MATH 151 Lab 7

Put team members' names and section number here.

In [49]:
```
from sympy import *
from sympy.plotting import (plot,plot_parametric)
```

Question 1

1a

In [97]:
```
from sympy import symbols, diff, solve, limit
x = symbols('x')
f = (x**2 + 8*x + 5)/(x**2) # this part will solve for derivative
f_der = diff(f, x)
# Find the critical points by finding where f'(x) = 0 or undefined
critical_points = solve(f_der, x)
# Check the endpoints of the interval
endpoints = [-3, -1]
critical_points += endpoints
# Calculate the function values at the critical points and endpoints
extrema_values = [f.subs(x, point) for point in critical_points]
# Find the maximum and minimum values
max_value = max(extrema_values)
min_value = min(extrema_values)
print(f"(a) Absolute maximum value: {max_value} at x ={critical_points[extrema_values.
print(f" Absolute minimum value: {min_value} at x ={critical_points[extrema_values.inc
```

```
(a) Absolute maximum value: -10/9 at x =-3
 Absolute minimum value: -11/5 at x =-5/4
```

1b

In [51]:
```
endpoints = [-3, 1] # New interval
critical_points += endpoints
extrema_values = [f.subs(x, point) for point in critical_points]
max_value = max(extrema_values)
min_value = min(extrema_values)
print(f"(b) Absolute maximum value: {max_value} at x ={critical_points[extrema_values.
print(f" Absolute minimum value: {min_value} at x ={critical_points[extrema_values.inc
```

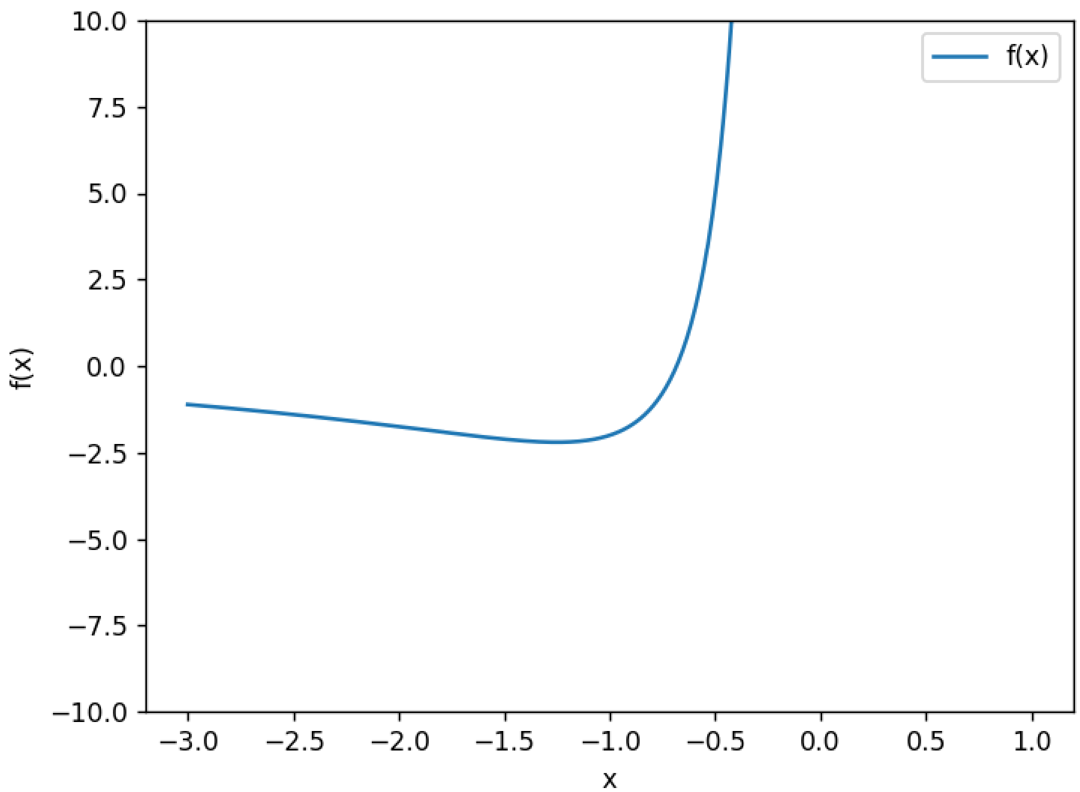
```
(b) Absolute maximum value: 14 at x =1
 Absolute minimum value: -11/5 at x =-5/4
```

1c

In [ ]:

In [110…
```
import matplotlib.pyplot as plt
import numpy as np
# Define the function using numpy
f_np = lambda x: (x**2 + 8*x + 5) / (x**2)
# Create an array of x values within the interval
```

```
x_values = np.linspace(-3, 1, 400)
y_values = [f_np(x) for x in x_values]
# Plot the function
plt.plot(x_values, y_values, label='f(x)')
# Add labels and legend
plt.xlabel('x')
plt.ylabel('f(x)')
plt.legend()
# Set ylim to see the graph accurately
plt.ylim(-10, 10)
# Show the plot
```



Out[110]:

```
(-10.0, 10.0)
```

Question 2

2a

In [90]:

```
from sympy import *
x = symbols('x')
a = 2
b = 7
fx = 4 * x**3 - 3 * x**2 + 2 * x - 1
f_p_x = fx.diff(x)
fa = fx.subs(x, a)
fb = fx.subs(x, b)
mvt_eq = Eq(f_p_x, (fb - fa) / (b - a))
c = solve(mvt_eq, x)
```

```
c=c[1]
print(c)
```

```
1/4 + sqrt(2901)/12
```

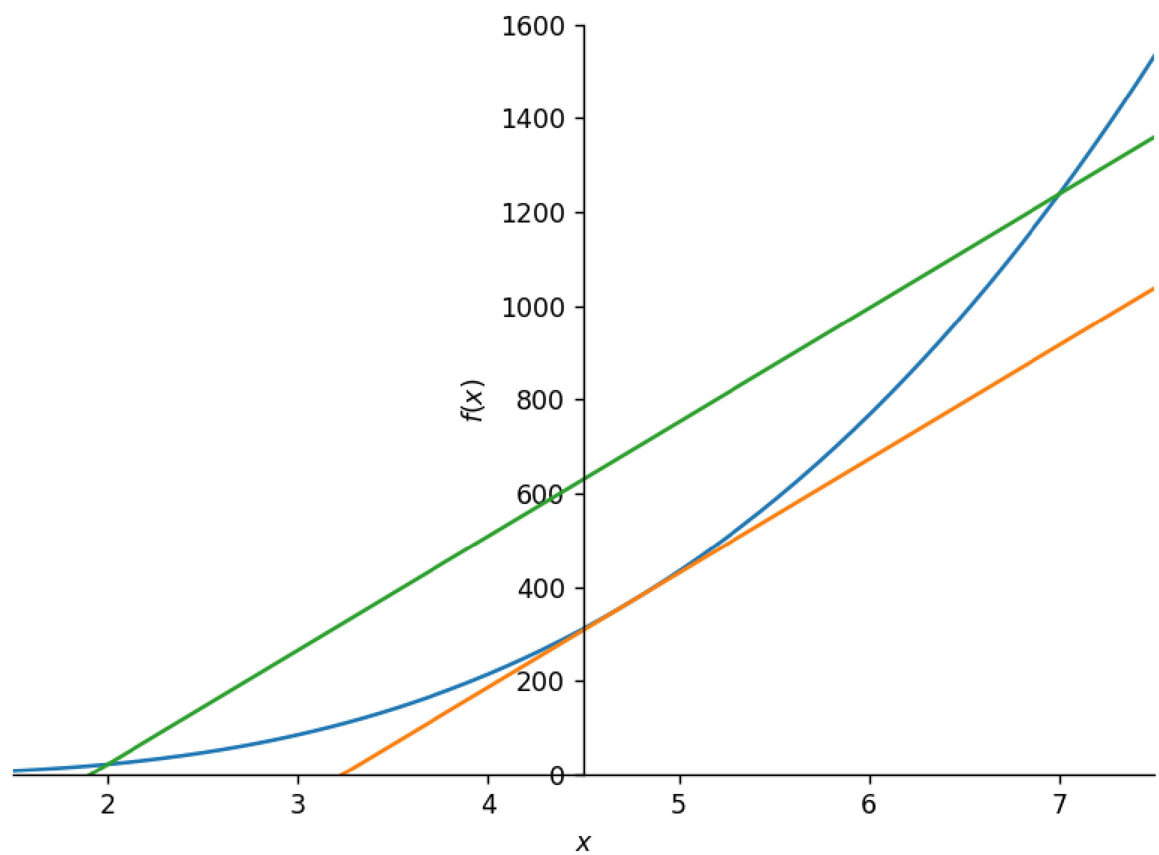
2b

In [55]:

```

```

In [111…

```
from sympy import *
from sympy.plotting import (plot,plot_parametric)
# Secant Line
secl = ((fb - fa) / (b - a)) * (x - a) + fa
# Tangent Line
tanl = f_p_x.subs(x, c) * (x - c) + fx.subs(x,c)
# Plot
p1 = plot(fx, tanl, secl,(x,1.5,7.5),show=False)
p1.ylim = [0, 1600]
p1.xlim= [1.5, 7.5]
p1.show()
```



Question 3

3a

In [114…

```
from sympy import*
from math import *
x = symbols('x')
```

```
eq1 = 8-x**2
eq2 = 5*E**((x-2)/2)-x
deq1 = diff(eq1)
deq2 = diff(eq2)
deq1,deq2 = solveset(deq1,x,Interval(-oo,0)),solveset(deq2,x,Interval(0,oo))
print(f'The critical value is at x={deq2}')
```

```
The critical value is at x={2*log(2/5) + 2}
```

3b

In [115…

```
lb =eq1.subs(x,-5)
cp = eq2.subs(x,2*log(2/5)+2)
cp1 = eq2.subs(x,0)
ub = eq2.subs(x,5)
ae = max([lb,cp,cp1,ub])
print('The absolute extrema is:')
ae
```

```
The absolute extrema is:
```

Out[115]:

$$-5 + 5e^{\frac{3}{2}}$$

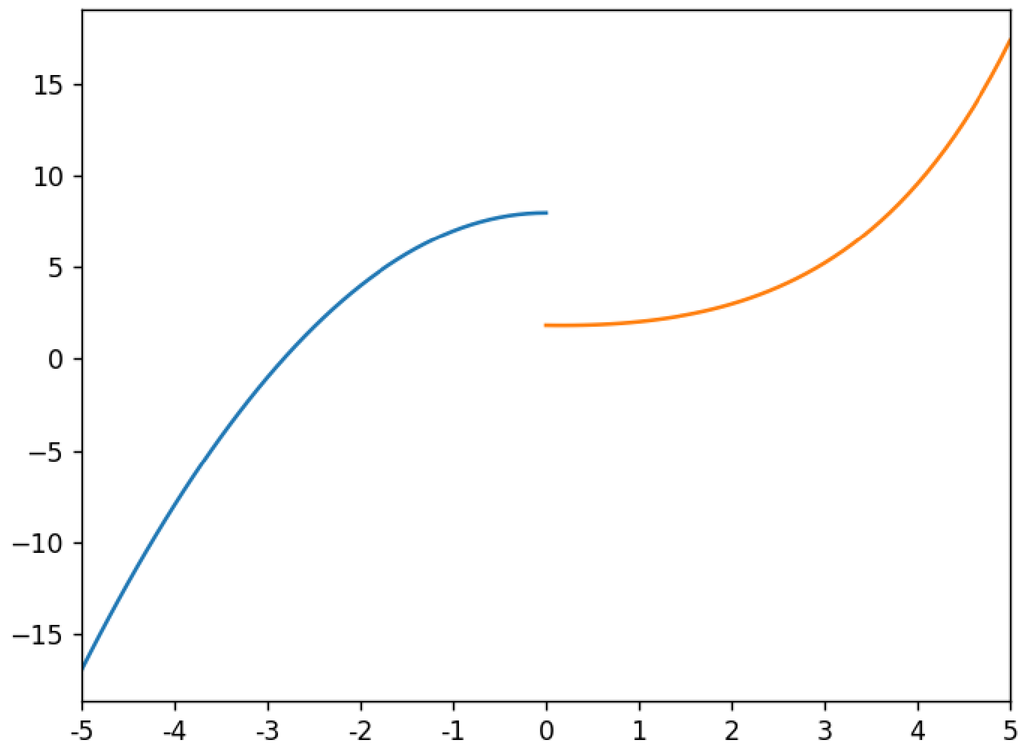
3c

In [59]:

```

```

In [112…

```
import matplotlib.pyplot as plt
import numpy as np
peqx1=[]
peq1=[]
peqx2=[]
peq2=[]

for i in range(51):
    x1 = -i/10
    eq1=8-(x1)**2
    x2 = i/10
    eq2 = 5*E**((x2-2)/2)-x2
    peqx1.append(x1)
    peq1.append(eq1)
    peqx2.append(x2)
    peq2.append(eq2)
plt.xlim(-5,5)
xTicks,xValues = [],[]
for i in range(-5,6):
    xTicks.append(i)
    xValues.append(i)

plt.xticks(xValues,xTicks)
plt.plot(peqx1,peq1)
plt.plot(peqx2,peq2)
plt.show()
```



In [ ]:

In [ ]: