Answer the questions below within **100 minutes**. Use **Visual Studio Code** and submit your answer in an “**ipynb**” file.

1. **Solving Linear Equation**

Solve the following system of linear equations with the following requirements,

* + - You must determine whether the equations are **diagonally dominant programmatically**. **If the equation is not diagonal**, then **print an error message**.
    - If the equations are **diagonally dominant**, use **Gauss-Seidel method** and set **max-iterations** with **15**. **Otherwise**, show a message telling the equations are **not diagonally dominant**.
    - Use a **pre-defined threshold** , where xx is the last two digits of your NIM. If the last two digits of your NIM are **00**, so make the threshold **0.010**.

*Example:*

NIM = 24401073**74**, then the will have a value of

NIM = 24401073**00**, then the will have a value of

* + - Use the value 0 as the initial value of x1, x2, and x3.

Then, show the result for each equation and check whether the **equations** below are **convergent** or **not** and **print** the **value** of **x1**, **x2**, and **x3** in **each iteration**.

Below are the systems of linear equations that you need to solve:

Snippet code for the equations:

Xs = [

  [

    [7, -2, -3],

    [4, -6, 1],

    [1, 2, -4]

  ],

  [

    [12, -4, 2, 3],

    [3, 7, -1, -3],

    [-1, 3, 9, 2],

    [-5, 1, -2, 9]

  ],

  [

    [2, -5, 3],

    [3, 6, 4],

    [-1, -3, 2]

  ],

  [

    [7, -3, -1],

    [2, -5, 1],

    [-3, 2, -8]

  ],

  [

    [7, -1, 5, 1],

    [-1, 4, -1, 1],

    [3, -3, 10, 2],

    [-2, 3, -4, 11]

  ],

  [

    [1, -3, 2],

    [3, -2, 4],

    [2, 1, -2]

  ],

  [

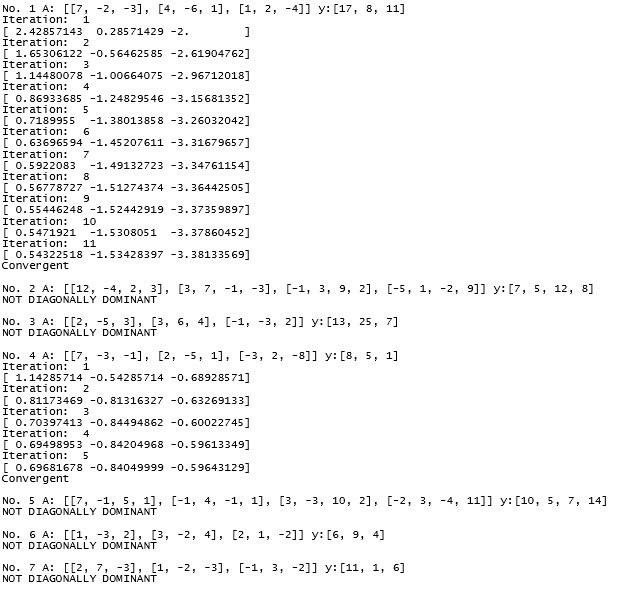
    [2, 7, -3],

    [1, -2, -3],

    [-1, 3, -2]

  ]

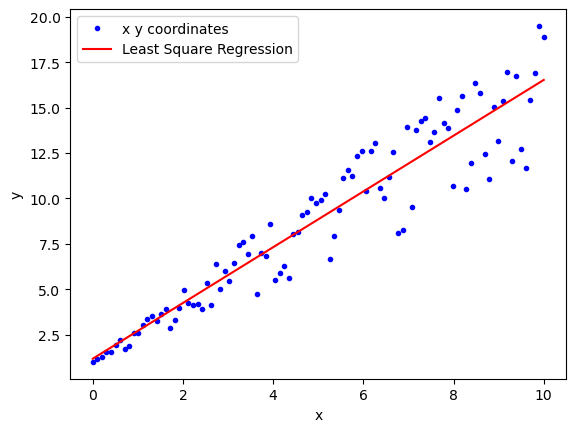
]



**Figure 1. Gauss Seidel Result with Epsilon 0.01**

1. **Regression and Plotting**

Create a **Least Square Regression** from the **matrix (x, y)** in **“matrix.txt”** file, then show the **plotting** result with the **Least Square Regression** using **matplotlib**. You need to show each **plotting** **point** of **matrix (x,y)** and the **Least Square Regression** in the **graphic** with **different** **colors**.



**Figure 2. Least Square Regression Plot Result**

1. **Newton Raphson**

Find the **root** of the equation **f(x)** below by using **Newton Raphson** method:

To find the **root**, here are the requirements you need to know:

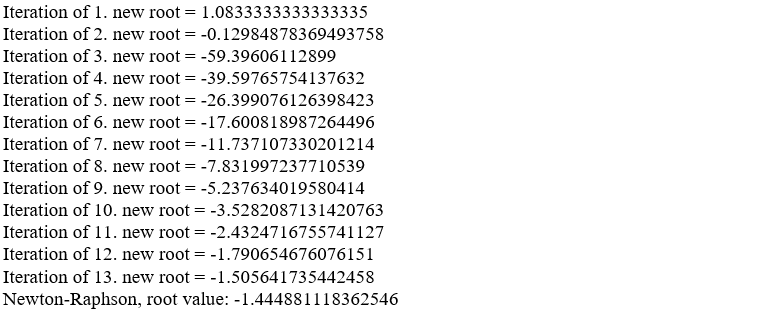
* + **First guess** of the value is **2**.
    - The **tolerate error** , where **xx** is the last two digits of your NIM. If the last two digits of your NIM are **00**, so make the **tolerate error** **0.010**.

*Example:*

NIM = 24401073**74**, then the will have a value of

NIM = 24401073**00**, then the will have a value of

* + The number of **max iterations** is **20**.
  + **Print** the **result** of the **root** in **each iteration**. If the **iteration** is **over** the **max iterations**, **print** an **error message**.



**Figure 3. Newton Raphson Result with Tolerate Error 0.055**

1. **Integral Riemann**

You can freely choose **one of the methods** between **Left Riemann**, **Right Riemann**, **Midpoint Rule**, or **Trapezoid Rule** to find the **approximate** result of with **3000 evenly spaced grid points** over the whole interval, then **print** the result.