MATH 151 Lab 3

Put team members' names and section number here.

Kevin Lei Jonathan Madding Pan Zhou John Schumacher

Section number 576

In [1]:
```
from sympy import *
from sympy.plotting import (plot,plot_parametric)
```

Question 1

In [6]:
```
#conditions of IVT
print("For f(x) to have a root in the domain of [0, 1], f(0) must be less than zero ar
```

```
For f(x) to have a root in the domain of [0, 1], f(0) must be less than zero and f(1)
must be greater than zero.
```

1a

In [5]:
```
#result of IVT
a, b, c, d = symbols('a b c d')
expr0 = a ** 5 + 4 * a ** 3 - 2 * a ** 2 + 8 * a - 1
print("Plugging in 0 for x produces "+ str(expr0.subs(a, 0)) + ", and plugging in 1 fc
```

```
Plugging in 0 for x produces -1, and plugging in 1 for x produces 10, meaning that th
e y-value 0 must be contained within the domain of [0, 1].
```

1b

In [24]:
```
#root
i = 0
while i < len(solve(expr0, a)):
    if solve(expr0, a)[i] >= 0 and solve(expr0, a)[i] <= 1:
        root = solve(expr0, a)[i]
        break
    i += 1
print("In the domain of [0, 1], f(x) has a root of " + str(float(root)) + ".")
```

```
In the domain of [0, 1], f(x) has a root of 0.12804489141174547.
```

Question 2

2a

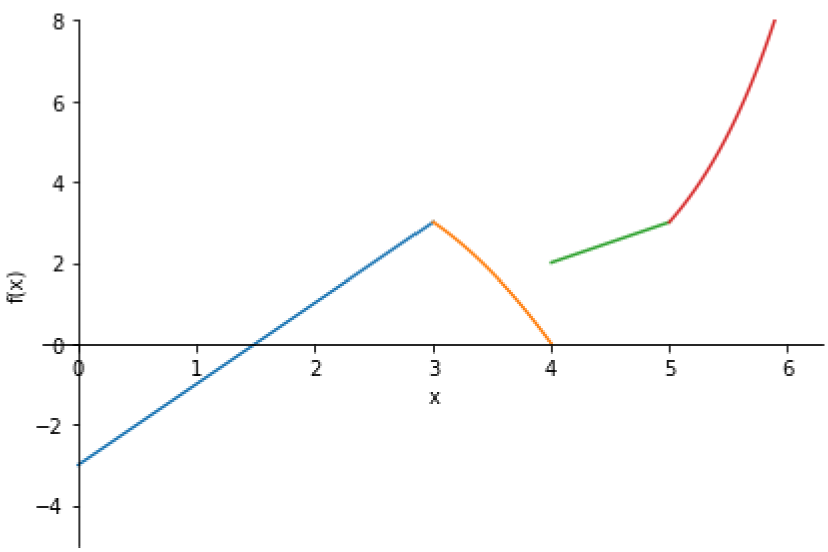
In [25]:
```
#limits
expr1a = 2 * a - 3
expr1b = 4 * a - a ** 2
expr1c = (a ** 2 - 6 * a + 8) / (a - 4)
expr1d = E ** ((a - 4) * ln(3))
```

```
print(f"As x approaches 3 from the left, f(x) approaches {limit(expr1a, a, 3)}.")
print(f"As x approaches 3 from the right, f(x) approaches {limit(expr1b, a, 3)}.")
print("f(x) is continuous at x = 3")
print(f"As x approaches 4 from the left, f(x) approaches {limit(expr1b, a, 4)}.")
print(f"As x approaches 4 from the right, f(x) approaches {limit(expr1c, a, 4)}.")
print("f(x) is discontinuous at x = 4. f(x) is left continuous at x = 4.")
print(f"As x approaches 5 from the left, f(x) approaches {limit(expr1c, a, 5)}.")
print(f"As x approaches 5 from the right, f(x) approaches {limit(expr1d, a, 5)}.")
print("f(x) is continuous at x = 5.")
```

```
As x approaches 3 from the left, f(x) approaches 3.
As x approaches 3 from the right, f(x) approaches 3.
f(x) is continuous at x = 3
As x approaches 4 from the left, f(x) approaches 0.
As x approaches 4 from the right, f(x) approaches 2.
f(x) is discontinuous at x = 4. f(x) is left continuous at x = 4.
As x approaches 5 from the left, f(x) approaches 3.
As x approaches 5 from the right, f(x) approaches 3.
f(x) is continuous at x = 5.
```

2b

In [26]:
```
#graph
plot((expr1a, (a, 0 , 3)), (expr1b, (a, 3, 4)), (expr1c, (a, 4, 5)), (expr1d, (a, 5, 6
```

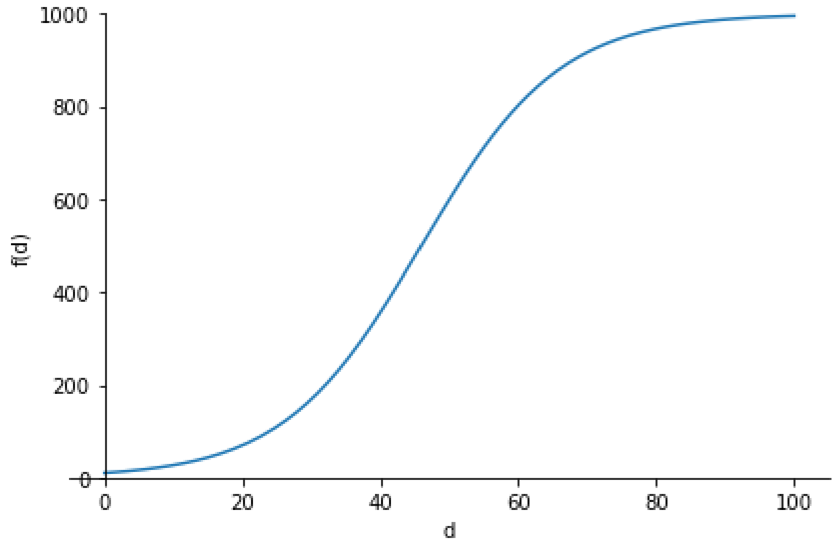


Out[26]: `<sympy.plotting.plot.Plot at 0x2502a995a30>`

Question 3

3a

In [27]:
```
#graph
expr2 = (a * b) / (b + (a - b) * (E ** (-1 * c * d)))
plot(expr2.subs([(a, 1000), (b, 10), (c, 0.1)]), (d, 0, 100), ylim = (0, 1000))
```



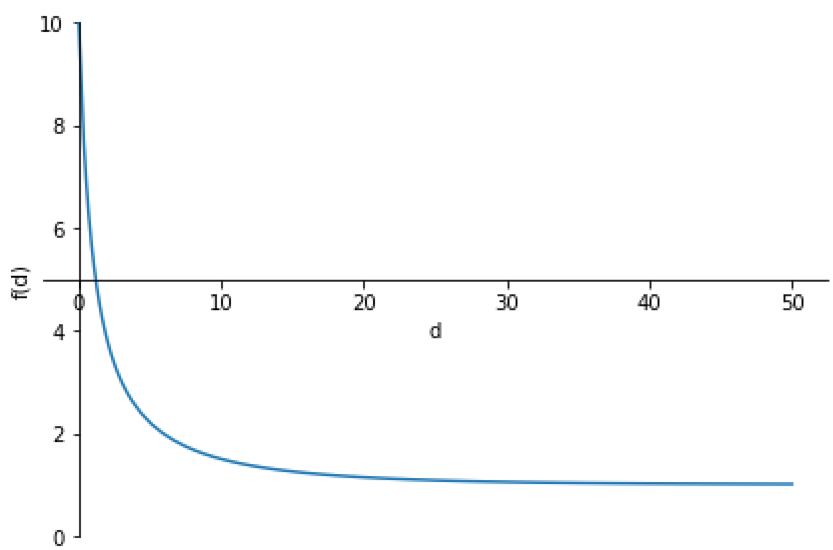
Out[27]: <sympy.plotting.plot.Plot at 0x2502aa03a60>

In [28]:
```
#limit
print(f"The limit of P(t) where K = 1000 as t approaches infinity is {limit(expr2.subs
```

```
The limit of P(t) where K = 1000 as t approaches infinity is 1000.
```

3b

In [29]:
```
#graph
plot(expr2.subs([(a, 1), (b, 10), (c, 0.1)]), (d, 0, 50), ylim = (0, 10))
```



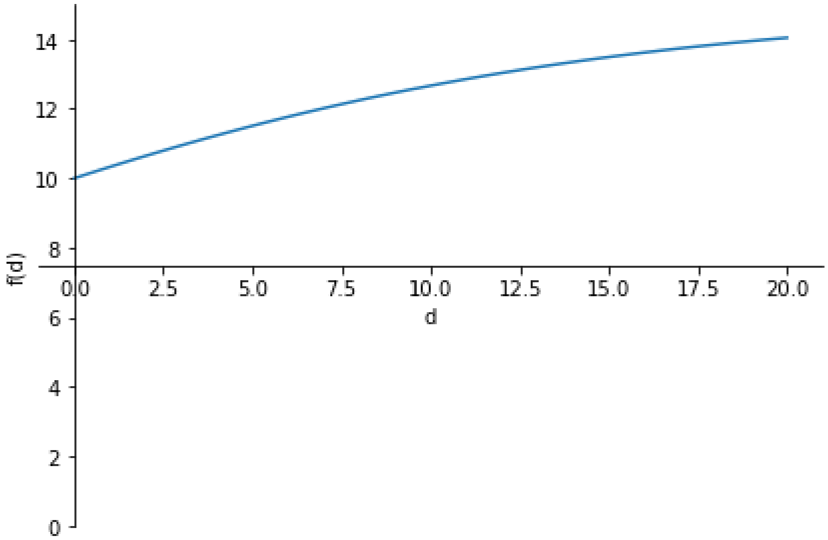
Out[29]: <sympy.plotting.plot.Plot at 0x2502a995460>

In [30]:
```
#limit
print(f"The limit of P(t) where K = 1 as t approaches infinity is {limit(expr2.subs([(
```

```
The limit of P(t) where K = 1 as t approaches infinity is 1.
```

3c

In [31]:
```
#graph
plot(expr2.subs([(a, 15), (b, 10), (c, 0.1)]), (d, 0, 20), ylim = (0, 15))
```



Out[31]: <sympy.plotting.plot.Plot at 0x2502aa98760>

In [32]:
```
#limit
print(f"The limit of P(t) where K = 15 as t approaches infinity is {limit(expr2.subs([
```

```
The limit of P(t) where K = 15 as t approaches infinity is 15.
```

3d

In [33]:
```
#observation on K
print("As the variable t goes to infinity, population size goes to whatever value vari
```

```
As the variable t goes to infinity, population size goes to whatever value variable K
is. From this we can infer that the variable K represents the maximum population size
of the model.
```

In [ ]: