Adamson University
College of Engineering
Computer Engineering Department

Linear Algebra

Laboratory Activity No. 8

System of Linear Equation

Submitted by:
Gatchalian, Paulo Valentin M.

Instructor:
Engr. Dylan Josh D. Lopez

May, 14, 2021

I. Objectives

This laboratory activity aims to teah the student to be familiarized with the system of linear equations, as well as to be able to solve it using various linear algebra methods and techniques through the use of Python Programming.

II. Methods

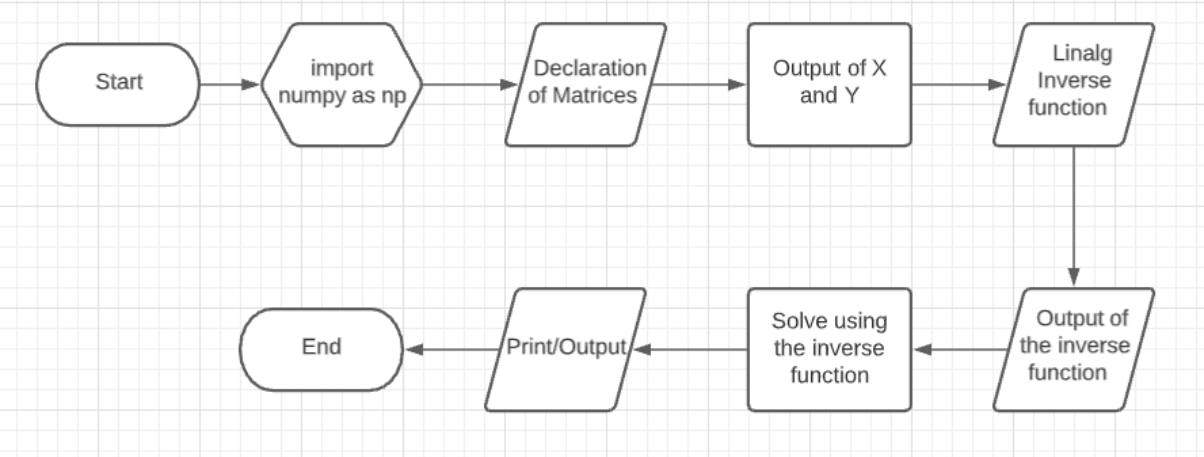

Figure 1: Flowchart used in the following Exercises

Given in Figure 1 expresses the methods that have been used in the 2 exercises showing the step by step process of the flow of the program and expresses the thought on how the system of linear equations presented in matrix form and the processes and its functionality using the np.linalg.solve() function and np.linalg.inv() function.

$$Matrices = \begin{cases} 3x + 4y + 5z = 2 \\ -1x - 2y - 3z = 4 \\ -5x + 2y + 3z = 6 \end{cases}$$

$$\begin{bmatrix} 3 & 4 & 5 \\ -1 & -2 & -3 \\ -5 & 2 & 3 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix}$$

Figure 2: Represents the System of Linear Equation

```
[12]  import numpy as np

[18]  A = np.array([
            [3,4,5],
            [-1,-2,-3],
            [-5,2,3]
      ])

      B = np.array([
            [2],
            [4],
            [6]
      ])
```

Figure 3: Representation of Importing Numpy and the Matrices

As shown in figure 3 it delares the usage of numpy and the 2 matries namely A and B from the given and will be further used in order to proceed into the next step.

```
[15]  X_inv = np.linalg.inv(X)
      Xinv_Y = X_inv @ Y

      print(Xinv_Y)
```

Figure 4: Representation of the Inverse Equation

III. Results

Given in the figures that will be represented in this chapter are the results and code snippets identifying the functionalities and processes and its given output.

```
[22]  print("A = \n", X)
      print("\nB = \n", Y)

      A =
       [[ 3  4  5]
       [-1 -2 -3]
       [-5  2  3]]

      B =
       [[2]
       [4]
       [6]]
```

Figure 5: Output of the two Matrices

Figure 5 shows the output of the declared matrices through the use of python programming.

```
[23]  X_inv = np.linalg.inv(X)
      Xinv_Y = X_inv @ Y

      print(Xinv_Y)

      [[ -1.66666667]
       [ 16.33333333]
       [-11.66666667]]

[24]  x_y = np.linalg.solve(X, Y)

      print(x_y)

      [[ -1.66666667]
       [ 16.33333333]
       [-11.66666667]]

[25]  ans1 = X @ Xinv_Y
      np.allclose(ans1, Y)

      True
```

Figure 6: Representation of Inverse Equation and the functions np.linalg.inv() and np.linalg.solve().

Figure 6 shows the Functionality of the equation through the use of np.linalg.inv() and np.linalg.solve(). And its results and outputs through the use of python programming,

IV. Conclusion

I therefore concluded that I was able to manifest and establish the use of the system of linear equations and being able to solve complex mathematical equations using Python Programming. In robotics, the use of the systems of linear equations can be used in different kinds of varities an example are the electrical, technological and scientifical aspects of robotics an example is the movement of the devices used in robotics since there are linear equations when it comes to the process of movement and as well as the electrical components that are being used in the system of robotics.

References

[1] D.J.D. Lopez. “Adamson University Computer Engineering Department Honor Code,” AdU-CpE Departmental Policies, 2020.

Github Repository: https://github.com/Gatchplease/Lin-AlgLab8