Midterm-Version1

November 4, 2023

Midterm Exam Version 1- Recitation MATH 102 - PYTHON

1 Question 1

The estimate of the area under the graph of $f(x) = \frac{37}{23x^2+1}$ from $x = 0$ to $x = 3$, using $n = 15$ approximating rectangles with right end-points is:

[1]:
```
import sympy as smp   #Here 'as' means that we type 'smp' whenever we want to␣
 ↪recall this library
import numpy as np    #Here 'as' means that we type 'np' whenever we want to␣
 ↪recall this library

# We now define the variables and the function
x=smp.symbols('x', real=True)
f = smp.lambdify(x, 37/(1 + 23*x**2), "numpy")
# The lambdify keyword is used to allow the function to take values over
# an array of values, namely, xvalues as you are going to see below.
# This is like the substitution but more straightforward
a=0; b=3; n=15

# You can now display them
display(f(x))
print("a=",a)
print("b=",b)
print("n=",n)
```

$$\frac{37}{23x^2 + 1}$$

```
a= 0
b= 3
n= 15
```

[2]:
```
# We now define the line-space for x and then the line space for the values of␣
 ↪the function f at each x.
xvalues = np.linspace(a, b, n+1)
yvalues = f(xvalues) #You can check the sample points by using display(xvalues)

# Now we evaluate the width of subinterval, midpoints and then the Riemann sums.
```

```
deltax = (b-a)/n # The equal width of the each subinterval
#midpoints= (xvalues[1:]+xvalues[:-1])/2 # Array of midpoints
#display(midpoints) # To display the midpoints
#yvalues_M = f(midpoints) # The values of the function on midpoints
#Re_L = (np.sum(yvalues[:-1])) * deltax # Reimann Sum when the samples are the␣
↪left-end points
Re_R = (np.sum(yvalues[1:])) * deltax # Reimann Sum when the samples are the␣
↪right-end points
#Re_M = (np.sum(yvalues_M)) * deltax # Reimann Sum when the samples are the␣
↪midpoints
#print("Approximating area using left-end points is ", Re_L)
print("Approximating area using right-end points is ", Re_R)
#print("Approximating area using midpoints is ", Re_M)
```

```
Approximating area using right-end points is  7.935459178855537
```

2 Question 2

If the acceleration of a moving particle is

$$a(t) = 29t^2 - 43t + 11,$$

with initial velocity $v(0) = 7$, the TOTAL distance traveled by the particle when $0 \leq t \leq 13$ is:

[3]:
```
# (a) First we find the velocity as the antiderivative of the acceleration␣
↪function
t = smp.Symbol('t', real=True)
c = smp.Symbol('c') # We need to define the constant of integration
a=smp.Function('a')(t)
v =smp.Function('v')(t)
a= 29*t**2 -43*t+11
display(a)
v =smp.integrate(a)+c # Notice we added the constant of integration c
display(v)
```

$29t^2 - 43t + 11$

$c + \frac{29t^3}{3} - \frac{43t^2}{2} + 11t$

[4]:
```
v.subs(t,0)  # Use the command sub() to get the velocity at t=0
c1 =smp.solve(v.subs(t,0)-7,c)[0] # We write the equation as v(0)-6=0 and solve␣
↪with respect to c.
# IMPORTANT: [0] means we take only the entry instead of a the whole list of␣
↪solutions.
display(c1)
v = v.subs(c,c1)
display(v)
```

7

$$\frac{29t^3}{3} - \frac{43t^2}{2} + 11t + 7$$

[5]:
```
d=smp.integrate(smp.Abs(v), (t,0,13))
display(d)
display(float(d))
```

$$\frac{217191}{4}$$

54297.75

3 Question 3

Considering the function $f(x) = 7x^2 - 8x - 1$ on the interval $[-10, 20]$, the value(s) of c such that $f_{ave} = f(c)$ is (are):

[6]:
```
# PART 1
import numpy as np   #Here 'as' means that we type 'smp' whenever we want to␣
 ↪recall this library
import sympy as smp   #Here 'as' means that we type 'smp' whenever we want to␣
 ↪recall this library
from sympy import Symbol, Function, lambdify, sqrt, exp , integrate # we import␣
 ↪these functions from smp

x = smp.Symbol('x', real=True) # Set x as a real variable
f = smp.Function('f')(x) # define f as a function of x
f = 7*x**2-8*x-1 # The expression of f
display(f) # To check whether is the correct expression of f
a=-10 # Lower limit of the integral
b=20 # Upper limit of the integral

# value takes the results of the integration of f over the given interval [a,b]

value=smp.integrate(f,(x,a,b))
display(value)
```

$$7x^2 - 8x - 1$$

19770

[7]:
```
# We need to compute the average value of f
# NOTE: Make sure that b > a here

delta=0 # Initialize the variable delta

if (b <=a ):
    print("Error: value of b <= a")
else:
```

```
    delta=b-a
    ave=value/delta
    display("Average value of f = ", ave)
```

'Average value of f = '

659

[8]:
```
# Part 2
# Define c as the real solution to f(c)=ave
c = smp.Symbol('c', real=True)

# Define the solist of solutions that solve f(c)=ave
solist =smp.solve(f.subs(x,c)-ave,c)

# Print the items in solist
display(solist)
print(solist)
```

[4/7 - 2*sqrt(1159)/7, 4/7 + 2*sqrt(1159)/7]

[4/7 - 2*sqrt(1159)/7, 4/7 + 2*sqrt(1159)/7]

Note there are two elements in solist.

[9]:
```
# We can autmatize this and display the value of c only if it meets our␣
 ↪conditions

for i in range(len(solist)):
    if ((solist[i] >=a) & (solist[i]<=b)):
        display("Possible value of c = ", solist[i])
```

'Possible value of c = '

$$\frac{4}{7} - \frac{2\sqrt{1159}}{7}$$

'Possible value of c = '

$$\frac{4}{7} + \frac{2\sqrt{1159}}{7}$$

4 Question 4

If we use an appropriate substitution first, then evaluate the integral:

$$\int_0^5 \frac{111x dx}{\sqrt{317x^2+1}},$$

we get:

[110]:
```
# The command transform from Sympy will substitute an expression in the␣
↪integral, with
# the corresponding limits, by a variable.
i= smp.Integral(111*x / smp.sqrt(1 + 317*x**2),(x,0,5)) # We use the command␣
↪Integral to produce an unevaluated integral
display(i)

# Define u
u = smp.Symbol('u', real=True)

k=i.transform(317*x**2+1,u) # This will substitute 3*x+5 by u and changes the␣
↪limits of integration accordingly
display(k)
```

$$\int\limits_0^5 \frac{111x}{\sqrt{317x^2+1}}\, dx$$

$$\int\limits_1^{7926} \frac{111}{634\sqrt{u}}\, du$$

[111]:
```
# The integral value can now be found

smp.simplify(k)
display(float(k))
```

30.82371446574263

[125]:
```
val=float(k)+7926-2
val
```

[125]: 7954.823714465742

[126]:
```
valb=float(k)+2*7926-1
valb
```

[126]: 15881.823714465743

[128]:
```
valc=float(k)+3*7926-1
valc
```

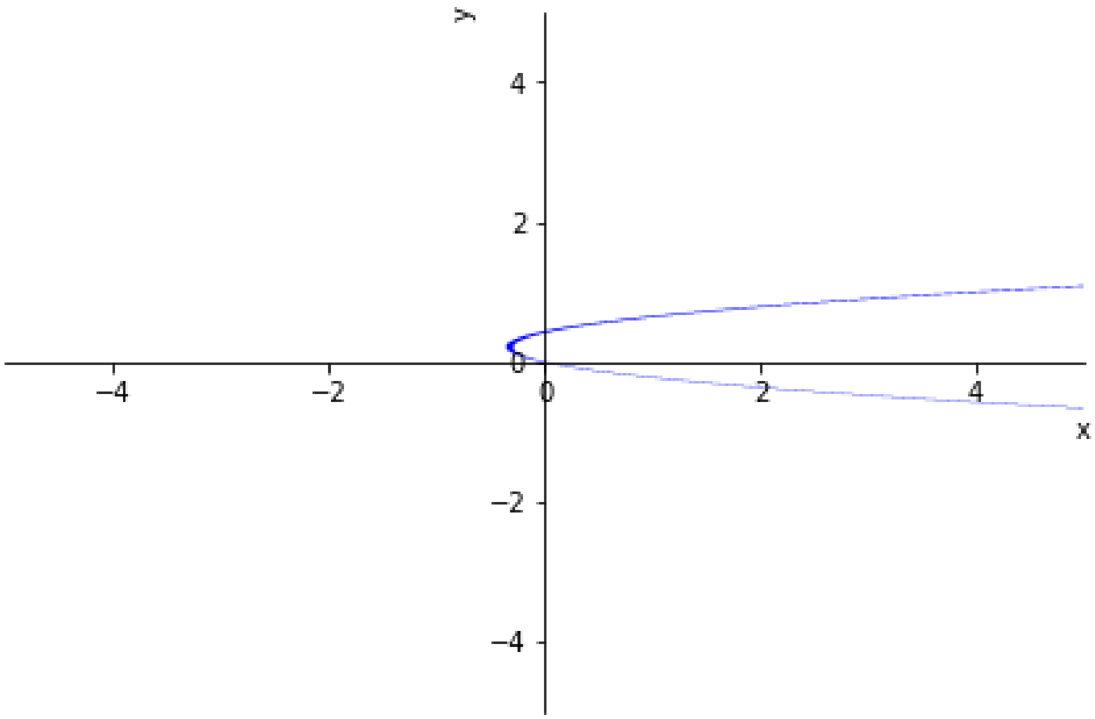
[128]: 23807.823714465743

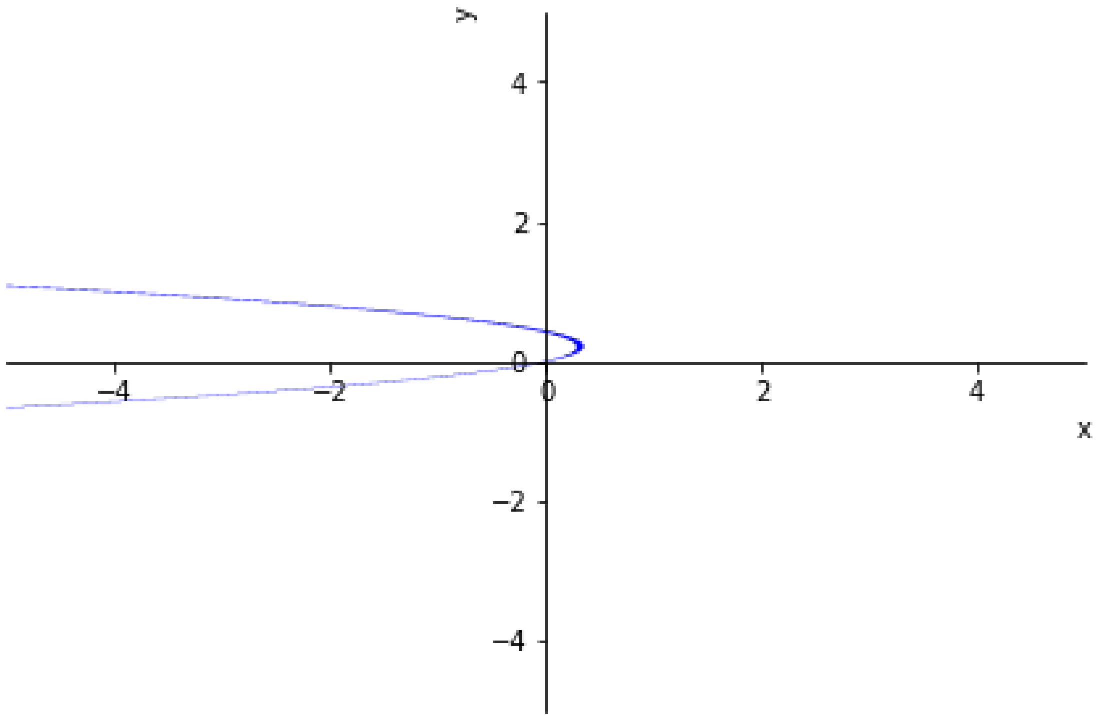
5 Question 5

The area of the region enclosed between the curves $x = 7y^2 - 3y$ and $x = 3y - 7y^2$ is equal to:

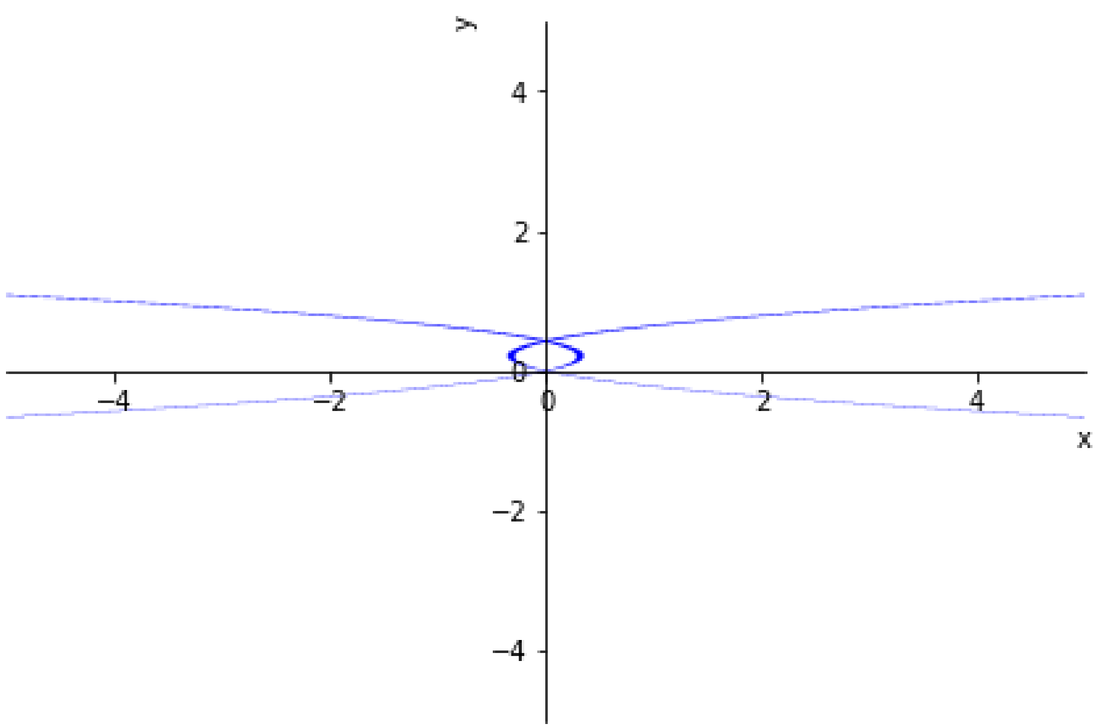
[116]:
```
from sympy import *

x, y = symbols("x y", real=True)
# store the intersection points in t
sol = solve((x-7*y**2+3*y, x-3*y+7*y**2), x, y)
display(sol)
p1 = plot_implicit(x-7*y**2+3*y)
p2 = plot_implicit(x-3*y+7*y**2)
```

```
[(0, 0), (0, 3/7)]
```

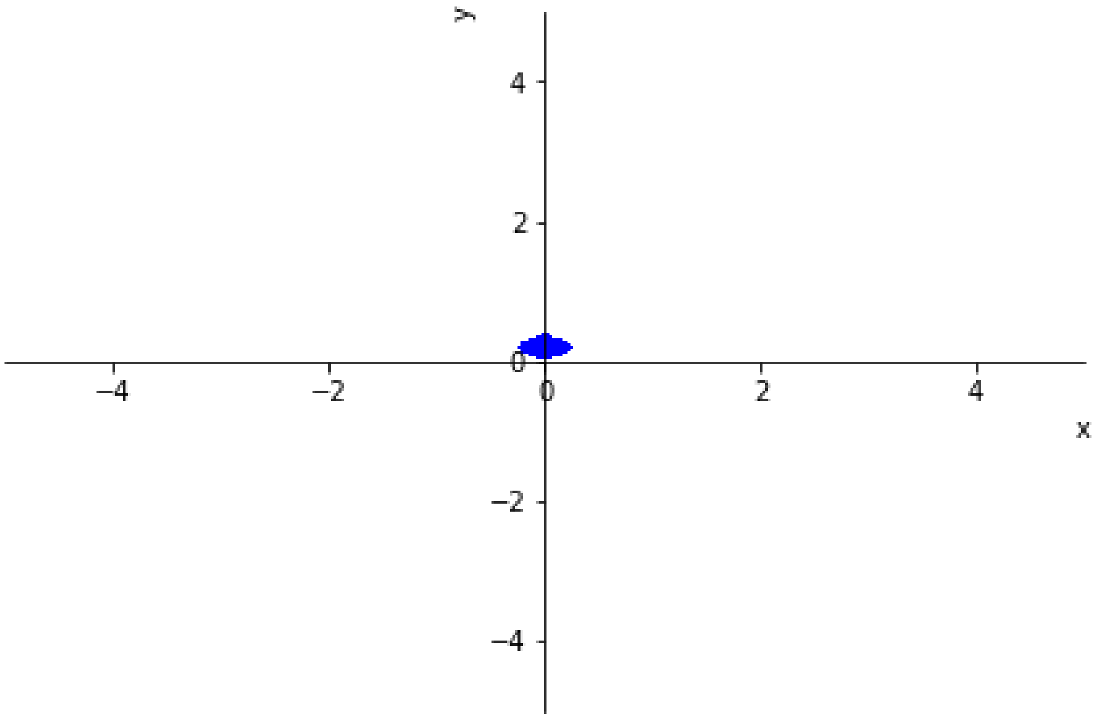




[113]:
```
p1.extend(p2)
p1.show()
```



[114]:
```
plot_implicit(And(x<3*y-7*y**2, x>7*y**2-3*y))
```



[114]: <sympy.plotting.plot.Plot at 0x199e9408ee0>

[115]:
```
integrate((3*y-7*y**2)-(7*y**2-3*y),(y,0,3/7))
```

[115]: 0.183673469387755

5.0.1 Question 6

If $f(x) = 23415\ coh^{-1}(3x)$, then $\int_{\frac{1}{3}}^{\frac{2}{3}} f(x)dx =$

[118]:
```
f=23415*smp.acosh(3*x)
i= smp.Integral(f,(x,1/3,2/3)) # We use the command Integral to produce an␣
↪unevaluated integral
display(i)
simplify(i)
```

$$\int\limits_{0.333333333333333}^{0.666666666666667} 23415 \operatorname{acosh}(3x)\, dx$$

[118]:

7039.0562179213

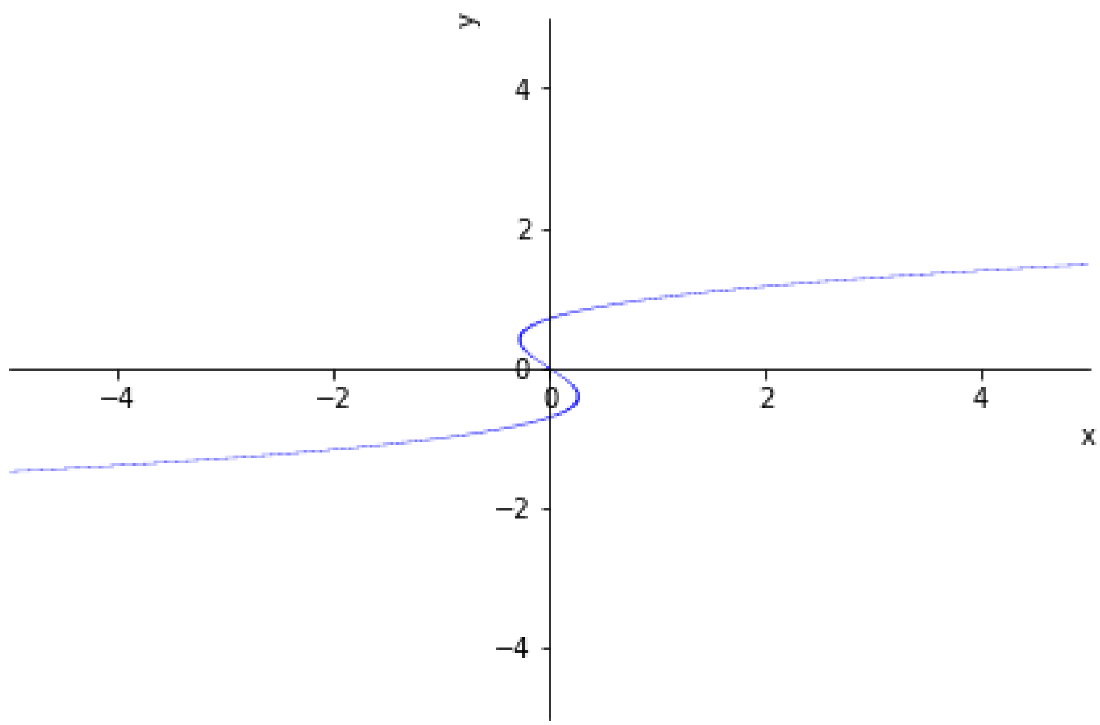
6 Question 7

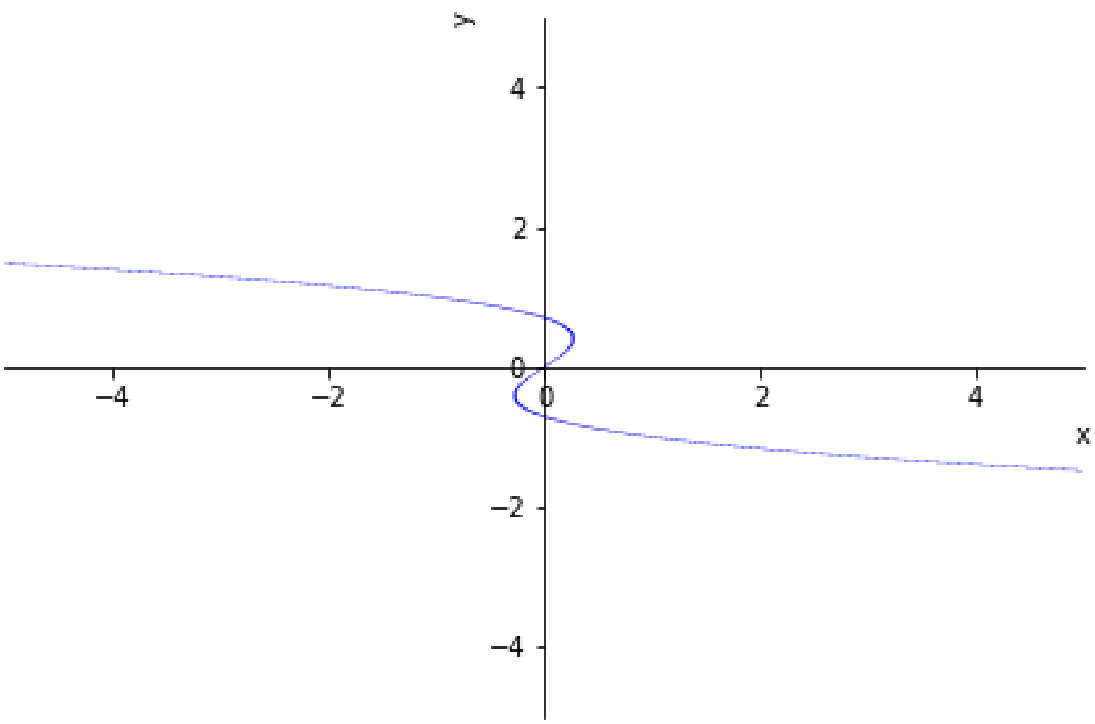
The area of the region enclosed between the curves $x = 2y^3 - y$ and $x = y - 2y^3$ is equal to:

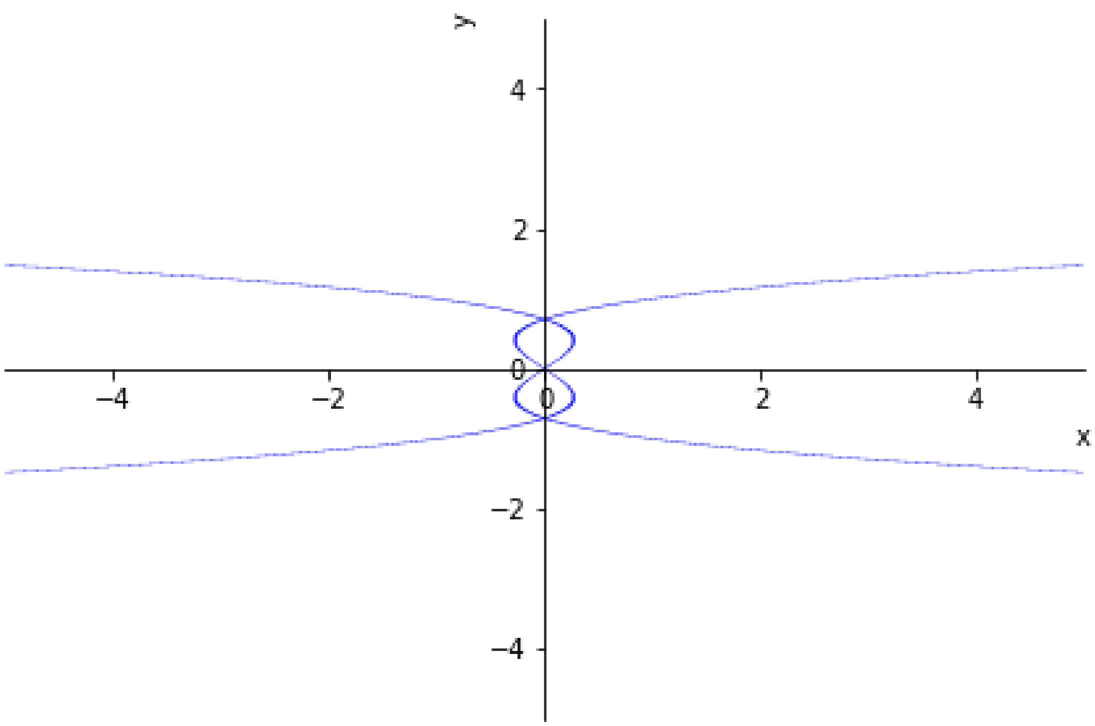
[119]:
```
x, y = symbols("x y", real=True)
# store the intersection points in sol
sol = solve((x-(2*y**3-y), x-(y-2*y**3)), x, y)
display(sol)
```

```
[(0, 0), (0, -sqrt(2)/2), (0, sqrt(2)/2)]
```

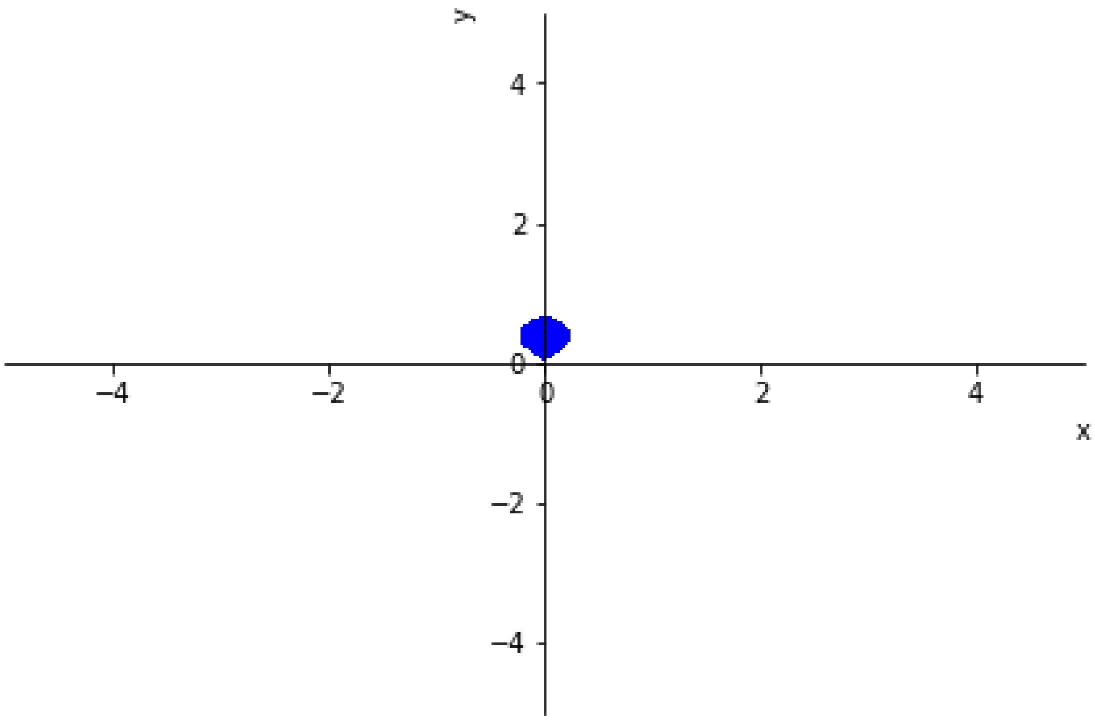
[120]:
```
p1 = plot_implicit(x-(2*y**3-y))
p2 = plot_implicit(x-(y-2*y**3))
p1.extend(p2)
p1.show()
```




y
4
2
0
−4
−2
0
2
4
x
−2
−4


y
4
2
0
−4
−2
0
2
4
x
−2
−4

[123]:
```
plot_implicit(And(x>2*y**3-y, x<y-2*y**3, y>0))
```



[123]: <sympy.plotting.plot.Plot at 0x199e81b90d0>

[124]:
```
A=2*integrate(2*y - 4*y**3,(y,0,sqrt(2)/2))
display(A)
```

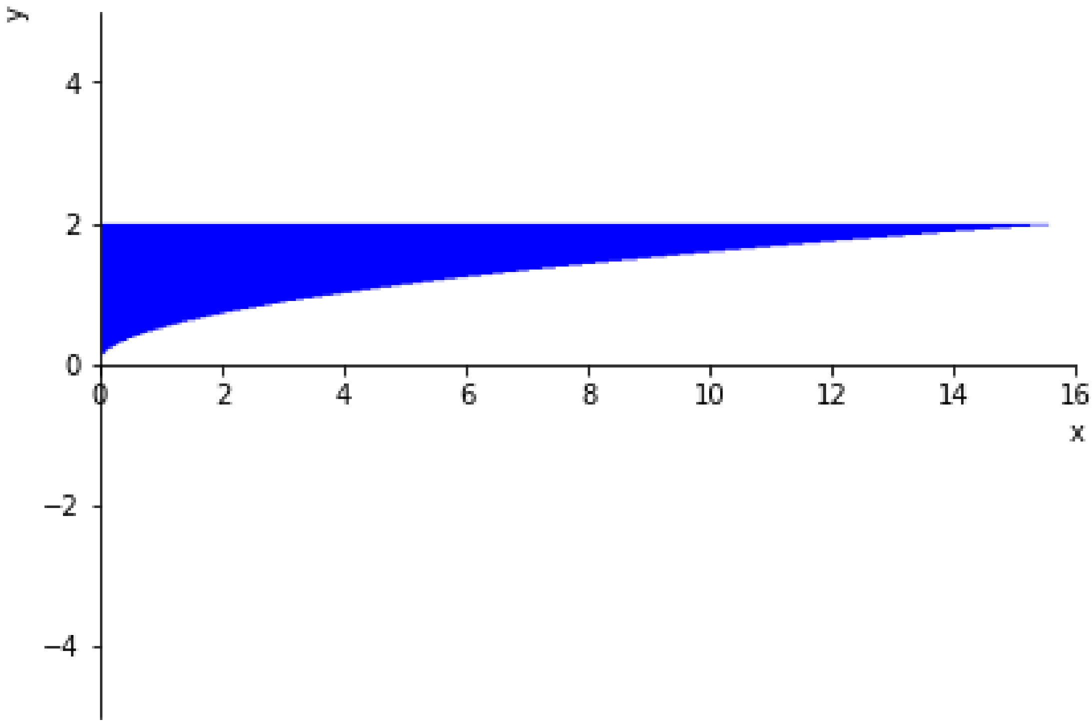
$$\frac{1}{2}$$

7 Question 8

The volume of the solid obtained by rotating the region bounded by the curves $3x = 12y^2$, $x = 0$ and $y = 2$ about the y-axis is given by:

Sketch the region.

[91]:
```
p3 = plot_implicit(And(y < 2 , 12*y**2 - 3*x > 0, y> 0, x>0), (x,0,16))
```



[41]:
```
r = (4*y**2)
vol = pi*integrate(r**2, (y,0,2))
display(vol)
```

$$\frac{512\pi}{5}$$

[ ]:
```
```