Національний технічний університет України «КПІ ім. Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

Лабораторна робота № 4

з дисципліни «Спеціальні розділи математики-2.  
Чисельні методи»

Виконав:

студент гр. ІС-34

Колосов Ігор

Викладач:

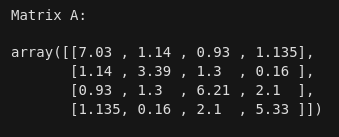
доц. Рибачук Л.В.

Київ – 2024

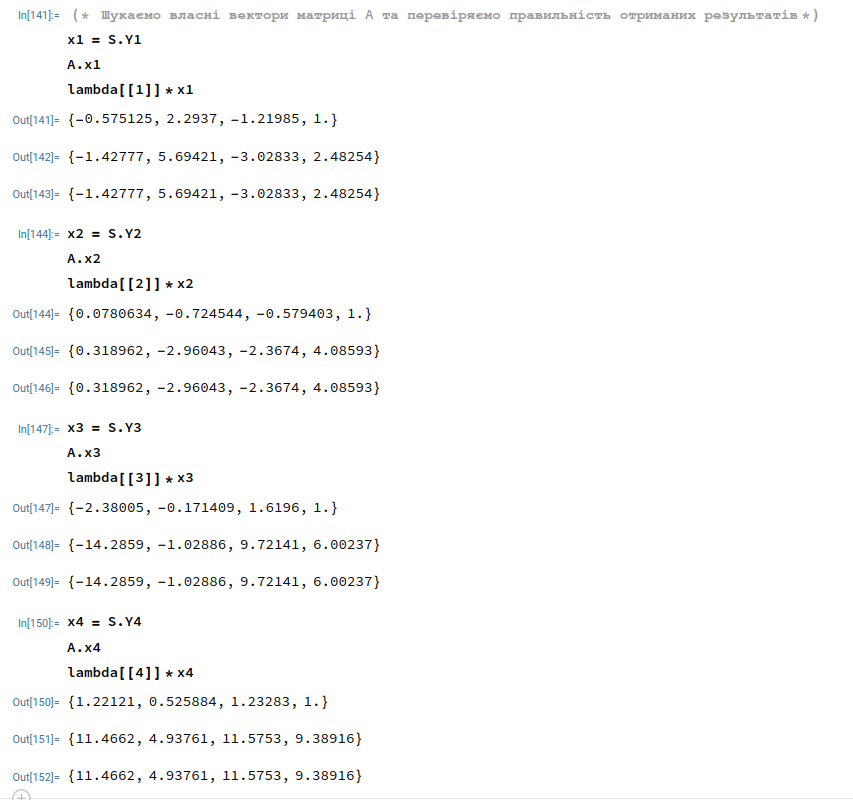
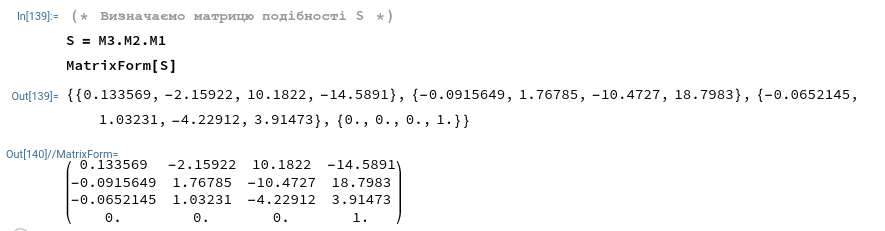
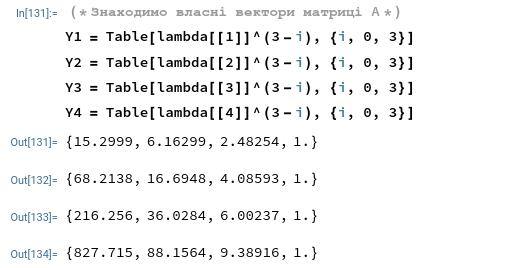
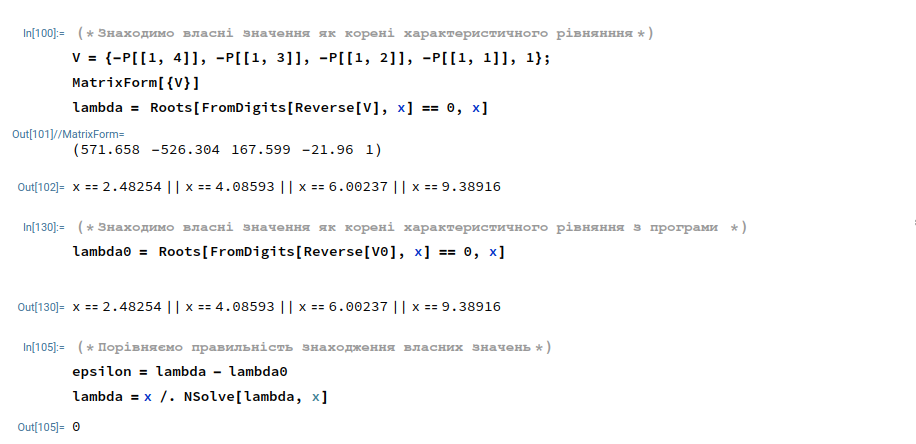
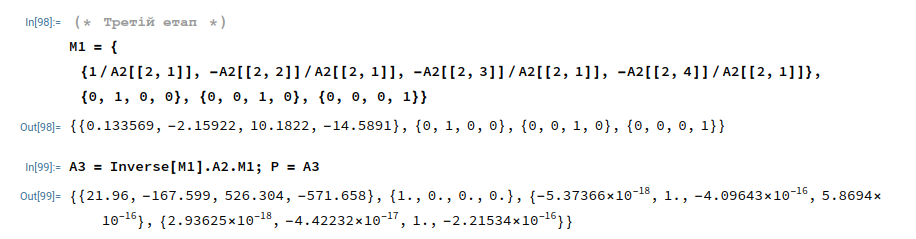
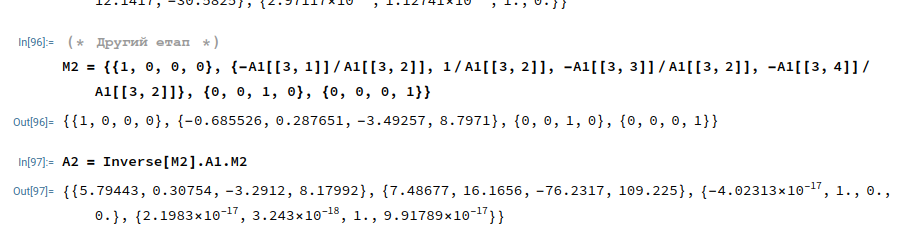
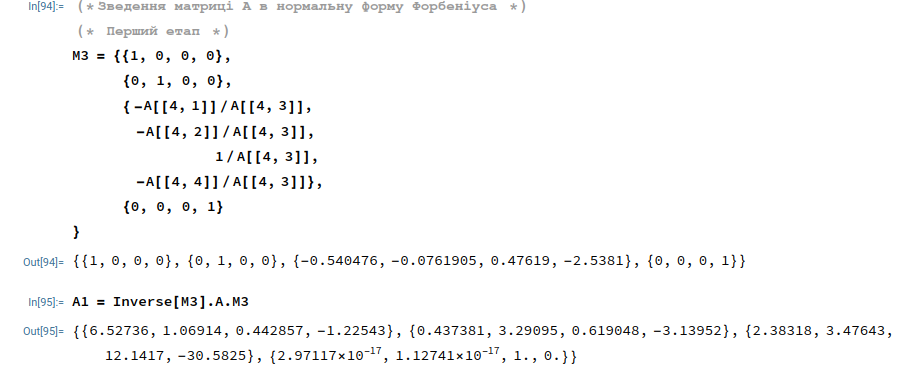
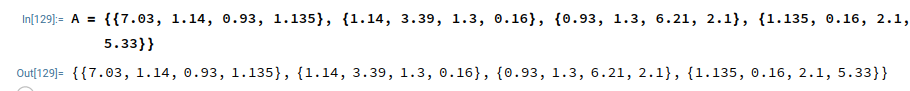
Тема: Обчислення власних значень та власних векторів матриць.

Постановка задачі:

Створити програму, для зведення матриці А до нормальної форми Фробеніуса Р. Отримане характеристичне рівняння розв’язати довільним способом у Mathcad і отримати всі власні числа λі, і = 1,…,m з точністю 5 знаків після коми. Для кожного власного числа знайти по одному власному вектору через власні вектори матриці Р. Перевірити точність знайдених результатів, підставляючи у рівняння (1) знайдені власні числа та власні вектори. Знайти власні числа матриці А виключно за допомогою Mathcad і порівняти з отриманими раніше результатами.



Розв’язок:



Лістинг програми:

**import** numpy **as** np

**from** scipy **import** linalg

*# Вхідна матриця*

A **=** np.array([

[7.03, 1.14, 0.93, 1.135],

[1.14, 3.39, 1.3, 0.16],

[0.93, 1.3, 6.21, 2.1],

[1.135, 0.16, 2.1, 5.33]

])

*# Перший етап*

M3 **=** np.eye(4)

M3[2, :] **=** [**-**A[3,0]**/**A[3,2], **-**A[3,1]**/**A[3,2], 1**/**A[3,2], **-**A[3,3]**/**A[3,2]]

print("M3")

print(M3)

A1 **=** np.linalg.inv(M3) **@** A **@** M3

print("A1")

print(A1)

*# Другий етап*

M2 **=** np.eye(4)

M2[1, :] **=** [**-**A1[2,0]**/**A1[2,1], 1**/**A1[2,1], **-**A1[2,2]**/**A1[2,1], **-**A1[2,3]**/**A1[2,1]]

print("M2")

print(M2)

A2 **=** np.linalg.inv(M2) **@** A1 **@** M2

print("A2")

print(A2)

*# Третій етап*

M1 **=** np.eye(4)

M1[0, :] **=** [1**/**A2[1,0], **-**A2[1,1]**/**A2[1,0], **-**A2[1,2]**/**A2[1,0], **-**A2[1,3]**/**A2[1,0]]

print("M1")

print(M1)

P **=** np.linalg.inv(M1) **@** A2 **@** M1

V **=** np.array([**-**P[0,3], **-**P[0,2], **-**P[0,1], **-**P[0,0], 1])

lambda\_values **=** np.roots(V[::**-**1])

print("Власні значення:")

**for** i, l **in** enumerate(lambda\_values, 1):

print(**f**"λ\_{i} = {l**:.5f**}")

Y **=** np.array([

[l**\*\***3, l**\*\***2, l, 1] **for** l **in** lambda\_values

])

S **=** M3 **@** M2 **@** M1

print("Матрция подібності")

print(S)

eigenvectors **=** S **@** Y.T

print("\nВласні вектори:")

**for** i, v **in** enumerate(eigenvectors.T, 1):

print(**f**"v\_{i} = {v}")

print("\nПеревірка точності:")

**for** i, (lambda\_i, v) **in** enumerate(zip(lambda\_values, eigenvectors.T), 1):

error **=** np.linalg.norm(A **@** v **-** lambda\_i **\*** v)

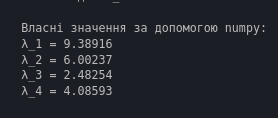
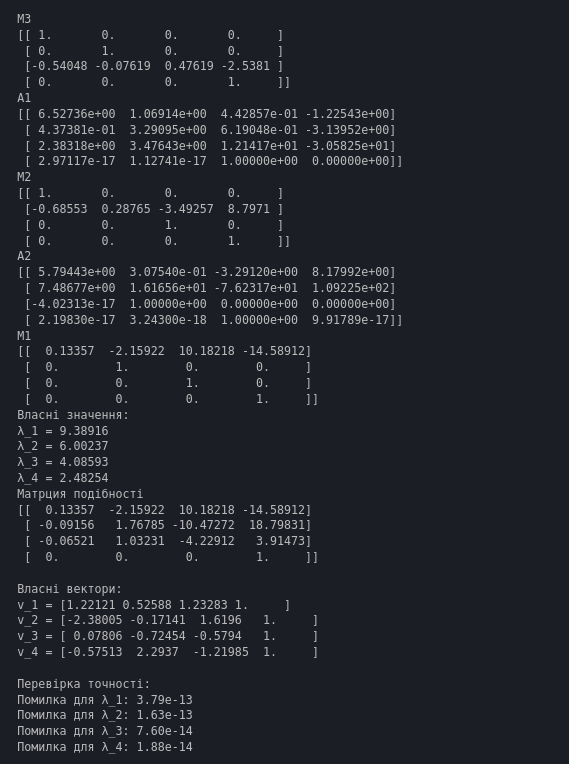
print(**f**"Помилка для λ\_{i}: {error**:.2e**}")

print("\nВласні значення за допомогою numpy:")

np\_eigenvalues, np\_eigenvectors **=** np.linalg.eig(A)

**for** i, ev **in** enumerate(np\_eigenvalues, 1):

print(**f**"λ\_{i} = {ev**:.5f**}")



Скріншоти виконання програми:

Висновок

У ході виконання лабораторної роботи я дізнався про методи обчислення власних значень та власних векторів матриць, а саме метод Данилевського. Дізнався про інструменти для знаходження власних значень матриці в програмі Wolfram Mathematica. Я навчився програмно реалізовувати метод Данилевського і зводити матриці до нормальної форми Форбеніуса та використовувати цей метод у середовищі Wolfram Mathematica для знаходження власих значень та векторів представлених матриць.