**Комп’ютерний практикум №4**

**Обчислення власних значень**

**Виконав:**

Студент 3 курсу ФТІ

групи ФІ-92

Поночевний Назар Юрійович

Варіант 12

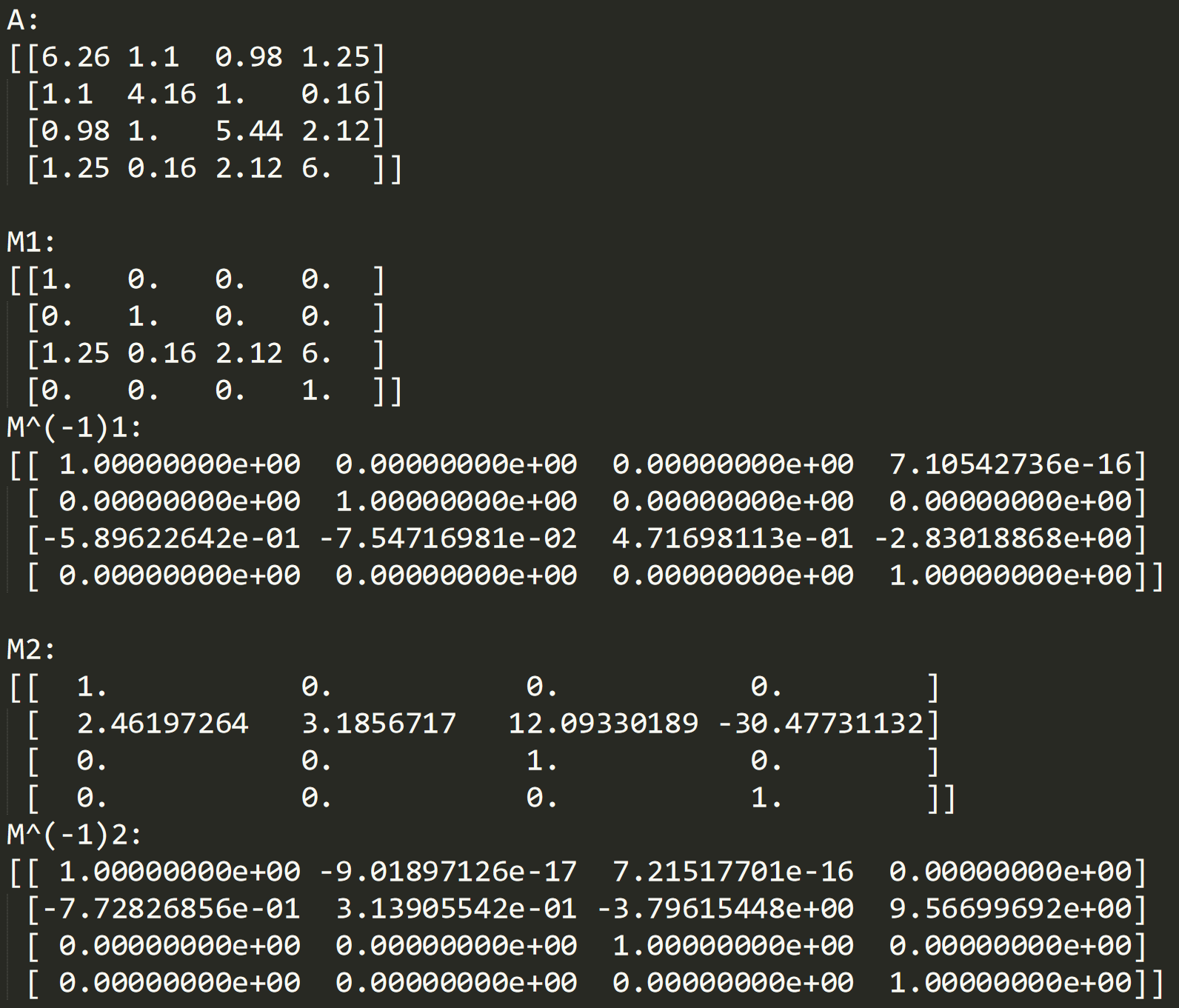
**Завдання:**

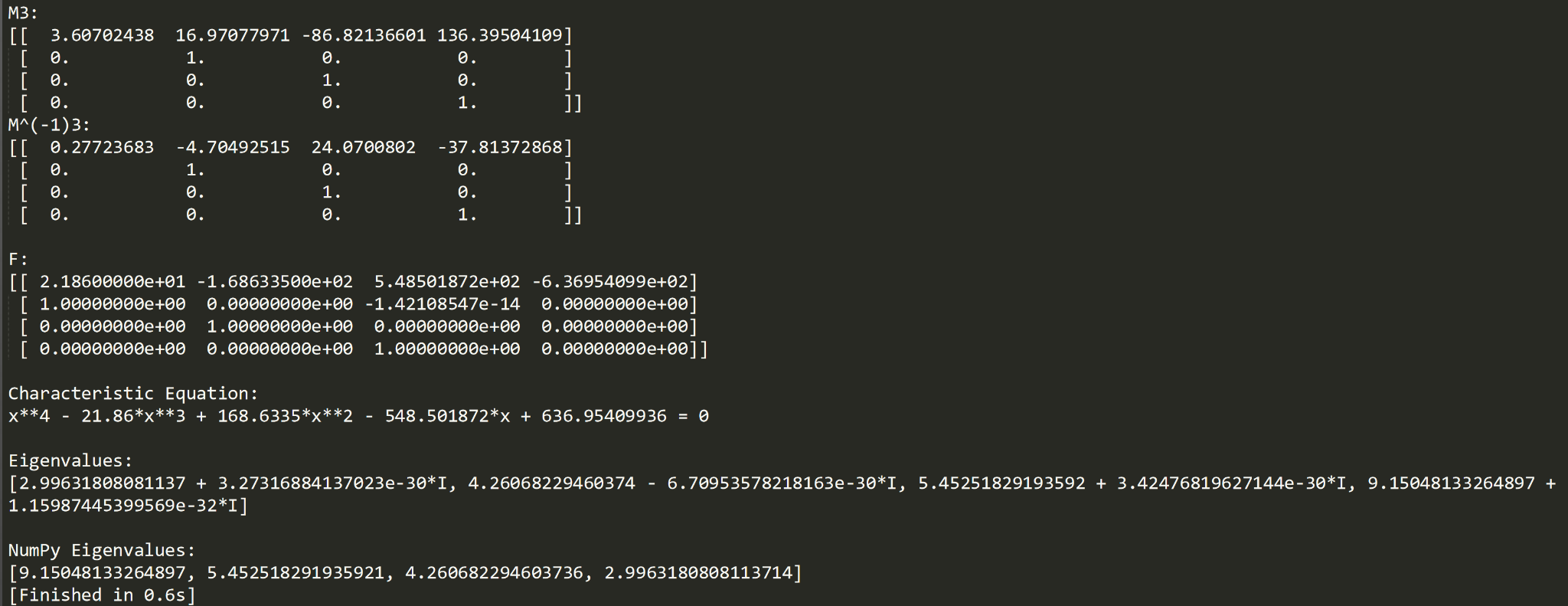
Для методу Данилевського: привести матрицю до вигляду Фробеніуса, розв’язати отриману систему за допомогою методу із практикуму 2 або 3, отримати коефіцієнти характеристичного рівняння. Розв’язати характеристичне рівняння за допомогою одного з методів із практикуму 1 і отримати власні числа. Для всіх варіантів: виконати перевірку отриманих результатів за допомогою математичного пакета (наприклад, можна використати функцію Matlab eig()).

1. Реалізуємо програму

| """ Finding all eigenvalues  """  # ------------ Input ------------   import numpy as np import sympy as sp   A = np.array([[6.26, 1.10, 0.98, 1.25],  [1.10, 4.16, 1., 0.16],  [0.98, 1., 5.44, 2.12],  [1.25, 0.16, 2.12, 6.]])   # ------------ Code ------------   def danilevsky\_method(f):  for i in range(len(f) - 1):  m = np.identity(len(f))  m[len(f) - 2 - i][:] = f[len(f) - 1 - i][:]  print(f"\nM{i + 1}:\n{m}")  print(f"M^(-1){i + 1}:\n{np.linalg.inv(m)}")  f = np.dot(m, f)  f = np.dot(f, np.linalg.inv(m))  return f   def equation\_solve(p):  x = sp.Symbol('x')  e = x\*\*4 - p[0] \* x\*\*3 - p[1] \* x\*\*2 - p[2] \* x - p[3]  return e, sp.solve(e, x)   def main():  print(f"A:\n{A}")  f = danilevsky\_method(A)  e, lambdas = equation\_solve(f[0][:])  print(f"\nF:\n{f}")  print(f"\nCharacteristic Equation:\n{e} = 0")  print(f"\nEigenvalues:\n{lambdas}")  print(f"\nNumPy Eigenvalues:\n{list(np.linalg.eig(A)[0])}")   if \_\_name\_\_ == "\_\_main\_\_":  main() |
| --- |

1. Результат





1. Контрольні запитання

**Коли метод Данилевського неможливо застосувати?**

Коли матриця не симетрична і додатно визначена.

**Які елементи матриці будуть зменшуватися при обертаннях за методом Якобі, а які будуть збільшуватись?**

Недіагональні елементи будуть зменшуватися, а діагональні - збільшуватися.