MATH 152 Lab 3

Blake Stewart and Samuel Molero

In [2]:
```
import sympy as sp
from sympy.plotting import (plot,plot_parametric)
```

Instructions: Complete the lab assignment in your assigned groups. Unless stated otherwise, your answers should be obtained using Python code.

Do not modify the cell above, as it contains all the packages you will need. It is highly recommended to not use any additional packages.

NOTE: If you took MATH 151 last semester, notice that the import statement for SymPy is different- for each SymPy command you use, you have to preface it with "sp." For example, "symbols('x')" becomes "sp.symbols('x')". **Except for plot and plot_parametric**- you don't need to type "sp." for those.

ANOTHER NOTE: Approximate answers are acceptable for all non-plotting parts of this week's lab.

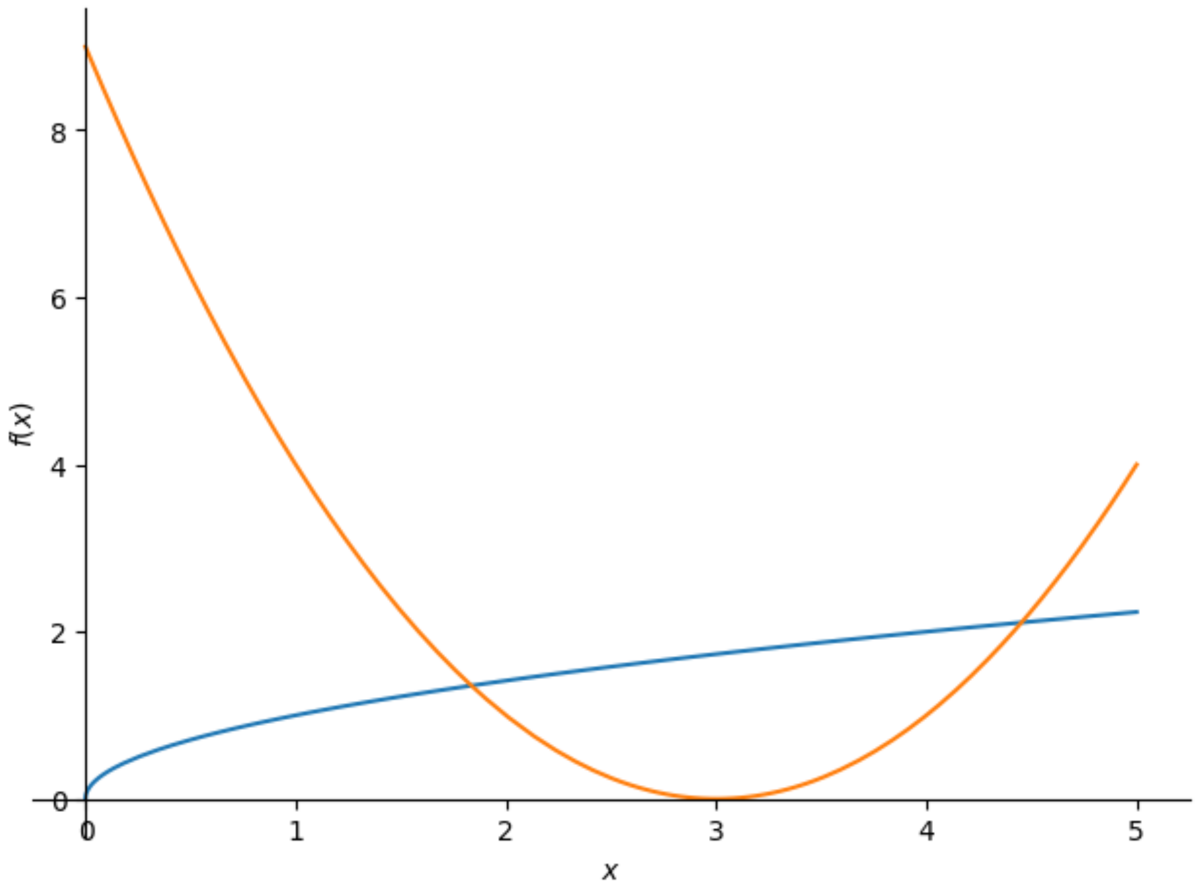
Question 1

1a

In [2]:
```
matplotlib notebook
```

In [18]:
```
# Enter your Python code here
x = sp.symbols('x')
f_x = sp.sqrt(x)
g_x = (x-3)**2

plot(f_x, g_x, (x, 0, 5))
```



Out[18]: <sympy.plotting.plot.Plot at 0x7264b770a860>

1b

In [19]:
```
# Enter your Python code here
#washer method:
bounds = sp.solve(g_x - f_x)
ac = [u.evalf() for u in bounds]
washer = sp.integrate(2*sp.pi*((x-1)- (f_x-g_x)), (x,ac[1],ac[0]))
print('The value of the region by the two curves about x = 1 is: ')
display(washer)
```

The value of the region by the two curves about x = 1 is:

5.10622971522477π

1c

In [20]:
```
# Enter your Python code here
bounds = sp.solve(g_x - f_x)
ac = [u.evalf() for u in bounds]
washer_2 = sp.integrate(sp.pi*((g_x-4)**2 - (f_x-4)**2),(x,ac[1],ac[0]))
print('The value of the region by the two curves about y = 4 is: ')
display(washer_2)
```

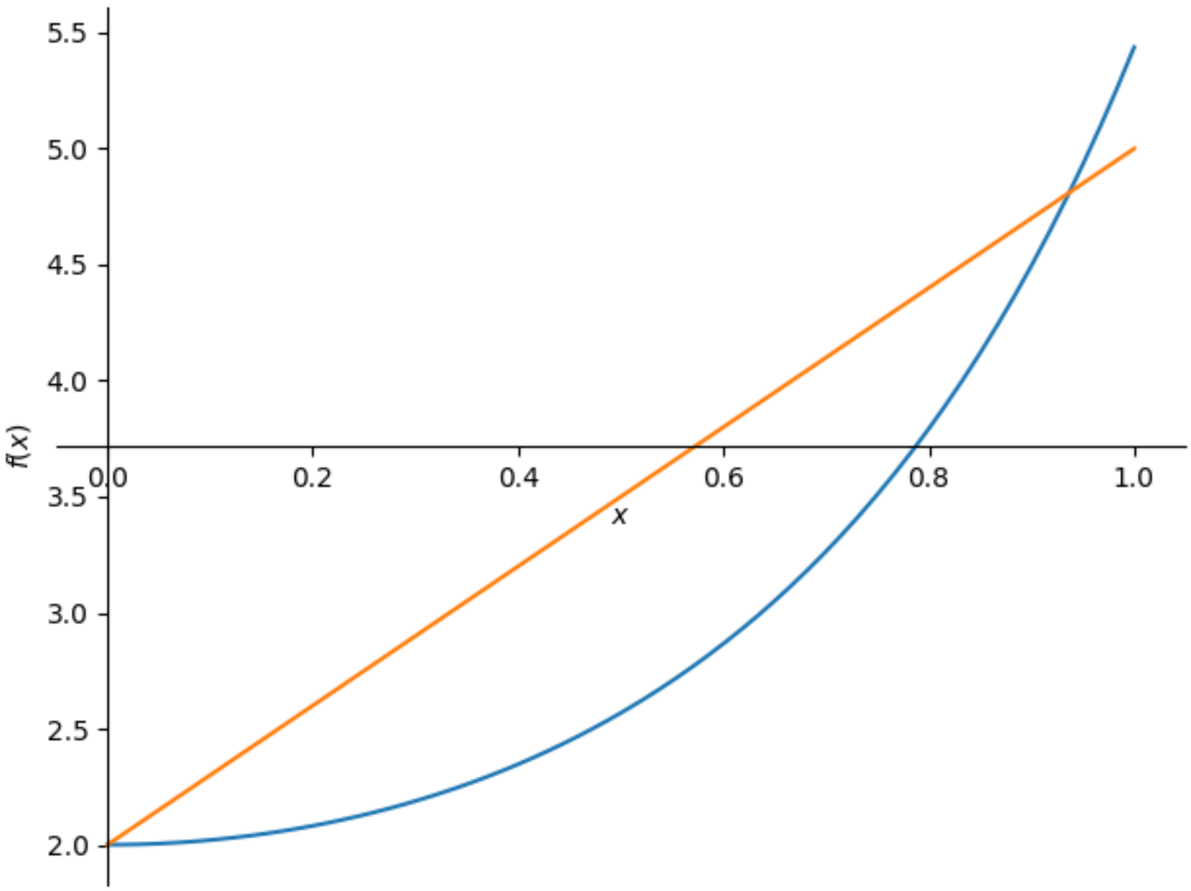
The value of the region by the two curves about y = 4 is:

17.9557286519222π

Question 2

2a

In [50]:
```
# Enter your Python code here
f_x = 2*sp.exp(x**2)
g_x = 3*x+2
plot(f_x, g_x, (x, 0, 1))
```



Out[50]: <sympy.plotting.plot.Plot at 0x7a886b29a170>

2b

In [66]:
```
# Enter your Python code here
bound_1 = sp.nsolve(g_x - f_x,0.9)
bound_2 = sp.nsolve(g_x - f_x,0.0)

volume = sp.integrate((sp.sqrt(3)/4) * (g_x - f_x)**2,(x,bound_2,bound_1))
display(volume.evalf())
```

0.191894750193957

Question 3

3a

In [11]:
```
# Enter your Python code here
y, h= sp.symbols('y h', real =True, positive = True)
pg = 900*9.8*sp.pi
formula = (22-y)*(400-y**2)
work = sp.integrate(formula,(y,-20,20))
print('The amount of work needed is:')
display(pg*work)
```

```
The amount of work needed is:
```

$$2069760000.0\pi$$

3b

In [16]:
```
# Enter your Python code here
formula = (42-y)*(40*y-y**2)
work = sp.integrate(formula,(y,0,40-h))
print('The amount of work needed to have a remaining oil of depth h meters i
display(pg*work)
```

```
The amount of work needed to have a remaining oil of depth h meters is:
```

$$8820.0\pi\left(\frac{(40-h)^4}{4} - \frac{82(40-h)^3}{3} + 840(40-h)^2\right)$$