Gaussian Elimination

AIM:

To write a program to find the solution of a matrix using Gaussian Elimination.

Equipments Required:

1. Hardware – PCs
2. Anaconda – Python 3.7 Installation / Moodle-Code Runner

Algorithm

step 1: Import the numpy module to use the built-in functions for calculation. Step 2: Import the sys module to use the built-in functions.

Step 3: Get input from the user for number of rows and add it by 1 for number of columns.

Step 4: Using np.zeros() set the matrix as null matrix.

Step 5: Using for loop get input from the user for each element in the matrix.

Step 6: Using for loop we can perform the elementary row operations and find the final matrix.

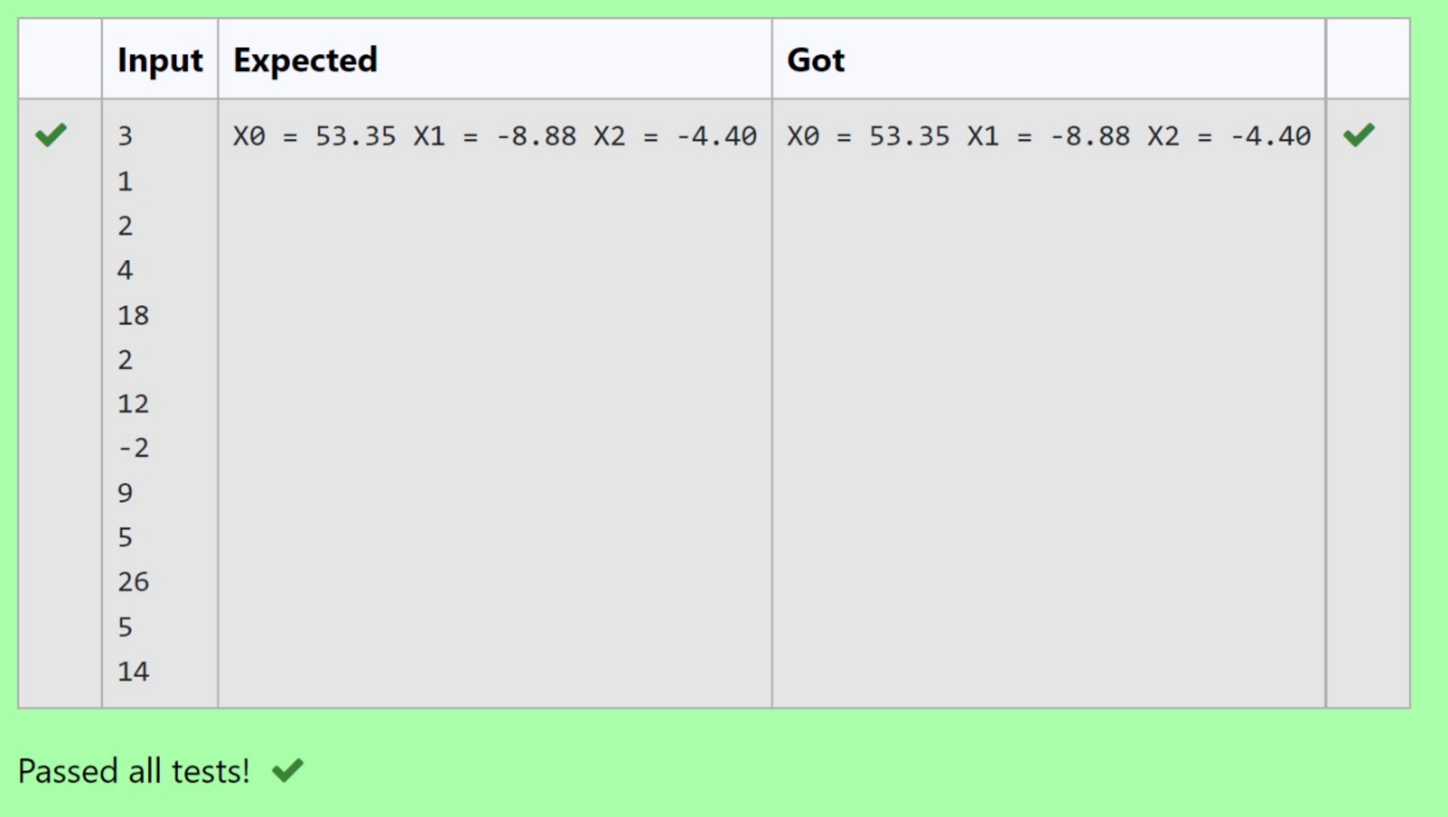
Step 7: Print the values by solving the matrix using for loop with 2 point precision.

Step 8: End the program.

Program:

```
/*
Program to find the solution of a matrix using Gaussian Elimination.
Developed by:
RegisterNumber:
*/import numpy as np
import sys
n = int(input())
a = np.zeros((n,n+1))
X = np.zeros(n)
for i in range(n):
    for j in range(n+1):
        a[i][j] = float(input())
for i in range(n):
    if a[i][i] == 0.0:
        sys.exit('Divide by zero detected')
    for j in range(i+1, n):
        ratio = a[j][i]/a[i][i]
        for k in range(n+1):
            a[j][k] = a[j][k] - ratio * a[i][k]
X[n-1] = a[n-1][n]/a[n-1][n-1]
for i in range(n-2,-1,-1):
    X[i] = a[i][n]
    for j in range(i+1,n):
        X[i] = X[i] - a[i][j]*X[j]
    X[i] = X[i]/a[i][i]
for i in range(n):
    print('X%d = %0.2f' %(i,X[i]), end = ' ')
```

Output:



	Input	Expected	Got	
✓	3 1 2 4 18 2 12 -2 9 5 26 5 14	X0 = 53.35 X1 = -8.88 X2 = -4.40	X0 = 53.35 X1 = -8.88 X2 = -4.40	✓

Passed all tests! ✓

Result:

Thus the program to find the solution of a matrix using Gaussian Elimination is written and verified using python programming.