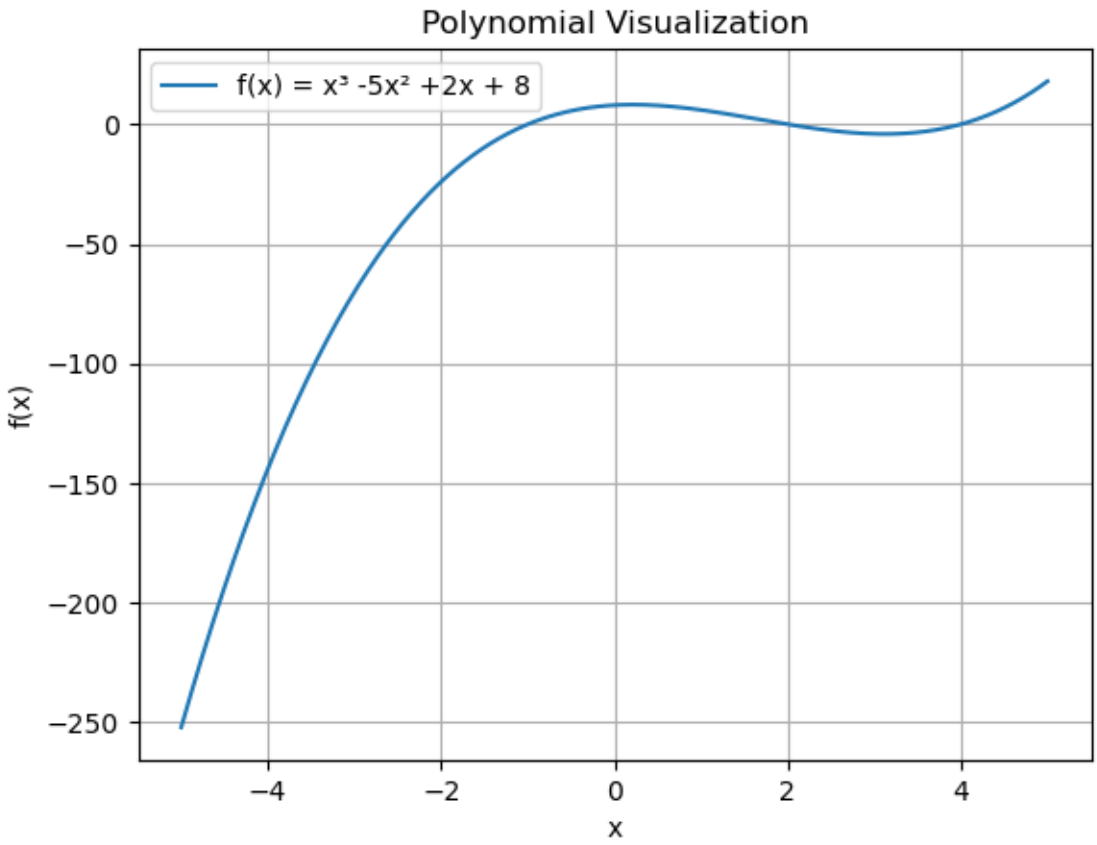
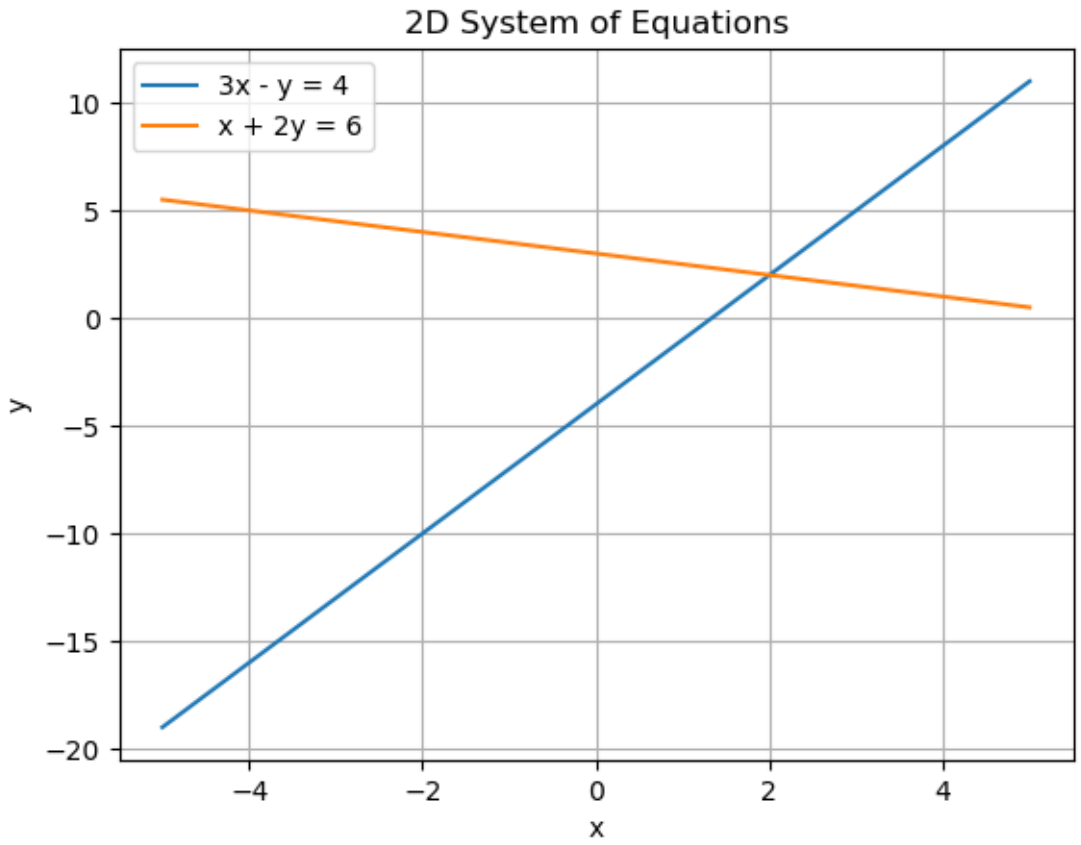
Lab2 Sol

January 31, 2025

[1]:
```
import numpy as np
import matplotlib.pyplot as plt

# Define the domain and polynomial function
x = np.linspace(-5, 5, 500)
f_x = x**3 - 5*x**2 + 2*x+8

# Plot the polynomial
plt.plot(x, f_x, label="f(x) = x³ -5x² +2x + 8")
plt.xlabel("x")
plt.ylabel("f(x)")
plt.title("Polynomial Visualization")
plt.grid()
plt.legend()
plt.show()
```

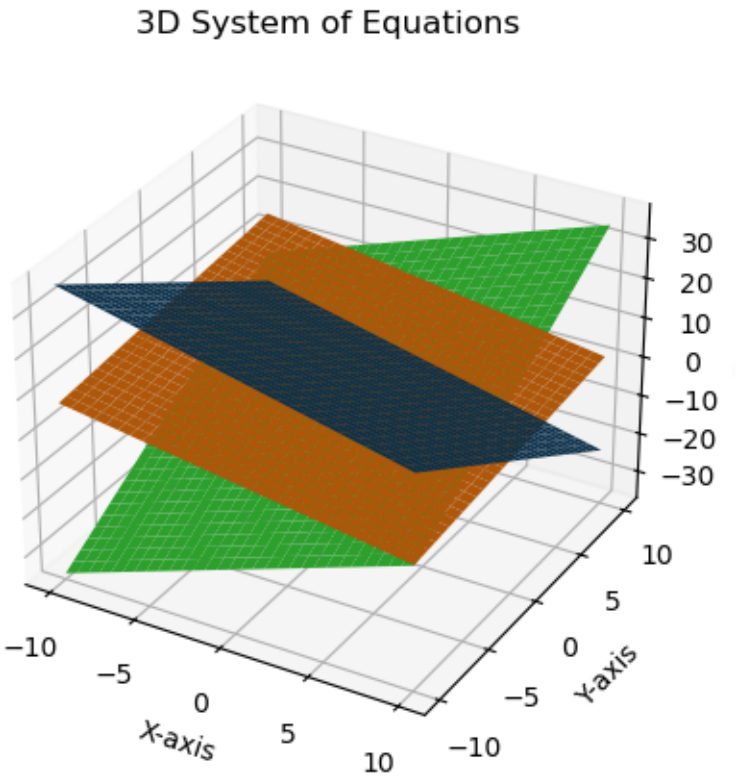


[4]:
```
# Define the domain
x = np.linspace(-5, 5, 500)

# Define the equations
y1 = (3*x - 4)  # 3x - y = 4
y2 = (6 - x)/2  # x + 2y = 6

# Plot the equations
plt.plot(x, y1, label="3x - y = 4")
plt.plot(x, y2, label="x + 2y = 6")
plt.xlabel("x")
plt.ylabel("y")
plt.title("2D System of Equations")
plt.grid()
plt.legend()
plt.show()
```



[5]:
```
from mpl_toolkits.mplot3d import Axes3D

# Define the grid
x = np.linspace(-10, 10, 30)
y = np.linspace(-10, 10, 30)
x, y = np.meshgrid(x, y)

# Define the planes
z1 = 7 - x - 2*y        # x + 2y + z = 7
z2 = (14 - 2*x + y)/3   # 2x - y + 3z = 14
z3 = (1 - 3*x - 4*y)/-2 # 3x + 4y - 2z = 1

# Plot the planes
fig = plt.figure()
ax = fig.add_subplot(projection="3d")
ax.plot_surface(x, y, z1, label="Plane 1")
ax.plot_surface(x, y, z2, label="Plane 2")
ax.plot_surface(x, y, z3, label="Plane 3")

# Set labels
```

```
ax.set_xlabel("X-axis")
ax.set_ylabel("Y-axis")
ax.set_zlabel("Z-axis")
ax.set_title("3D System of Equations")
plt.show()
```



[ ]: