Львівський національний університет імені Івана Франка

**Звіт про виконання лабораторної роботи №2**

на тему:

**«**Векторизація даних**»**

Виконав:

Студент групи Феп-31

Скребуха Володимир

Перевірив:

Ас. Рибак А. В.

Львів – 2022

Мета роботи: Дослідити базові методи бібліотеки NumPy та провести аналіз особливостей бібліотеки NumPy на основі реалізованих методів в порівнянні Python vs Python + NumPy.

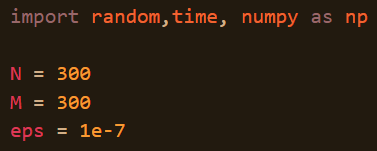
**Хід роботи**

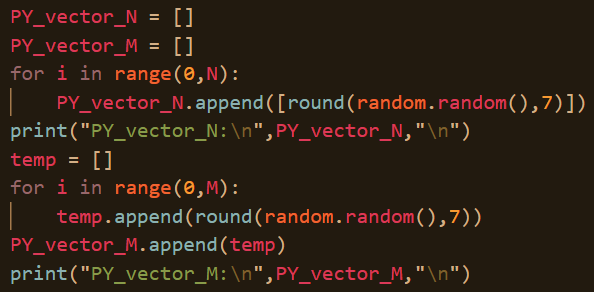
1. Пишу програму, до якої підключаю модулі random, numpy, time.

Описую масиви даних PY\_N та PY\_M, в яких генерую випадкові значення на проміжку [0,1).

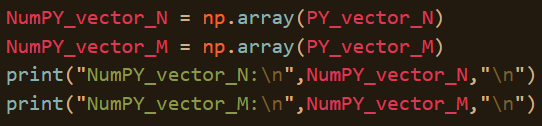
Кількість елементів масивів позначаю змінними N та M, де перший відповідає за кількість елементів в одному рядку першого масиву, другий за кількість елементів в колонці другого масиву.

Також пишу змінну точності.

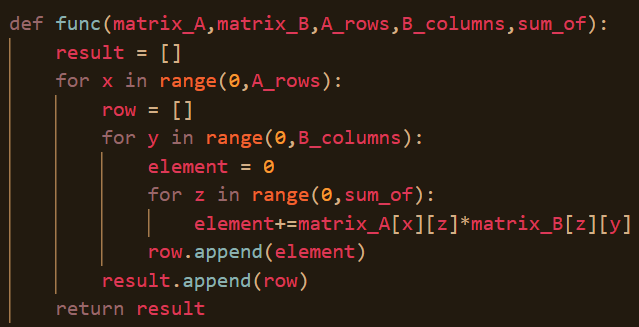




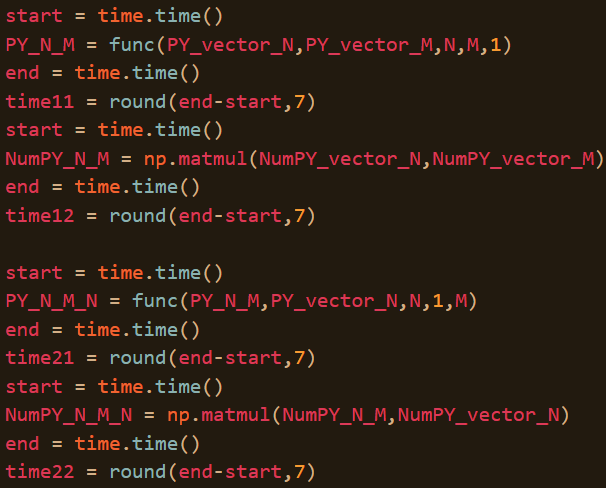
1. Створюю дублікати масивів типу numpy командою array(). Це допоможе перевірити чи правильно працюють зовнішні функції, які прийматимуть звичайні масиви.

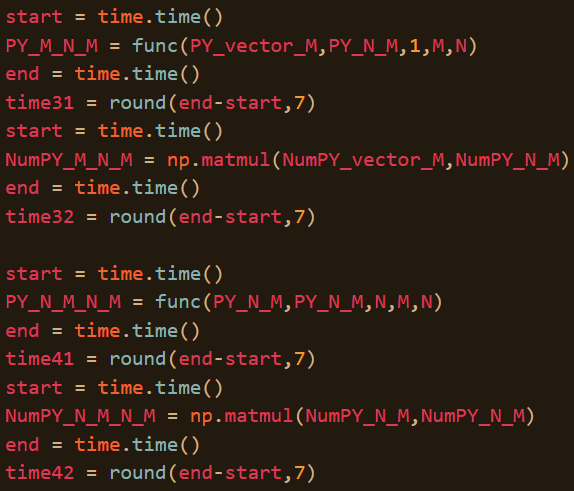


1. Описую зовнішню функцію func(), якою буду виконувати усі 4 види множення. Параметрами функції є масиви А та В, кількість рядків масиву А, кількість стовпців масиву В, та кількість елементів, які потрібно буде додавати.

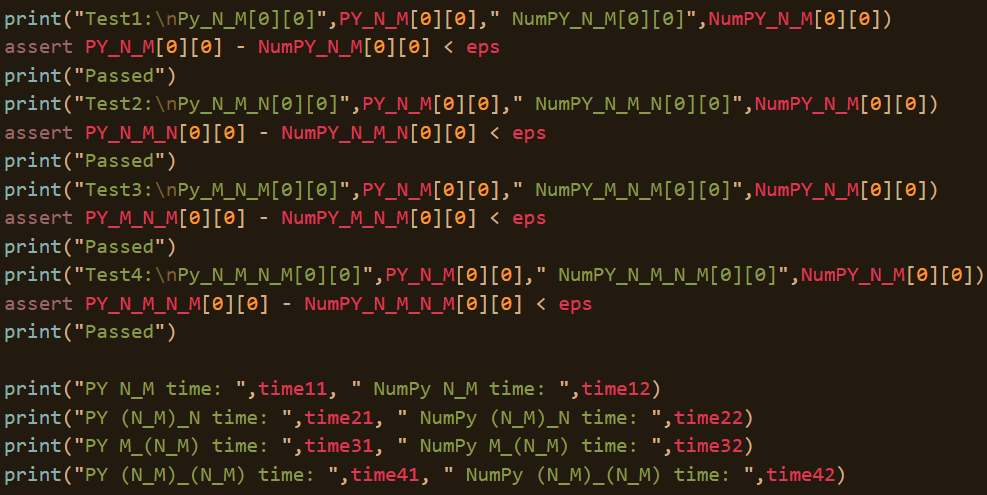


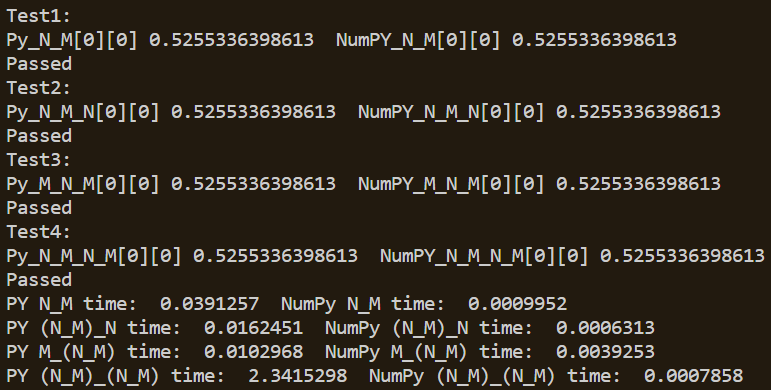
1. Почергово виконую множення вектора на вектор, матрицю на вектор, вектора на матрицю, та матрицю та матрицю. Використано було для цього лише два початкових вектори та результати множень. Крім цього командою time.time() записую час роботи кожного прикладу. Почергово зі своєю функцію виконує та записую час виконання множення функцію numpy.matmul().





1. Повертаючись до тої точності, описую тестування своїх функцій командою assert, яка перевірятиме чи перший елемент масиву PY є рівний з заданою точністю до першого елементу NumPY, який, очевидно, виконує правильне обчислення. В кінці програми виводжу час роботи кожної пари функцій над кожним прикладом.





Висновок: на цій лабораторній роботі було розглянуто алгоритм множення матриць та векторів, після чого інтерпретовано у мові програмування Python для порівняння часу виконання написаних вручну функцій у парі зі звичайними масивами, та модулем функцій numpy у парі зі спеціальним типом масивів. В результаті мені вдалося добити наближеного до правди результату обчислення, проте зрівнятися з часом роботи numpy в одну тисячну секунду не є можливим. Засвоєно, що при роботі з матрицями та векторами numpy є оптимальний варіант.