vector[1]
print(f"Part (a): Tangent line equation at t = /6: y = {tangent_line}")
```

```
Part (a): Tangent line equation at t = /6: y = -(t - 0.523598775598299)*exp(-2*sin(t))
*exp(cos(t))*sin(t)/(2*cos(t)) + 2.37744267523617
```

1b

In [61]:
```
# Part (b): Find the points where the tangent line is horizontal and vertical
horizontal_t_values = [[vector_function[1].subs(t,i),vector_function[0].subs(t,i)] for
vertical_t_values = [[vector_function[1].subs(t,i),vector_function[0].subs(t,i)] for i
print("Part (b):")
print(f"(i) Points where the tangent line is horizontal: {horizontal_t_values}")
print(f"(ii) Points where the tangent line is vertical: {vertical_t_values}")
```

```
Part (b):
(i) Points where the tangent line is horizontal: [[E, 1], [exp(-1), 1]]
(ii) Points where the tangent line is vertical: [[1, exp(2)], [1, exp(-2)]]
```

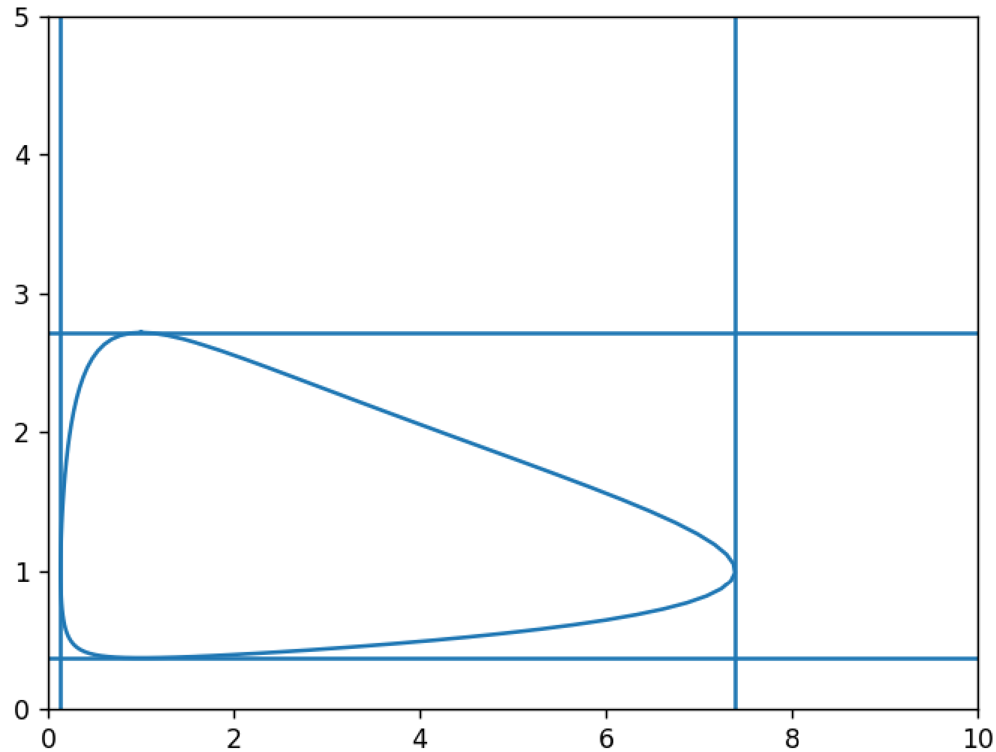
1c

In [62]:
```
matplotlib notebook
```

In [63]:
```
# Part (c): Plot the graph of the vector function and tangent lines
from sympy import *
t_values = np.linspace(0, 2 * np.pi, 100)
x_values = [vector_function[0].subs(t, val) for val in t_values]
y_values = [vector_function[1].subs(t, val) for val in t_values]
plt.plot(x_values, y_values)
plt.axvline(x=E**-2)
plt.axvline(x=E**2)
```

```
plt.axhline(y=E)
plt.axhline(y=E**-1)

plt.xlim(0, 10) # Adjust the x-axis limits as need
plt.ylim(0, 5) # Adjust the y-axis limits as needed
plt.show()
```



Question 2

2a

In [64]:
```
from sympy import *
x,y=symbols('x y')
eqn = x**2 + 2*x + 3*y**2
dydx=idiff(eqn,y,x)
print('dy/dx=', dydx)
```

```
dy/dx= -(x + 1)/(3*y)
```

2b

In [65]:
```
from sympy import *
x, y = symbols('x y')
eqn = Eq(x**2 + 2*x + 3*y**2, 8)
horizontal_eq = Eq(numer(dydx), 0)
h_tan = solve(horizontal_eq, x)
x_coor = h_tan[0]
print(f"the horizontal tangent is at x = {x_coor}")
```

```
print("Now, we have to plug it back into the original equation")
y_points = solve(eqn.subs(x, x_coor), y)
print(y_points)
print(f"Now, the points where the graph of the equation has a horizontal tangent line
for y_point in y_points:
    print(f"({x_coor}, {y_point})")
```

```
the horizontal tangent is at x = -1
Now, we have to plug it back into the original equation
[-sqrt(3), sqrt(3)]
Now, the points where the graph of the equation has a horizontal tangent line are:
(-1, -sqrt(3))
(-1, sqrt(3))
```

2c

In [66]:

```
from sympy import *
x, y = symbols('x y')
eqn = Eq(x**2 + 2*x + 3*y**2, 8)
vert_eq = Eq(denom(dydx), 0)
v_tan = solve(vert_eq, y)
y_coor = v_tan[0]
print(f"the vertical tangent is at y = {y_coor}")
print("Now, we have to plug it back into the original equation")
x_points = solve(eqn.subs(y, y_coor), x)
print(x_points)
print(f"Now, the points where the graph of the equation has a vertical tangent line ar
for x_point in x_points:
    print(f"({x_point}, {y_coor})")
```

```
the vertical tangent is at y = 0
Now, we have to plug it back into the original equation
[-4, 2]
Now, the points where the graph of the equation has a vertical tangent line are:
(-4, 0)
(2, 0)
```

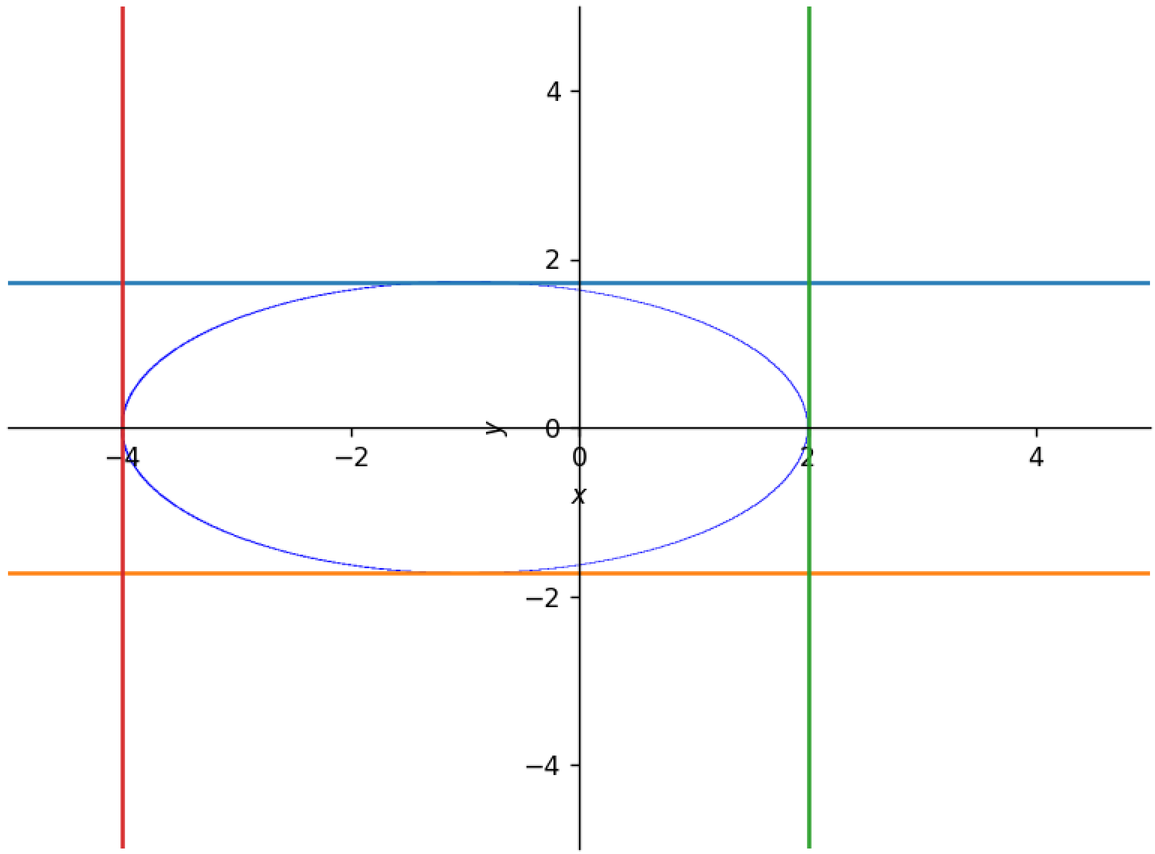
2d

In [67]:

```
matplotlib notebook
```

In [68]:

```
x, y = symbols('x y')
eqn = Eq(x**2 + 2*x + 3*y**2, 8)
t = symbols('t')
horizontal_tangent_points = [(-1, sqrt(3)), (-1, -sqrt(3))]
vertical_tangent_points = [(2, 0), (-4, 0)]
pcurve = plot_implicit(eqn, (x, -5, 5), (y, -5, 5), show=False)
phoriz = plot_parametric((t, horizontal_tangent_points[0][1], (t, -5, 5)),
                         (t, horizontal_tangent_points[1][1], (t, -5, 5)), show=False)
pvert = plot_parametric((vertical_tangent_points[0][0], t, (t, -5, 5)),
                        (vertical_tangent_points[1][0], t, (t, -5, 5)), show=False)
pcurve.extend(phoriz)
pcurve.extend(pvert)
pcurve.show()
```



Question 3

3a

In [69]:
```
from sympy import *
from math import*
k,y0,t,y=symbols('k y0 t y',real = True)
eq1  = y0*E**(k*2)-1000
eq2 = y0*E**(k*6)-40000
y0_k = solve((eq1,eq2),(k,y0))
K = y0_k[0][0]
YO = y0_k[0][1]
print(f'k is {K.evalf()} and y0 is {YO.evalf()}')
```

```
k is 0.922219863528484 and y0 is 158.113883008419
```

3b

In [70]:
```
eq3  = YO*E**(K*t)-2000000
T = solve(eq3,t)
print(f'It reaches 2,000,000 after {T[0].evalf()} hours')
```

```
It reaches 2,000,000 after 10.2419635058178 hours
```

3c

In [71]:
```
Y0 = 1000
eq4 = Y0*E**(k*4)-40000
```

```
K = solve(eq4,k)
eq5 = Y0*E**(K[0]*-2)-y
Y = solve(eq5,y)
print(f'k is {K[0].evalf()} and the amount of bacteria 2 hours before the initial time
print()
print(f'The amount of bacteria 2 hours before the initial time is the same as the y0,{
```

```
k is 0.922219863528484 and the amount of bacteria 2 hours before the initial time is15
8.113883008419

The amount of bacteria 2 hours before the initial time is the same as the y0,158.11388
3008419, in part a
```

In [ ]: