

Гончаренко Я.В.

кандидат фіз.-мат. наук, доцент,

Сергійко Д. М.

студентка,

Український державний університет імені Михайла Драгоманова

ОЗНАЙОМЛЕННЯ З МОЖЛИВОСТЯМИ МАТЕМАТИЧНИХ БІБЛІОТЕК PYTHON ЯК ОДНЕ З ЗАВДАНЬ ОБЧИСЛЮВАЛЬНОЇ ПРАКТИКИ

Навчальними планами підготовки студентів освітнього рівня «бакалавр» спеціальностей 111 Математика та 014 Середня освіта (математика) в НПУ імені М.П. Драгоманова передбачена обчислювальна практика. Організація і проведення обчислювальної практики повинні сприяти:

- поглибленню і закріпленню здобутих за час навчання теоретичних і практичних знань з математичного аналізу, лінійної алгебри, аналітичної геометрії;
- набуттю студентами вмінь та навичок використання систем комп'ютерної математики, використання та розробки програмного забезпечення, орієнтованого на розв'язання задач обчислювальної математики;
- поглибленню знань та удосконаленню навичок використання інформаційних технологій у навчальному процесі;
- розвитку логічного мислення студентів, підвищенню рівня їх математичної і обчислювальної культури.

При вирішенні завдання використання та розробки програмного забезпечення для розв'язання математичних задач, на нашу думку, одним найкращих середовищ програмування є Python.

Python є універсальною мовою програмування, за допомогою якої можна вирішувати велику кількість задач: від веб-розробки до наукових обчислень. Особливістю Python є велика кількість бібліотек, які суттєво спрощують розв'язання окремих класів завдань. Математичні бібліотеки Python за простотою використання порівнянні з кращими зразками систем комп'ютерної математики (Maple, Mathematica тощо). В той же час Python є мовою програмування, що надає широкі можливості для створення програмних кодів.

Дана робота присвячена огляду деяких бібліотек Python, які використовуються при розв'язання математичних задач. Вміння використовувати саме ці бібліотеки, на нашу думку, доцільно формувати під час обчислювальної практики студентів математичних спеціальностей.

Бібліотека **Numpy** Python. Містить інструменти для чисельних обчислень, обчислення значень математичних функцій, дії з матрицями, векторами та масивами даних, перетворення Фур'є, генератори випадкових чисел.

Matplotlib – графічна бібліотека Python, за допомогою якої можна створювати статичні, анімовані або інтерактивні візуалізації даних.

Pandas – бібліотека для аналізу великих наборів даних. Численні вбудовані методи дозволяють генерувати, групувати, індексувати, отримувати, розділяти,

реструктурувати дані, фільтрувати набори даних, створювати та аналізувати одновимірні та багатовимірні таблиці.

SciPy – це розширення NumPy, яке додає інструменти для вирішення математичних задач, зокрема:

- `scipy.special` – спеціальні функції для вирішення задач математичної фізики;
- `scipy.integrate` – функції для чисельного інтегрування звичайних диференціальних рівнянь
- `scipy.optimize` – алгоритми оптимізації (задачі математичного програмування);
- `scipy.interpolate` – методи інтерполяції;
- `scipy.fft` – перетворення Фур'є;
- `scipy.linalg` – операції лінійної алгебри. Аналогічні модулі є в NumPy, але в SciPy їх кількість та функціонал розширені;
- `scipy.sparse.csgraph` – методи для роботи з графами;
- `scipy.stats` – методи для статистичних розрахунків.

SymPy – бібліотека для виконання символьних обчислень. SymPy включає різноманітні функції, зокрема: символьну арифметику та алгебру, комбінаторику, дискретну математику, геометричні модулі, елементи статистики та криптографії. Результати обчислень в SymPy можна форматовувати як код LaTeX.

Представимо приклад програми, розробленої з використанням наступних бібліотек та функцій Python:

- Бібліотека NumPy:
 - `np.array` – задання масиву (матриці);
 - `np.cross` – обчислення векторного добутку двох векторів у тривимірному просторі;
 - `np.dot` – обчислення скалярного добутку двох масивів (у нашому випадку – векторів);
 - `np.linalg.norm` – обчислення норми масиву (тобто, довжини вектора).
- Бібліотека Matplotlib:
 - `plt.figure` – відображення поля найвищого рівня, таке собі тла, на якому малюється графік;
 - `fig.add_subplot` – додавання області графіка до новоствореного тла;
 - `plt.show` – функція ввімкнення відображення сітки.
- Бібліотека Tkinter: було використано ряд функцій для задання та форматування вікон, робочих областей, кнопок тощо.

Створена програма дозволяє вирішувати наступні завдання:

- ввести координати чотирьох точок в тривимірній декартовій системі координат;
- визначити чи лежать задані точки в одній площині;
- обрати один з двох об'єктів: паралелепіпед або піраміду, що визначаються даними точками;
- обчислити об'єм обраного тіла;
- знайти координати векторів напрямку бічних ребер та координати центру мас;
- обчислити довжини бічних ребер, площі бічних граней, довжини висот, кути між бічними ребрами;

- побудувати тіло в тривимірній декартовій системі координат.

На рис. 1 та 2 представлені результати роботи програми.

Зазначимо, що дана програма була підготована під час ознайомлення з можливостями Python на гуртку з комп'ютерної математики.

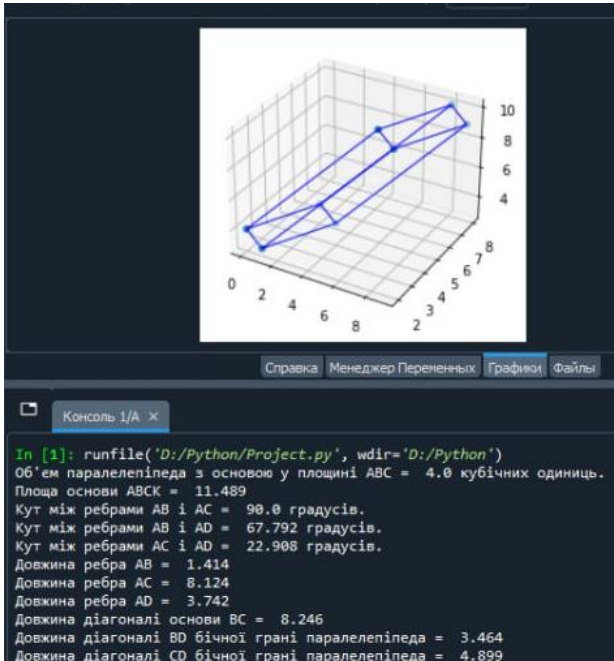


Рис.1.

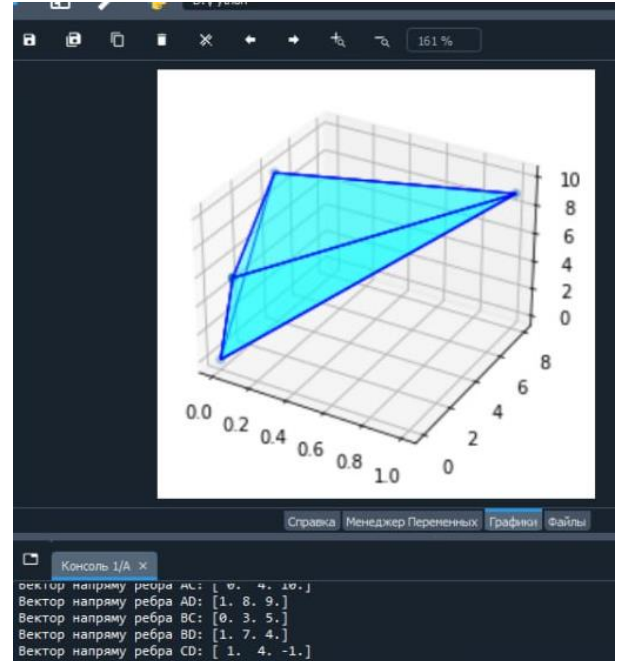


Рис.2.

Виходячи з досвіду використання мови програмування Python та навчання студентів елементів програмування в Python для розв'язання математичних задач, можна зробити наступні висновки: використання Python під час обчислювальної практики студентами математичних спеціальностей є доцільним і, поряд з ознайомленням з можливостями деяких систем комп'ютерної математики та ППЗ спеціального призначення, має ряд переваг. До таких переваг можна віднести:

- сучасність, поширеність та динамічність Python, великий набір постійно оновлюваних бібліотек;
- широкі функціональні можливості, які дозволяють відносно просто автоматизувати розв'язання найрізноманітніших математичних задач;
- відносну простоту в засвоєнні принципів написання коду;
- безкоштовність;
- можливість формувати навички програмування мовою високого рівня, використовуючи наявні, а також розширюючи та поглиблюючи математичні знання студентів.

Список використаних джерел

1. Python Software Foundation. The Python Tutorial [Електронний ресурс]. – Режим доступу до ресурсу: <https://docs.python.org/3/tutorial/index.html>
2. Python Software Foundation. Python 3.7.12 documentation [Електронний ресурс]. – Режим доступу до ресурсу: <https://docs.python.org/3.7/>
3. Lutz M. Learning Python, 5th Edition. – O'Reilly Media Inc., 2013. – 1648 p.