```python
"""
Registration: xxxx;
Description: Gaussian Elimination
Author: AKB
"""

import time
import numpy as np

# Input 4 equations: 9a+b+c+d=75, a+8b+c+d=54, a+b+7c+d=43, a+b+c+6d=34
n = int(input("Enter the number of equations/rows: "))
print ('Enter the coefficients: ')
a = np.array([[float(input("a"+str(i)+str(j)+" : ")) for j in range(n)] for i in range(n)])
b = np.array([float(input("b"+str(i)+" : ")) for i in range(n)])

print ('Determinant calculated in direct method ', np.linalg.det(a))

# Print Solution that uses Gaussian Elimination in one line
start_time = time.time()
print ('Using direct Solver : ', np.linalg.solve(a,b))
exec_time = time.time() - start_time
print ('Execution time = ', exec_time, ' seconds')

start_time = time.time()
# Forward Elimination Stage
for k in range (n-1):        # pivot equation/row k=[0,1,2]
    for i in range (k+1,n):  # k=0;i=[1,2,3] k=1;i=[2,3] k=2;i=[3]
        if a[i,k] != 0.0:    # check whether pivot element=0 then stop
           factor      = a[i,k]/a[k,k]
           a[i,k+1:n] -= factor*a[k,k+1:n]
           b[i]       -= factor*b[k]

# Determinant after Elimination Stage = a11*a22*...*ann
det = 1
for k in range(n):
    det *= a[k,k]
print ('Determinant of a is ', det)

# Back Substitution
for k in range(n-1,-1,-1):
    b[k] = (b[k] - np.dot(a[k,k+1:n],b[k+1:n]))/a[k,k]

# Print solution
print ('The values of x are ', b)
exec_time = time.time() - start_time
print ('Execution time = ', exec_time, ' seconds')

"""
Results:
 Enter the number of equations: 4
 Enter the coefficients
     (9 1 1 1)       (75)
 a = (1 8 1 1),  b = (54)
     (1 1 7 1)       (43)
     (1 1 1 6)       (34)
Determinant calculated in direct method  2746.0
Using direct Solver :  [7. 5. 4. 3.]
Execution time =  0.02528214454650879  seconds
Determinant of a is  2746.000000000001
The values of x are  [7. 5. 4. 3.]
Execution time =  0.03007960319519043  seconds
"""
```