Assignment No 4

Title:

Implement Gradient Descent Algorithm to find the local minima of a function. For example, find the local minima of the function $y=(x+3)^2$ starting from the point x=2

Objective:

Objective: Implement the Gradient Descent algorithm to find the local minimum of the function $y = (x + 3)^2$ starting from the point $x = 2$.

Requirements: Python (NumPy library for mathematical operations)

Theory: Gradient Descent Algorithm for Finding Local Minima

Gradient Descent is an optimization algorithm used to find the local minimum of a function. It works by iteratively adjusting the parameters of the function in the direction of steepest descent (negative gradient) until convergence is achieved. The gradient of a function at a specific point indicates the direction of the greatest increase in the function's value.

In mathematical terms, for a function *f*(*x*), the gradient descent update rule is given by:

$$x_{n+1} = x_n - \alpha \cdot \nabla f(x_n)$$

Where:

- x_n is the current value of the parameter x at iteration n.
- α is the learning rate, controlling the step size in each iteration.
- $\nabla f(x_n)$ is the gradient of the function f at x_n.

The process is repeated until a stopping criterion is met, such as reaching a certain number of iterations or when the change in *x* becomes negligible.

Implementation Steps:

1. Open a Python interpreter or create a Python script.
2. Import the required libraries:
3. Define the function f(x)

4. Initialize parameters.
5. Implement the gradient descent algorithm:
6. Print the final result:
7. Run the code and observe the output. You should see the value of x approaching the local minimum of the function $y=(x+3)^2$

Conclusion:

Hence Gradient descent is a foundational optimization algorithm used in machine learning and various other fields to minimize functions and find optimal solutions.