**Lab Session #8**

**Gradient Descent Optimization using python**

**Aim:** To implement gradient descent optimization algorithms in python.

**Problem Definition**: Download the dataset from the following link: <https://www.kaggle.com/sazid28/advertising.csv> . Use the dataset to implement gradient descent algorithms and project the costs for varying values of alpha.

**Theory:** Gradient Descent is an optimisation algorithm which helps you find the optimal weights for your model. It does it by trying various weights and finding the weights which fit the models best i.e. minimizes the cost function. Cost function can be defined as the difference between the actual output and the predicted output. Hence, the smaller the cost function is, the closer the predicted output from your model is to the actual output. Cost function can be mathematically defined as:

𝑦=𝛽+θnXn,

where x is the parameters(can go from 1 to n), 𝛽 is the bias and θ is the weight

While on the other hand, the learning rate of the gradient descent is represented as α. Learning rate is the size of steps taken by each gradient. While a large learning rate can give us poorly optimized values for 𝛽 and θ, the learning rate can also be too small which takes a substantial increment in number of iterations required to get the the convergence point(the optimal value point for 𝛽 and θ). This algorithm, gives us the value of α, 𝛽 and θ as output.