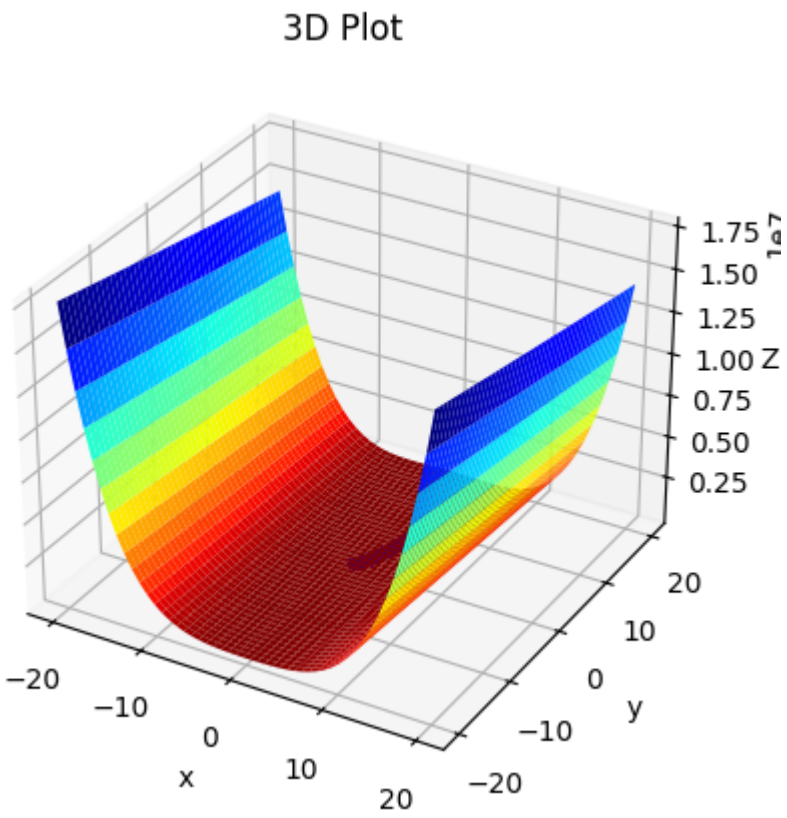
In [ ]:
```
import numpy as np
import matplotlib.pyplot as plt
```

In [ ]:
```
lr = 0.005 # Learning rate
epochs = 1000
eps = 0.0001

def f(x, y, a, b):
    return (a - x)**2 + b*(y - x**2)**2

def gradient_f_x(x, y, a, b):

    return 4*b*x**3 + 2*x - 4*b*x*y -2*a

def gradient_f_y(x, y, a, b):
    return 2*b*y - 2*b*x**2

def draw_progress(Xs, Ys, a, b):
    fig, ax = plt.subplots(subplot_kw={"projection": "3d"})
    reso = 50
    X = np.linspace(-20, 20, reso)
    Y = np.linspace(-20, 20, reso)
    X, Y = np.meshgrid(X, Y)
    Z = (a - X)**2 + b*(Y - X**2)**2
    Zs = (a - Xs)**2 + b*(Ys - Xs**2)**2
    surf = ax.plot_surface(X, Y, Z, cmap='jet_r',
        linewidth=0, antialiased=True, label="search space")
    ax.set_xlabel("x")
    ax.set_ylabel("y")
    ax.set_zlabel('Z', labelpad=1)
    ax.set_title('3D Plot')
    ax.plot(
        Xs, Ys, Zs, '.-b',
        label = 'gradient descent'
    )
    plt.show()

def gradient_decent_booth(x,y, a, b):
    value_before = f(x, y, a, b)
    Xs = [x]
    Ys = [y]
    for i in range(epochs):
        x = x - lr*gradient_f_x(x,y, a, b)
        y = y - lr*gradient_f_y(x,y, a, b)
        x = np.clip(x, -20, 20)
        y = np.clip(y, -20, 20)
        value = f(x, y, a, b)
        Xs.append(x)
        Ys.append(y)
        if (abs(
            value_before - value
        ) < eps):
            draw_progress(np.array(Xs), np.array(Ys), a, b)
            print(f'final x : {x} - final y : {y} - final f(x, y) : {value_before}'
```

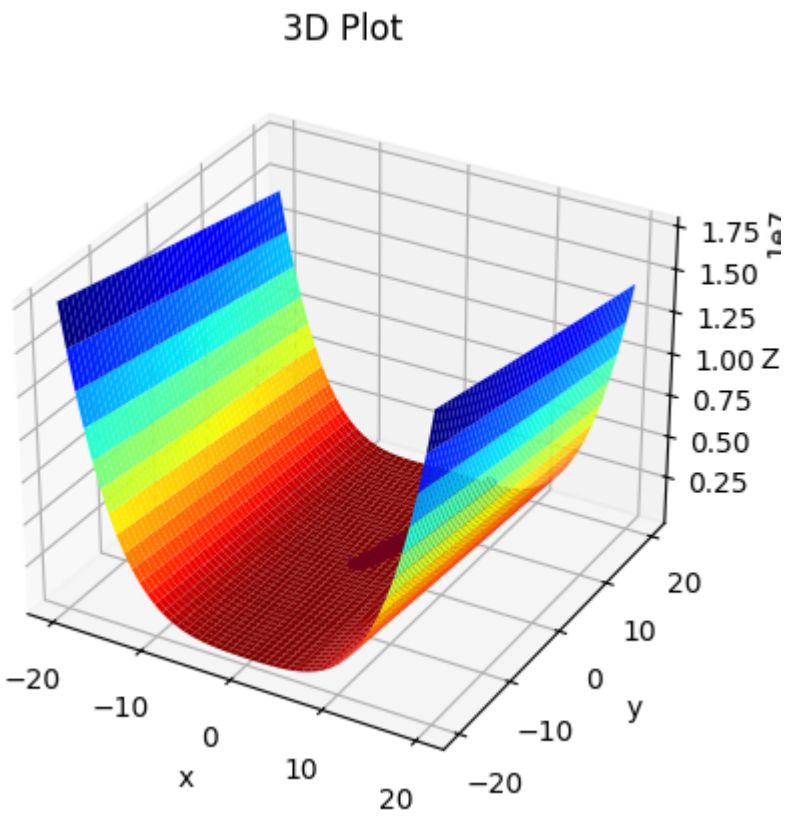
```
            return (x, y)
        value_before = value
    draw_progress(np.array(Xs), np.array(Ys), a, b)
    print(f'final x : {x} - final y : {y} - final f(x, y) : {value_before}')

gradient_decent_booth(0, 0, 3, 100)
```




```
final x : 2.9300666980564105 - final y : 8.585290855059196 - final f(x, y) : 0.00498
9967065332935
```

Out[ ]: (2.9300666980564105, 8.585290855059196)

In [ ]:
```
gradient_decent_booth(0, 0, 10, 100)
```




```
final x : 3.033030927711535 - final y : 9.199276608454694 - final f(x, y) : 48.53865
8054224
```

In [ ]: