## **Assignment 11 - Manciu Patricia 914**

```
import numpy as np
import matplotlib.pyplot as plt
\# f(x) = 1/2 * x^T * A * x
def f(x, A):
   return 0.5 * np.dot(x.T, np.dot(A, x))
def gradient(x, A):
    return np.dot(A, x)
if __name__ == "__main__":
    # Matrices I chose for:
    # - unique maximum
         [-2, 1], [1, -3]
   # - unique minimum
   # [2, 1], [1, 3]
# - saddle point
        [2, 1], [1, -3]
   matrix = [np.array([[-2, 1], [1, -3]]), np.array([[2, 1], [1, 3]]), np.array([[2, 1], [1, -3]])]
   title = ['Unique Maximum | Manciu Patricia - 914',
             'Unique Minimum | Manciu Patricia - 914',
             'Saddle Point | Manciu Patricia - 914']
    for i in range(len(title)):
       M = matrix[i] # current matrix
       x_values = np.linspace(-5, 5, 100)
       y_values = np.linspace(-5, 5, 100)
       X, Y = np.meshgrid(x_values, y_values)
       points = np.vstack([X.ravel(), Y.ravel()])
       Z = np.array([[f(np.array([x, y]), M) for x in x_values] for y in y_values])
       fig = plt.figure(figsize=(15, 10))
        ax = fig.add_subplot(121, projection='3d')
       ax.plot_surface(X, Y, Z, cmap='inferno')
       contour = ax.contour(X, Y, Z, levels=10, cmap='inferno', offset=np.min(Z), linewidths=2, alpha=0.7,
                             label="Contour Lines")
       ax.clabel(contour, fontsize=8)
       p = np.array([[2, 2], [-2, -2], [4, -4]])
        colors = ["magenta", "purple", "orange"]
        for j in range(3):
            gradient_vector = gradient(p[j], M)
            ax.quiver(p[j][0], p[j][1], f(p[j], M), gradient_vector[0], gradient_vector[1], -3,
                      color=colors[j], arrow_length_ratio=0.05, label=f"Gradient at {p[j][0], p[j][1]}")
       ax.set_title(title[i])
        ax = fig.add_subplot(122)
       contour = ax.contour(X, Y, Z, levels=10, cmap='inferno')
        ax.set_title("Contour " + title[i])
        plt.show()
```









