Lab Problem: 03.

Course code: CCE-312.
Course Title: Numerical Methods sSessional.

Name of the Lab Report: Solve Real world problem and Simul equation using Gauss-elimination method.

Remarks & Signature:

Submitted To

Professor Dr. Md. Samsuzzaman.
Chairman,
Department of Computer and Communication Engineering.
Faculty of Computer Science & Engineering.

Submitted By
HASAN AHAMMAD
ID No: 1902073
Reg No: 08779
Level- 3, Semester- 1
Session: 2019-2020
Faculty of Computer Science & Engineering.

Patuakhali Science & Technology University.
Dumki, Patuakhali-8602.

1. **Consider an online store selling T-shirts (T) and Hoodies (H). The prices are as follows:**
   - ✓ **A T-shirt costs $10.**
   - ✓ **A Hoodie costs $20.**

   A customer buys three items (either T-shirts or Hoodies) for a total cost of $50. How many T-shirts (x) and Hoodies(y) did the customer buy?

   Solve Using Gauss-elimination method after that implement it using Python.

✪ Solve using Gauss-elimination method.

The objective is to find the values of x and y that satisfy the following system of linear equations:
10x + 20y=50 (Total cost)
x + y=3 (Total Number of items)
Converting given equations into matrix form

$$\left[\begin{array}{cc|c} 10 & 20 & 50 \\ 1 & 1 & 3 \end{array}\right]$$

R_2<-R_2-0.1 x R_1

$$\left[\begin{array}{cc|c} 10 & 20 & 50 \\ 0 & -1 & -2 \end{array}\right]$$

10x+20y=50 ---(i)
-y=-2---(ii)
Now, use back substitution method from (ii)
y=2
Using y=2 in (i) we find:
x=1
so, the customer bought x=1 Shirts and y=2 Hoodie

✪ **Implement using Python:**

```
import numpy as np
coefficients = np.array([[10, 20], [1, 1]])

constants = np.array([50, 3])

augmented_matrix = np.column_stack((coefficients, constants))

n = len(constants)

for i in range(n):
    augmented_matrix[i, :] = augmented_matrix[i, :] /
augmented_matrix[i, i]

    for j in range(n):
        if i != j:
            augmented_matrix[j, :] -= augmented_matrix[j, i] *
augmented_matrix[i, :]

solutions = augmented_matrix[:, -1]

print("Number of T-shirts (x):", solutions[0])
print("Number of Hoodies (y):", solutions[1])
```

1. **Solve the following system by the Gauss-Elimination method and Implement it using Python.**

   $3x_1+.1x_2-.2x_3 = 7.85$
   $.1x_1+7x_2-.3x_3 = -19.3$
   $.3x_1-2x_2+10x_3 = 71.4$

Sol^n: We'll create the augmented matrix and perform row operations:

$$\left[\begin{array}{ccc|c} 3 & 0.1 & -0.2 & 7.85 \\ 0.1 & 7 & -0.3 & -19.3 \\ 0.3 & -2 & 10 & 71.4 \end{array}\right]$$

Step 1: Perform row operations to create zeros below the leading coefficient in the first column.

- ✓ R2 = R2 - (0.1/3) * R1
- ✓ R3 = R3 - (0.3/3) * R1

The augmented matrix becomes:

$$\left[\begin{array}{ccc|c} 3 & 0.1 & -0.2 & 7.85 \\ 0 & 6.99967 & -0.2933 & -19.56167 \\ 0 & -2.01 & 10.02 & 70.615 \end{array}\right]$$

Step 2: Create zeros below the leading coefficient in the second column.

- ✓ R3 = R3 + (2.01/6.97) * R2

The augmented matrix becomes:

$$\left[\begin{array}{ccc|c} 3 & 0.1 & -0.2 & 7.85 \\ 0 & 6.99967 & -0.2933 & -19.56167 \\ 0 & 0 & 9.935 & 64.99774 \end{array}\right]$$

Step 3: Solve for x_3 using the last row:

$9.935x_3 = 64.99774$
$x_3 \approx 64.99774/ 9.935$
$x_3 \approx 6.54$

Step 4: Substitute the value of x_3 into the second row to solve for x_2:

$6.99967x_2 - 0.2933x_3 = -19.56167$
$6.99967x_2 - -0.2933(6.54) = -19.56167$
$6.99967x_2 - 1.98 \approx -19.56167$
$6.99967x_2 \approx -19.56167 + 1.98$
$6.99967x_2 \approx -17.5817$
$x_2 \approx -17.5817/ 6.99967$
$x_2 \approx -2.51$

Step 5: Substitute the values of x_2 and x_3 into the first row to solve for x_1:

$3x_1 + 0.1x_2 - 0.2x_3 = 7.85$
$3x_1 + 0.1(-2.51) - 0.2(6.54) \approx 7.85$
$3x_1 - 0.251 – 1.308 \approx 7.85$
$3x_1 \approx 7.85 + 1.559$
$3x_1 \approx 9.409$
$x_1 \approx 9.409 / 3$
$x_1 \approx 3.136$

Please check the calculation twice

✪ **Implement using python:**

```
✪ import numpy as np
coefficients = np.array([[3, 0.1, -0.2],
                         [0.1, 7, -0.3],
                         [0.3, -2, 10]])
constants = np.array([7.85, -19.3, 71.4])
augmented_matrix = np.column_stack((coefficients, constants))

# Perform Gaussian elimination
n = len(constants)

for i in range(n):
    augmented_matrix[i, :] = augmented_matrix[i, :] / augmented_matrix[i, i]

    for j in range(n):
        if i != j:
            augmented_matrix[j, :] -= augmented_matrix[j, i] *
augmented_matrix[i, :]

solutions = augmented_matrix[:, -1]

print("Solution:")
for i, sol in enumerate(solutions):
    print(f"x{i + 1} =", sol)
```