Math 151 – Python Lab 2

September 15, 2022

0.1 MATH 151 Lab 2

Section Number: 568

Members:

- Brighton Sikarskie
- Colton Hesser
- Gabriel Gonzalez
- Gabriel Cuevas

[1]:
```
from sympy import *
from sympy.plotting import (plot,plot_parametric, PlotGrid)
%matplotlib inline
```

0.1.1 Question 1

1a

[2]:
```
x = symbols('x')
f = 1 + x * (x + 1) * (x + 2) * (x + 3)
solved_values = solve(f, x)
print(f"All of the values for f(x) = {f} when f(x) = 0 are {solved_values[0]}␣
↪and {solved_values[1]}.")
```

```
All of the values for f(x) = x*(x + 1)*(x + 2)*(x + 3) + 1 when f(x) = 0 are
-3/2 - sqrt(5)/2 and -3/2 + sqrt(5)/2.
```

1b

[3]:
```
print(f"For the expression f(x) = {f}, f(x) in expanded form is {expand(f)}.")
```

```
For the expression f(x) = x*(x + 1)*(x + 2)*(x + 3) + 1, f(x) in expanded form
is x**4 + 6*x**3 + 11*x**2 + 6*x + 1.
```

1c

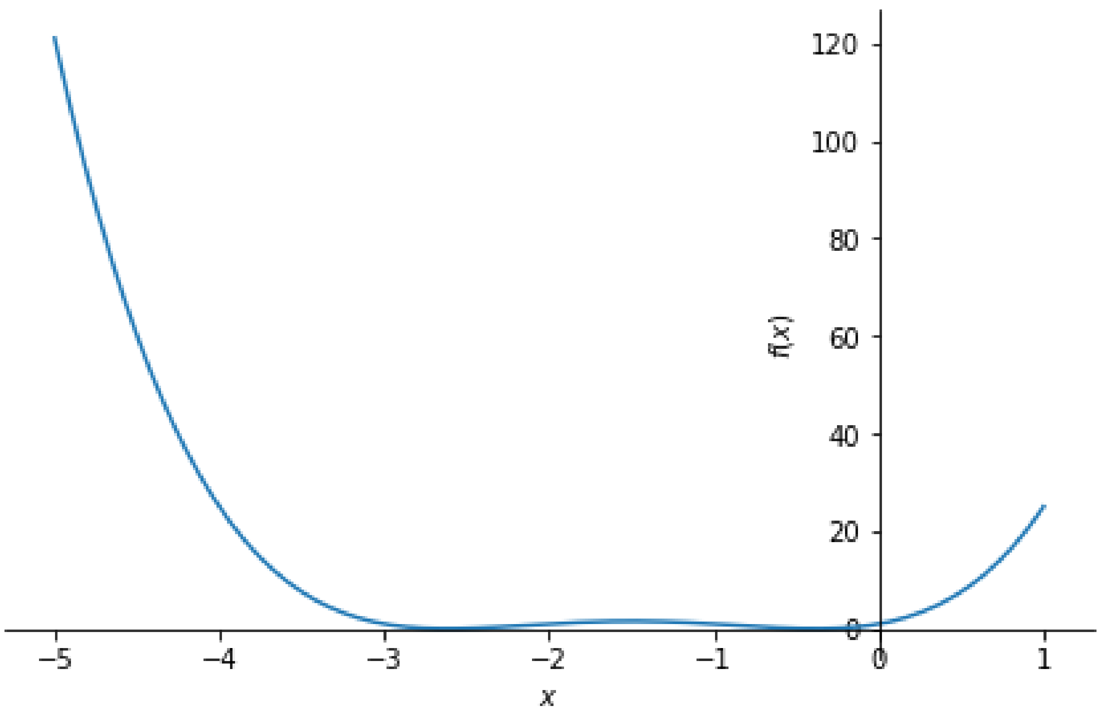
[4]:
```
print(f"For the expression f(x) = {f}, f(x) in factored form is {factor(f)}.")
```

```
For the expression f(x) = x*(x + 1)*(x + 2)*(x + 3) + 1, f(x) in factored form
is (x**2 + 3*x + 1)**2.
```

1d

```
[5]: print(f"For the expression f(x) = {f}, f(x) plotted on the window where x is an element of [-5, 1] is shown below and it does agree with the values we found in part (a) and the expression we found in part (c).")
plot(f, (x, -5, 1))
```

```
For the expression f(x) = x*(x + 1)*(x + 2)*(x + 3) + 1, f(x) plotted on the
window where x is an element of [-5, 1] is shown below and it does agree with
the values we found in part (a) and the expression we found in part (c).
```




```
[5]: <sympy.plotting.plot.Plot at 0x7f57cae59580>
```

0.1.2 Question 2

2a

```
[6]: g = 9.8
s_o = 6
t = Symbol('t')
s_0 = (-1 / 2) * g * (t ** 2) + s_o
time = solve(s_0, t)[1]

print(f"When supposing that a textbook is dropped from a window 6 meters off the ground with gravity being 9.8 m/s^2 and iggnoring air restistance, we can use the formula s = {s_0} to conclude that the textbook hits the ground at t = {time} seconds.")
```

When supposing that a textbook is dropped from a window 6 meters off the ground with gravity being 9.8 m/s^2 and iggnoring air restistance, we can use the formula s = 6 - 4.9*t**2 to conclude that the textbook hits the ground at t = 1.10656667034498 seconds.

2b

[7]:
```
R = 2
s_1 = -R * g * t - (R ** 2) * g * (E ** (-t / R)) + (R ** 2) * g + s_o
time = solve(s_1, t)[1]

print(f"When supposing that a textbook has an air resistance constant of {R},␣
↪we can use the formula s = {s_1} to conclude that the textbook hits the␣
↪ground at t = {time} seconds.")
```

When supposing that a textbook has an air resistance constant of 2, we can use the formula s = -19.6*t + 45.2 - 39.2*exp(-t/2) to conclude that the textbook hits the ground at t = 1.21873095569557 seconds.

2c

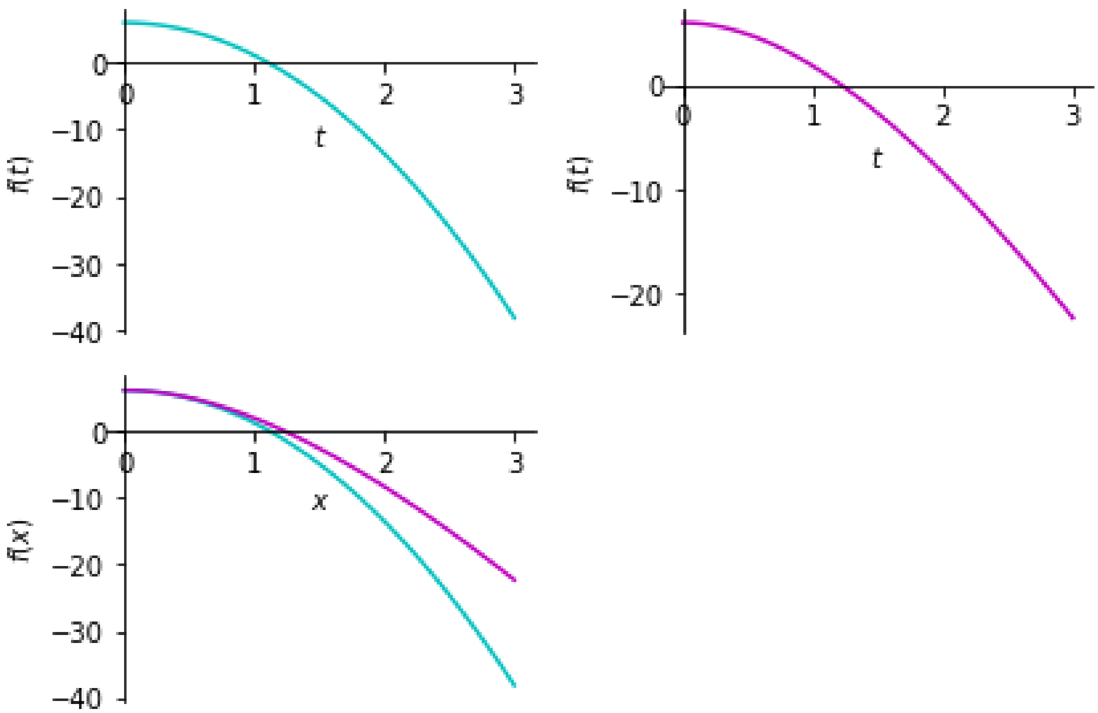
[8]:
```
plot_0 = plot(s_0, (t, 0, 3), show=False)
plot_1 = plot(s_1, (t, 0, 3), show=False)
plot_0[0].line_color = 'c'
plot_1[0].line_color = 'm'

plot_extended = plot(show=False)
plot_extended.extend(plot_0)
plot_extended.extend(plot_1)

print(f"The expressions s = {s_0} and s = {s_1} plotted on the domain where t␣
↪is an element of [0, 3] are shown below. The expression s = {s_0} is the␣
↪textbook falling without air resistance which is shown with the cyan line␣
↪while the expression s = {s_1} is the textbook falling with air resistance␣
↪is shown with the magenta line.")

PlotGrid(2, 2, plot_0, plot_1, plot_extended)
```

The expressions s = 6 - 4.9*t**2 and s = -19.6*t + 45.2 - 39.2*exp(-t/2) plotted on the domain where t is an element of [0, 3] are shown below. The expression s = 6 - 4.9*t**2 is the textbook falling without air resistance which is shown with the cyan line while the expression s = -19.6*t + 45.2 - 39.2*exp(-t/2) is the textbook falling with air resistance is shown with the magenta line.




```
[8]: <sympy.plotting.plot.PlotGrid at 0x7f57c84b5a60>
```

0.1.3 Question 3

3a

```
[9]: plot_extended = plot(show=False)
# curve 1
x1 = sin(2*t)
y1 = cos(t)
plot_1 = plot(x1, y1, show=False)
plot_1[0].line_color = 'c'
plot_1[1].line_color = 'c'

# curve 2
x2 = 3 * sin(2*t)
y2 = cos(t)
plot_2 = plot(x2, y2, show=False)
plot_2[0].line_color = 'm'
plot_2[1].line_color = 'm'

# curve 3
x3 = sin(3 * t)
y3 = 2 * cos(t)
```

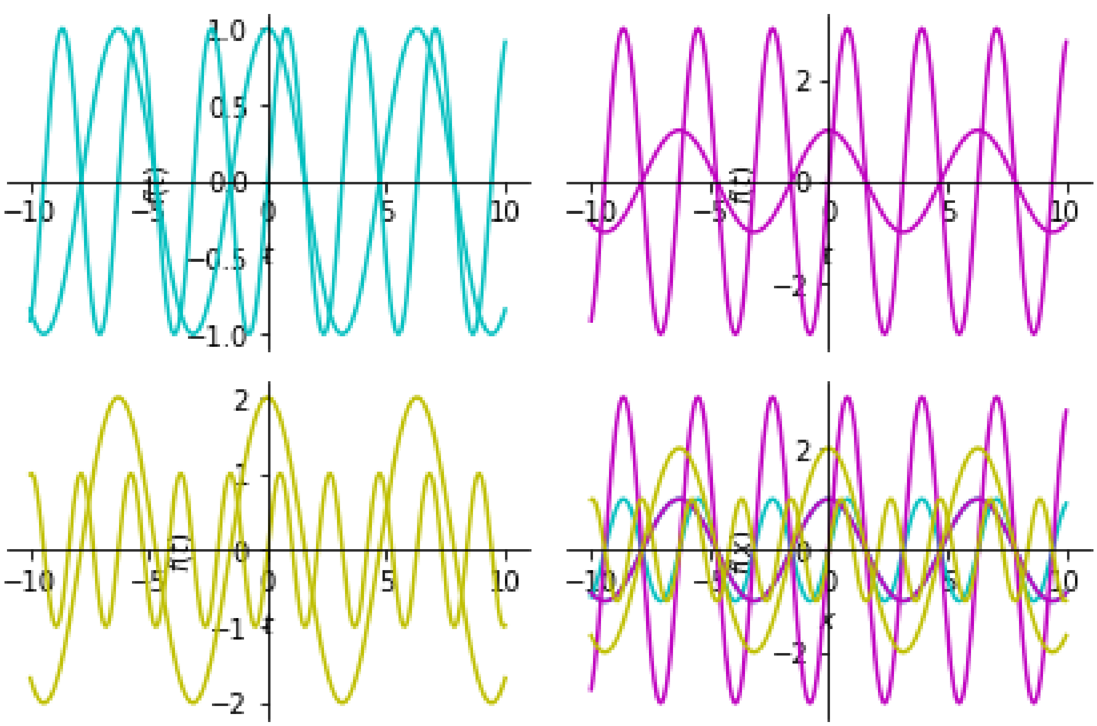
```
plot_3 = plot(x3, y3, show=False)
plot_3[0].line_color = 'y'
plot_3[1].line_color = 'y'


# extend the plot
plot_extended.extend(plot_1)
plot_extended.extend(plot_2)
plot_extended.extend(plot_3)

print(f"Curves of the form of x = a sin(nt) and y = b cos(t) with n a positive␣
 ↪integer are called Lissajous figures.")
print(f"Shown below are examples of Lissajous figures.")
print(f"The curve x = {x1} and y = {y1} is shown below with the cyan line.")
print(f"The curve x = {x2} and y = {y2} is shown below with the magenta line.")
print(f"The curve x = {x3} and y = {y3} is shown below with the yellow line.")

PlotGrid(2, 2, plot_1, plot_2, plot_3, plot_extended)
```

```
Curves of the form of x = a sin(nt) and y = b cos(t) with n a positive integer
are called Lissajous figures.
Shown below are examples of Lissajous figures.
The curve x = sin(2*t) and y = cos(t) is shown below with the cyan line.
The curve x = 3*sin(2*t) and y = cos(t) is shown below with the magenta line.
The curve x = sin(3*t) and y = 2*cos(t) is shown below with the yellow line.
```



[9]: <sympy.plotting.plot.PlotGrid at 0x7f57c7d931f0>

3b As the absolute value of 'a' increases the amplitude of the function increases.

As the absolute value of 'n' increases the period of the function decreases.

As the absolute value of 'b' increases the amplitude of the function increases.