```
"""
Registration: xxxx;
Description: Gauss-Seidel Iterative Method
Author: AKB
"""

import numpy as np
error = 1.0    # Initialize with a random error
tol   = 1E-3   # Tolerance

# Input 4 equations: 5x-y+z=7, 2x+4y-z=6, x+y+3z=5
n = int(input("Enter the number of equations: "))

print ('Enter the matrix coefficients : ')
a = np.array([[int(input('a'+str(i)+str(j)+' : ')) for j in range(n)] for i in range(n)])
a.astype(float)
b = np.array([int(input('b'+str(i)+' : ')) for i in range(n)])
b.astype(float)

# Initialize solution vector to zero
x = np.zeros(n);

# Do the iteration
count = 0
while error > tol:
   for i in range (n):
       sumSij = 0.0
       for j in range (n):
           if i!=j:
              sumSij += a[i,j]*x[j];
       x_new = (-sumSij + b[i])/a[i,i]
       error = abs(x[i] - x_new)
       count += 1
       if error > tol:
          x[i] = x_new

# Print the solution
print ('x = ', x, ', steps to converge = ', count)

"""
Results:
Enter the number of equations: 3
Enter the matrix coefficients :
a00 : 5
a01 : -1
a02 : 1
a10 : 2
a11 : 4
a12 : -1
a20 : 1
a21 : 1
a22 : 3
b0 : 7
b1 : 6
b2 : 5
x =  [1.42817778 0.99977222 0.85735   ] , steps to converge =  15
"""
```