聯立方程組演算法觀察報告

* 碰到問題
  + N個未知數，需要使用到N個或以上的方程式。當亂數的矩陣出現了平行或是重疊的狀況時，由於方程式不足，容易導自程式錯誤，需要以例外處裡。當係數相同，但B不相同例外處理為無解(No Solution)，而當係數與B皆相同，則是以無限多組解(Infinite Solutions)
* 作業心得
  + 透過小數的N寫出來的code，也能相同使用在大數N的矩陣上，可看出Gaussian Elimination 演算法，相對其他解方程式的方法，是比較適用於電腦code，因為它具有一定的規律性。解的部分也只有單純的三種方式，不會出現其他例外狀況。
* 程式

import numpy as np

import sys

for n in range(2,100):

# uniform -> random

# randint -> int

print("=====================================")

print("n=",n)

A = np.random.randint(1,100,[n,n])

b = np.random.randint(1,100,n)

ans = np.zeros(n)

print("A=",A)

print("b=",b)

flag = 0

for j in range(0,n-1):

for i in range(j+1,n):

if(A[j][j] == 0 and b[i] != b[j]):

flag = 1

break

if(A[j][j] == 0 and b[i] == b[j]):

flag = 2

break

mul = A[i][j] / A[j][j]

A[i][j] -= A[j][j] \* mul

A[i][i] -= A[j][i] \* mul

b[i] -= b[j] \* mul

if(flag == 1):

print("No Solution")

continue

if(flag == 2):

print("Infinite Solutions")

continue

for i in range(n-1,-1,-1):

temp = 0

for j in range(n-1,i,-1):

temp += A[i][j] \* ans[j]

ans[i] = (b[i] - temp)/A[i][i]

print("ans=",ans)