Лабораторна робота №3

*Чисельні методи розв’язання нелінійних рівнянь. Уточнення кореня нелінійного рівняння. Метод Ньютона. Комбінований метод*

Виконала Гальчинська Софія, студентка ФІТ 2-8. Варіант 4.

[*https://github.com/GalchynskaSofiia/Numerical\_Methods\_Of\_Programming*](https://github.com/GalchynskaSofiia/Numerical_Methods_Of_Programming)

* 1. **Знайти матрицю C = AB – BA (варіант 2)**

import numpy as np

a\_mx = np.matrix([[2, 3, 1], [-1, 1, 0], [1, 2, -1]])

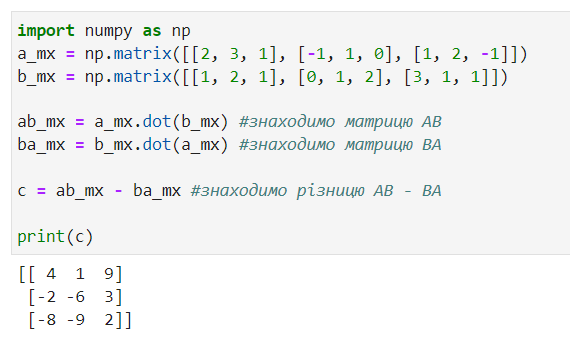
b\_mx = np.matrix([[1, 2, 1], [0, 1, 2], [3, 1, 1]])

ab\_mx = a\_mx.dot(b\_mx) #знаходимо матрицю AB

ba\_mx = b\_mx.dot(a\_mx) #знаходимо матрицю BA

c = ab\_mx - ba\_mx #знаходимо різницю AB - BA

print(c)



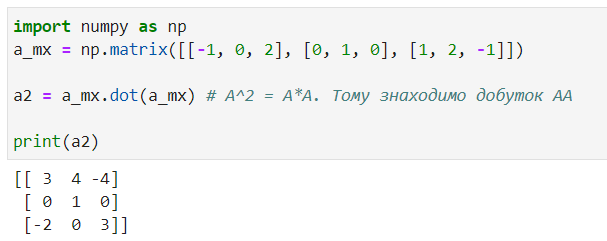
* 1. **Піднести матрицю до степені (варіант 2)**

import numpy as np

a\_mx = np.matrix([[-1, 0, 2], [0, 1, 0], [1, 2, -1]])

a2 = a\_mx.dot(a\_mx) # A^2 = A\*A. Тому знаходимо добуток AA

print(a2)



* 1. **Знайти добуток матриць (варіант 2)**

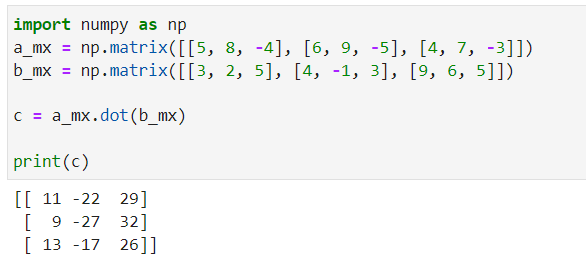
import numpy as np

a\_mx = np.matrix([[5, 8, -4], [6, 9, -5], [4, 7, -3]])

b\_mx = np.matrix([[3, 2, 5], [4, -1, 3], [9, 6, 5]])

c = a\_mx.dot(b\_mx)

print(c)



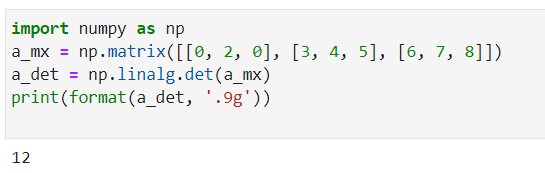
* 1. **Обчислити визначник (варіант 2)**

import numpy as np

a\_mx = np.matrix([[0, 2, 0], [3, 4, 5], [6, 7, 8]])

a\_det = np.linalg.det(a\_mx)

print(format(a\_det, '.9g'))



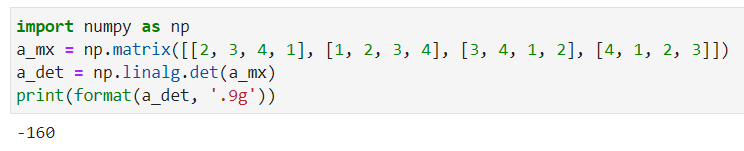
* 1. **Обчислити визначник (варіант 2)**

import numpy as np

a\_mx = np.matrix([[2, 3, 4, 1], [1, 2, 3, 4], [3, 4, 1, 2], [4, 1, 2, 3]])

a\_det = np.linalg.det(a\_mx)

print(format(a\_det, '.9g'))



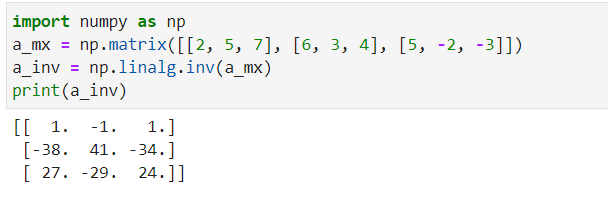
* 1. **Знайти обернену матрицю (варіант 2)**

import numpy as np

a\_mx = np.matrix([[2, 5, 7], [6, 3, 4], [5, -2, -3]])

a\_inv = np.linalg.inv(a\_mx)

print(a\_inv)



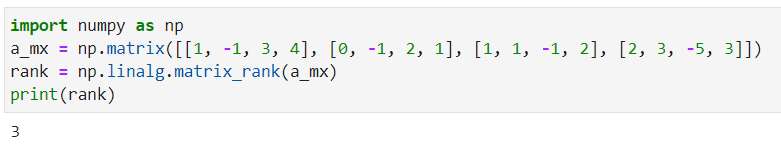
* 1. **Визначити ранг матриці (варіант 2)**

import numpy as np

a\_mx = np.matrix([[1, -1, 3, 4], [0, -1, 2, 1], [1, 1, -1, 2], [2, 3, -5, 3]])

rank = np.linalg.matrix\_rank(a\_mx)

print(rank)



* 1. **Розв’язати систему лінійних рівнянь методом Крамера і зробити перевірку за допомогою solve() пакету linalg (варіант 4)**

import numpy as np

a = np.matrix([[7, 3, -6], [7, 9, -9], [2, -4, 9]])

b = np.matrix([[-1], [5], [28]])

def kram(a,b):

det\_a = np.linalg.det(a)

if det\_a != 0:

a1 = np.matrix(a)

a2 = np.matrix(a)

a3 = np.matrix(a)

a1[:, 0] = b

a2[:, 1] = b

a3[:, 2] = b

x = np.linalg.det(a1)/det\_a

y =np.linalg.det(a2)/det\_a

z= np.linalg.det(a3)/det\_a

print('Solving a system of linear algebraic equations by Cramer\*s method')

print('x=',x.round(3),'y=',y.round(3),'z=',z.round(3))

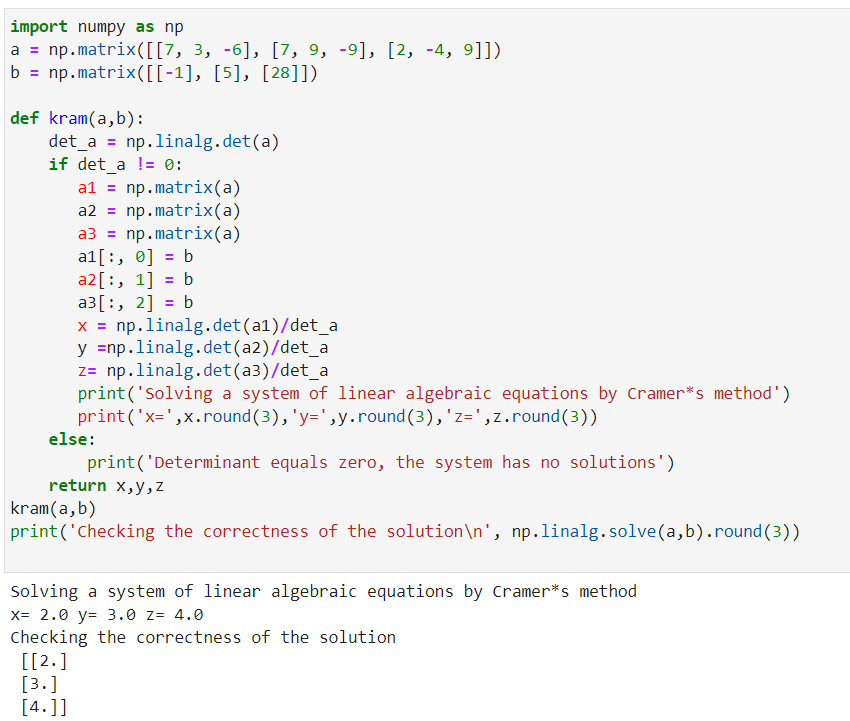
else:

print('Determinant equals zero, the system has no solutions')

return x,y,z

kram(a,b)

print('Checking the correctness of the solution\n', np.linalg.solve(a,b).round(3))



**Розв’язати ту ж систему лінійних рівнянь матричним методом**

import numpy as np

a = np.matrix([[7, 3, -6], [7, 9, -9], [2, -4, 9]])

b = np.matrix([[-1], [5], [28]])

print('A=', a)

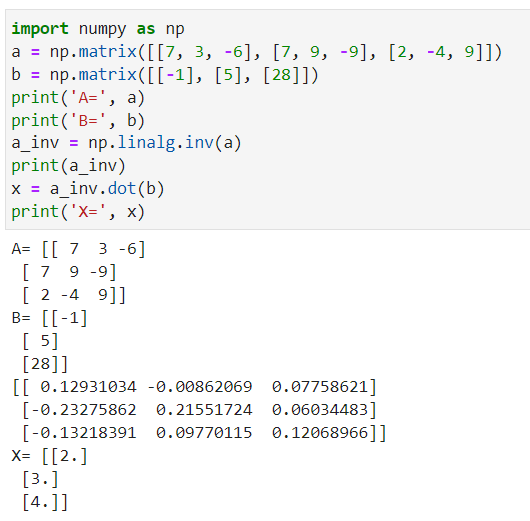
print('B=', b)

a\_inv = np.linalg.inv(a)

print(a\_inv)

x = a\_inv.dot(b)

print('X=', x)



Відповіді методом Крамера і матричним методом збігаються:

x = 2, y = 3, z = 4.

1. **Створіть прямокутну матрицю A, яка має N рядків і M стовпців з випадковими елементами. Знайдіть суму елементів всієї матриці. Визначте, яку долю в цій сумі складає сума елементів кожного стовпця (варіант 4).**

import numpy as np

n = 2 #кількість рядків

m = 4 #кількість стопців

a = np.random.randint( -11, 11, (n, m))

print(a)

sum\_columns = sum(a) #матриця із значеннями сум стовпців

sum\_mx = sum(sum\_columns) #сума усіх елементів матриці

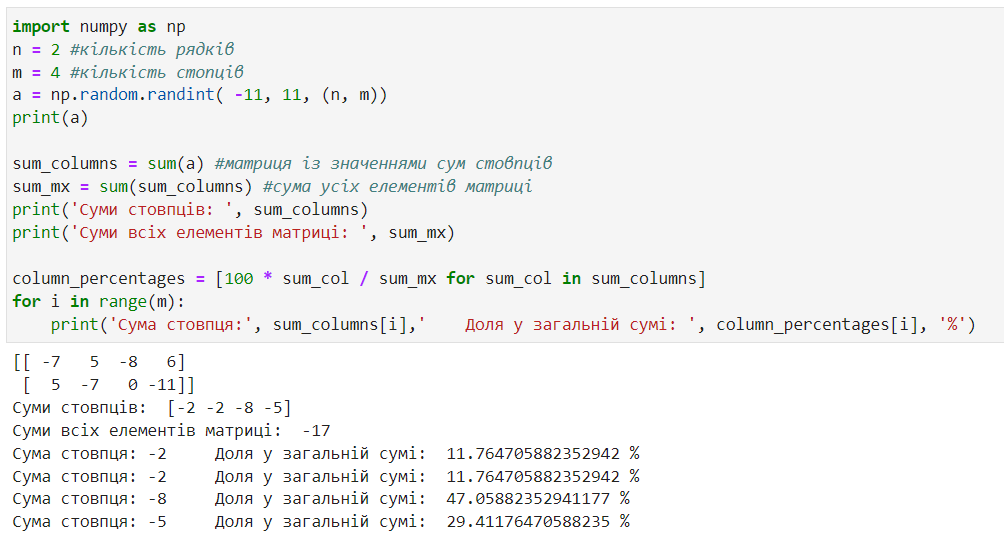
print('Суми стовпців: ', sum\_columns)

print('Суми всіх елементів матриці: ', sum\_mx)

column\_percentages = [100 \* sum\_col / sum\_mx for sum\_col in sum\_columns]

for i in range(m):

print('Сума стовпця:', sum\_columns[i],' Доля у загальній сумі: ', column\_percentages[i], '%')



**Висновки:** Отже, під час виконання практичної роботи ми навчились виконувати різні дії над матрицями за допомогою програмування, зокрема використання Python для множення, віднімання, знаходження оберненої матриці, рангу і визначника матриці. Також у результаті виконання практичної роботи була розв’язана система лінійних рівнянь за допомогою методу Крамера та матричного методу. Результати методів співпали.