МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Львівський національний університет імені Івана Франка

Факультет прикладної математики та інформатики

Кафедра програмування

Затверджено

На засіданні кафедри програмування факультету прикладної математики та інформатики Львівського національного університету імені Івана Франка

(протокол № 1 від 29 серпня 2025 р.)

Зав. кафедри: к. ф.-м. н., доц. Ярошко С. А.

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

“Бібліотеки Python”,

що викладається в межах ОПП “Системний аналіз і управління. Інтелектуальний аналіз даних”

першого (бакалаврського) рівня вищої освіти

для здобувачів зі спеціальності F4 “Системний аналіз та наука про дані”

Львів – 2025

Назва дисципліни	Бібліотеки Python
Адреса викладання дисципліни	Львівський національний університет імені Івана Франка, вул. Університетська 1, м. Львів, Україна, 79000
Факультет та кафедра, за якою закріплена дисципліна	Факультет прикладної математики та інформатики, кафедра програмування
Галузь знань, шифр та назва спеціальності	Галузь знань: F Інформаційні технології Спеціальність: F4 |Системний аналіз та наука про дані
Викладачі дисципліни	Селіверстов Роман Григорович, к. ф.-м. н., доц., доцент кафедри програмування Лисецька Олександра Юріївна, д-р філ. з матем., доцент кафедри математичного моделювання соціально-економічних процесів
Контактна інформація викладачів	*Електронна пошта:* roman.seliverstov@lnu.edu.ua oleksandra.lysetska@lnu.edu.ua *Вебсторінки:* https://ami.lnu.edu.ua/employee/seliverstov-roman-hryhorovych https://ami.lnu.edu.ua/employee/lysetska-o-yu
Консультації з питань навчання по дисципліні відбуваються	Один раз на тиждень згідно з оприлюдненим розкладом консультацій викладача. Можливі онлайнові консультації в середовищі Microsoft Teams. Для погодження часу онлайнових консультацій писати на електронну пошту викладача.
Сторінка курсу	https://ami.lnu.edu.ua/course/dodatkovi-biblioteky-net-python-sa
Інформація про дисципліну	Курс "Бібліотеки Python" є нормативною навчальною дисципліною зі спеціальності F4 "Системний аналіз та наука про дані" для освітньої-професійної програми "Системний аналіз і управління. Інтелектуальний аналіз даних", яку викладають у 5-му семестрі в обсязі 4 кредитів (за Європейською кредитно-трансферною системою ECTS)
Коротка анотація дисципліни	Дисципліна “Бібліотеки Python” фокусується на спеціалізованих модулях стандартної бібліотеки Python та сторонніх бібліотеках для обробки даних і наукових досліджень.
Мета та цілі дисципліни	Мета – формування компетенцій, необхідних для ефективного використання стандартної та сторонніх бібліотек мови Python для вирішення прикладних професійних і наукових задач. Цілі: ознайомлення з технологіями обробки та візуалізації

	даних засобами бібліотек мови Python, оволодіння навичками комплексного використання спеціалізованих бібліотек під час розробки програмного забезпечення для аналізу даних.
Література для вивчення дисципліни	*Основна* 1. Лисецька О. Ю. Бібліотека NumPy: теоретичні основи і застосування : навч. посібник / Лисецька О. Ю., Фундак Л. І., Ярошко С. М. – Львів : ЛНУ імені Івана Франка, 2025. – 180 с. 2. McKinney W. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, 2nd Edition. – O'Reilly Media, 2018. – 582 p. 3. Fowler M. Python Concurrency with asyncio. – Manning, 2022. – 378 p. 4. Mitchell R. Web Scraping with Python: Data Extraction from the Modern Web, 3rd Edition. – O'Reilly Media, 2024. – 352 p. 5. VanderPlas J. Python Data Science Handbook. – O'Reilly Media, 2017. – 548 p. 6. Dua R. Mastering TensorFlow 2.x. – BPB Publications, 2022. – 395 p. 7. Abhinav A. What Is the Python Global Interpreter Lock (GIL)?. – https://realpython.com/python-gil/ 8. Python Module Index. – https://docs.python.org/3/py-modindex.html 9. NumPy. – http://numpy.org. 10. Matplotlib. – http://matplotlib.org. 11. Plotly Open Source Graphing Library for Python. – https://plotly.com/python/ 12. Pandas. – http://pandas.pydata.org. 13. Beautiful Soup Documentation. – https://beautiful-soup-4.readthedocs.io/en/latest/ 14. User Guide – scikit-learn. – https://scikit-learn.org/stable/user_guide.html 15. TensorFlow. – https://www.tensorflow.org/ 16. Gradio. – https://www.gradio.app/ *Додаткова* 17. Маттес Е. Пришвидшений курс Python. – Львів : ВСЛ, 2021 – 600 с. 18. Селіверстов Р., Мельничин А. Основи програмування мовою Python: навч. посібник. – Львів : ЛНУ імені Івана Франка, 2020. – 190 с. 19. Селіверстов Р., Семчук І. Цінова оптимізація замовлень у онлайн-книгарнях на основі вебскрапінгу // Вісник Львівського університету. Серія прикладна математика та інформатика. Випуск 28, 2020. – С. 139-150. http://dx.doi.org/10.30970/vam.2020.28.10941 20. GIL у Python. Ключ до стабільності чи ворог продуктивності. – https://dou.ua/forums/topic/52787/

	21. Python support for free threading. – https://docs.python.org/3/howto/free-threading-python.html 22. Streamlit: A faster way to build and share data apps. – https://streamlit.io/
Обсяг курсу	4 кредити ЄКТС (120 годин). З них 16 годин лекцій, 32 години лабораторних занять, 72 години самостійної роботи.
Очікувані результати навчання	Після завершення цього курсу студент буде *знати:* – призначення основних модулів стандартної бібліотеки Python; – принципи роботи з файлами та файловою системою засобами стандартної бібліотеки; – базові підходи до обробки даних за допомогою регулярних виразів; – призначення та місце бібліотеки NumPy в екосистемі Python для наукових і прикладних обчислень; – структуру та властивості масивів ndarray; – основні операції над масивами (арифметичні, логічні, порівняння), принципи індексації, зрізів і маскування; – базові лінійні алгебраїчні операції в NumPy; – методи обробки пропущених значень; – базові операції трансформації та агрегації даних; – принципи групування даних (groupby); – модель виконання програм у Python та вплив GIL; – основні підходи до паралелізму: багатопоточність, багатопроцесність, асинхронне програмування; – основні принципи вебскрапінгу та його застосування; – можливості бібліотек requests, BeautifulSoup, Scrapy; – правила етичного вебскрапінгу та обмеження доступу; – основні поняття та задачі машинного навчання (класифікація, регресія, кластеризація), етапи побудови проєкту машинного навчання; – основи роботи з бібліотеками scikit-learn і TensorFlow. *вміти:* – використовувати модулі стандартної бібліотеки для розв’язання практичних задач без сторонніх залежностей; – працювати з файлами, каталогами та шляхами у різних операційних системах; – виконувати обробку тексту з використанням регулярних виразів; – працювати з датами та часом, виконувати вимірювання часу виконання програм; – серіалізувати та десеріалізувати дані; – писати зрозумілий, ефективний і переносимий Python-код, спираючись на можливості стандартної бібліотеки; – виконувати векторизовані обчислення без використання циклів;

	– застосовувати індексацію, зрізи та логічні маски для обробки даних; – виконувати базові операції лінійної алгебри; – обчислювати статистичні характеристики наборів даних; – застосовувати NumPy як основу для подальшого вивчення pandas, машинного навчання та аналізу даних; – виконувати фільтрацію, сортування та вибірку даних; – застосовувати агрегацію, групування та зведені таблиці; – виконувати базову підготовку даних (data cleaning), обробляти пропущені значення; – готувати дані для подальшого аналізу, візуалізації або машинного навчання; – представляти, структурувати, обробляти та візуалізувати дані у комп'ютерних системах; – реалізовувати багатопоточні програми з використанням засобів синхронізації; – створювати багатопроцесні програми для CPU-bound задач; – писати асинхронний код; – надсилати HTTP-запити та обробляти відповіді серверів; – парсити HTML-сторінки та витягувати структуровані дані; – навчати та використовувати моделі машинного навчання з scikit-learn і TensorFlow; – розробляти прості графічні інтерфейси; – застосовувати отримані знання для розв'язання конкретних професійних задач.
Компетентності	*Інтегральна (ІК):* ІК01. Здатність розв'язувати складні задачі та практичні проблеми системного аналізу у професійній діяльності або у процесі навчання, що передбачає проведення досліджень із застосуванням теорії та методів системного аналізу, які характеризуються невизначеністю умов і вимог *Загальні (ЗК):* К01. Здатність до абстрактного мислення, аналізу та синтезу К02. Здатність застосовувати знання у практичних ситуаціях К04. Знання та розуміння предметної області та розуміння професійної діяльності К07. Здатність до пошуку, оброблення та аналізу інформації з різних джерел К14. Здатність оцінювати та забезпечувати якість виконуваних робіт *Спеціальні (фахові, предметні) компетентності (ПК):* К 18. Здатність формалізувати проблеми, описані природною мовою, у тому числі за допомогою математичних методів, застосовувати загальні підходи до математичного моделювання конкретних процесів.

	К22. Здатність до комп'ютерної реалізації математичних моделей реальних систем і процесів; проектувати, застосовувати і супроводжувати програмні засоби моделювання, прийняття рішень, оптимізації, обробки інформації, інтелектуального аналізу даних. К23. Здатність використовувати сучасні інформаційні технології для комп'ютерної реалізації математичних моделей та прогнозування поведінки конкретних систем а саме: об'єктно-орієнтований підхід при проектуванні складних систем різної природи, прикладні математичні пакети, застосування баз даних і знань.
Програмні результати навчання	ПР08. Володіти сучасними методами розробки програм і програмних комплексів та прийняття оптимальних рішень щодо складу програмного забезпечення, алгоритмів процедур і операцій. ПР09. Вміти створювати ефективні алгоритми для обчислювальних задач системного аналізу та систем підтримки прийняття рішень. ПР13. Проектувати, реалізовувати, тестувати, впроваджувати, супроводжувати, експлуатувати програмні засоби роботи з даними і знаннями в комп'ютерних системах і мережах.
Ключові слова	Python, стандартна бібліотека, стороння бібліотека, модуль, пакет, скрипт, модульне тестування, асинхронність, багатовимірні масиви, векторизація, серіалізація, датасет, модель, графічний інтерфейс користувача.
Формат курсу	Очний
Теми	Теми курсу подано у додатку "Схема курсу"
Підсумковий контроль, форма	Іспит у кінці семестру
Пререквізити	Студенти потребують базових знань з дисциплін ПП1.2.02 "Вступ до програмування", ПП1.2.06 "Основи програмування", ПП1.2.07 "Пакети прикладних програм" і ПП1.2.08 "Алгоритми обчислювальних процесів". Дисципліна є продовженням курсу ПП1.2.11 "Програмування (.Net, Python)". Окремі теми пов'язані з дисциплінами ПП1.2.17 "Комп'ютерні мережі" і ПП1.2.21 "Програмування (Web)", які вивчаються у тому ж семестрі.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Лекції з мультимедійними презентаціями; лабораторні заняття у вигляді виконання практичних завдань (у тому числі командних); самостійне опрацювання навчальних матеріалів, розміщених у хмарних сховищах (Moodle, Microsoft Teams); обговорення тем та консультації тощо.
Необхідне обладнання	Для проведення лекцій: комп'ютер, проєктор, доступ до мережі

	Інтернет. Для проведення лабораторних занять та виконання завдань: комп'ютер, ОС Windows/Linux, доступ до мережі Інтернет; прикладне програмне забезпечення: інтерпретатор Python 3.x, Python IDLE.
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	Результати складання семестрового контролю оцінюють за шкалою ЄКТС, національною чотирибальною шкалою («відмінно», «добре», «задовільно», «незадовільно») та 100-бальною шкалою Університету згідно п.7.11 Положення про організацію освітнього процесу у ЛНУ імені Івана Франка https://lnu.edu.ua/wp-content/uploads/2024/06/Polozhennia-proorhanizatsiiu-osvitnoho-protsesu_Nova-redaktsiia_2024.pdf. 50 балів нараховують за виконання лабораторних робіт, ще 50 балів – за виконання екзаменаційного завдання. Лабораторні завдання можуть бути індивідуальні та командні. Упродовж семестру студент виконує 8 лабораторних робіт / завдань, кожну(е) з яких оцінюють 6-ма балами. Ще два бали можна отримати за активність на лекційних заняттях. Вчасно виконані завдання оцінюють так (у відсотках від максимальної оцінки): 100% – умови завдання виконано повністю, автор відповідає на всі запитання щодо використаних підходів, чітко інтерпретує отримані результати, немає ознак недоброчесності; 80% – завдання виконано з незначними помилками, бракує виконання окремих пунктів, автор не досить повно пояснює використані підходи, немає ознак недоброчесності; 60% – завдання виконано з помилками, які автор уміє виправити після зауваження викладача, бракує виконання окремих пунктів, на запитання щодо використаних підходів автор відповідає з помилками, немає ознак недоброчесності; 40% – завдання виконано частково, автор не розуміє недоліків поданої роботи, не вміє їх виправити, але вміє інтерпретувати результати; немає ознак недоброчесності; 20% – завдання виконано частково, немає основних пунктів, автор не може самостійно інтерпретувати отримані результати, виправити помилки, немає ознак недоброчесності; 0% – завдання не виконано, не відповідає умові, або ж виявлено ознаки недоброчесності; автор не володіє відповідним теоретичним матеріалом. Можуть бути нараховані додаткові бали за повністю виконане завдання, яке містить кілька способів розв'язування, використовує особливо ефективний спосіб, демонструє креативність автора тощо. Оцінка за екзаменаційне завдання поділена на дві частини: 20 балів за засвоєння теоретичного матеріалу (тест) та 30 балів за написання комп'ютерної програми. У екзаменаційному завданні

	прописані кількість балів за правильну відповідь на кожне питання теоретичної частини і правильну програмну реалізацію кожного структурного елемента завдання практичної частини. Студенти зобов'язані дотримуватися усіх строків, визначених для виконання усіх видів завдань, передбачених курсом. Звіти про виконання лабораторних робіт завантажують у відповідне хмарне сховище. Викладач має право затребувати захист лабораторної роботи. **Усунення академічної заборгованості.** **Форма 2 .** Студент може обрати один з двох способів: 1) залишити за собою бали, отримані за поточний контроль у семестрі та виконати екзаменаційне завдання; 2) виконати тестове завдання (оцінюється 50 балами) та 5 практичних завдань, які полягають у написанні Python-скриптів, які оцінюються у 10 балів кожен. **Форма К.** Студент виконує тестове завдання (оцінюється 50 балами) та 5 практичних завдань, які полягають у написанні Python-скриптів, які оцінюються у 10 балів кожен. Якщо студент набрав менше 51 одного балу, то це веде за собою можливість проходження повторного курсу. **Академічна доброчесність.** Очікується, що роботи студентів будуть їхнім оригінальними дослідженнями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів, здавання чужих звітів становлять, але не обмежують приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в роботі студента є підставою для її незарахуванння викладачем, незалежно від масштабів плагіату чи обману.
Типові питання	Тематика теоретичної (тестової) частини: 1. Призначення та основні функції модулів стандартної бібліотеки Python. 2. Типи даних, основні оператори та методи NumPy. 3. Структури даних pandas та операціії над ними. 4. Принципи побудови та атрибути основних типів діаграм. 5. Багатопотоковість, багатопроцесорність. 6. Асинхронне програмування. 7. Основи вебскрапінгу та html-парсингу. 8. Програмна реалізація класичних задач машинного навчання в scikit-learn і TensorFlow. 9. Програмна реалізація елементів графічних інтерфейсів (вікон та віджетів). 10. Принципи програмної реалізації обробки подій у графічних інтерфейсах. Для успішного складання практичної частини (написання і

	виконання коду) потрібно вміти: 1. Застосовувати функції модулів стандартної бібліотеки для розв’язання типових прикладних задач. 2. Виконувати дії над векторами та матрицями з використанням NumPy. 3. Використовувати засоби бібліотеки pandas для опрацювання таблиць гетерогенних даних. 4. Візуалізовувати дані за допомогою графіків та діаграм. 5. Розподіляти виконання програми між процесами та потоками.
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

Додаток

СХЕМА КУРСУ

Тиждень	Тема, план, короткі тези	Форма діяльності	Завдання, год	Термін виконання
1-2	***Стандартна бібліотека Python.*** *Модулі abc, collections, contextlib, dataclasses, datetime, functools, itertools, os, pickle, re, shutil, sys, typing, venv [8, 17].*	*лекція*	*2*	
	Комплексне використання модулів стандартної бібліотеки Python під час розв'язування прикладних задач.	*лабораторне заняття*	*4*	*2 тижні*
	Тестування з використанням модулів doctest і unittest [6]	*самостійна робота*	*4*	
3-4	***Сторонні бібліотеки Python. NumPy. 2D графіка.*** *Огляд популярних сторонніх бібліотек. NumPy - фундаментальний пакет для наукових обчислень. Векторизація як основа продуктивності. Візуалізація даних з використанням Matplotlib і Plotly [1, 2, 5, 9, 10, 11].*	*лекція*	*2*	
	Застосування NumPy для розв'язування прикладних задач чисельними методами.	*лабораторне заняття*	*4*	*2 тижні*
	3D візуалізація [10, 11].	*самостійна робота*	*4*	
5-6	***Аналіз і обробка табличних даних з pandas.*** *Структури даних Series і DataFrame. Звичайні та ієрархічні індекси. Завантаження та збереження даних. Очищення даних. Фільтрування, групування і агрегування. Об'єднання таблиць. Застосування функцій. Категоризація [2, 5, 12].*	*лекція*	*2*	
	Первинний статистичний аналіз та візуалізація даних з відкритих джерел.	*лабораторне заняття*	*4*	*2 тижні*
	Аналіз часових рядів [2, 5, 12].	*самостійна робота*	*4*	
7-8	***Конкурентне та паралельне програмування.***	*лекція*	*2*	

	Багатопотоковість і GIL (модуль multithreading). Багатопроцесорність (модуль multiprocessing). Інструкції async/await (модуль asyncio). [3, 7, 8]			
	Ефективне розв'язання CPU-bound та IO-bound задач засобами паралельного та конкурентного програмування.	*лабораторне заняття*	*4*	*2 тижні*
	Доцільність GIL (Global Interpreter Lock) та можливість його відключення в нових версіях Python [7, 20, 21].	*самостійна робота*	*4*	
9-10	***Вебскрапінг.** Отримання даних з вебсторінок з використанням BeautifulSoup і Scrapy. Парсинг html-сторінок. Тестування вебсайтів скраперами. [4, 13, 19]*	*лекція*	*2*	
	Розробка мінізастосунку на основі вебскрапінгу.	*лабораторне заняття*	*4*	*2 тижні*
	Легальність та етичні аспекти вебскрапінгу. Файл robots.txt [4].	*самостійна робота*	*4*	
11-12	***Основи машинного навчання з scikit-learn.** Класифікація. Регресія. Кластеризація. [5, 14]*	*лекція*	*2*	
	Класичні задачі класифікації, кластеризації та регресії. Вибір метрик для оцінювання результатів роботи моделі. Підбір гіперпараметрів. Створення пайплайнів.	*лабораторне заняття*	*4*	*2 тижні*
	Зменшення розмірності (Dimensionality Reduction) [5, 14]	*самостійна робота*	*4*	
13-14	***Основи глибокого навчання з TensorFlow.** Класифікація зображень і тексту [6, 15].*	*лекція*	*2*	
	Класифікація зображень і символів з використанням різних типів нейронних мереж.	*лабораторне заняття*	*4*	*2 тижні*
	Оцінка ефективності моделей [6, 15].	*самостійна робота*	*4*	
15-16	***Розробка графічних користувацьких інтерфейсів.** Принципи створення віконних програм. Віджети. Обробка подій. [8, 16, 18]*	*лекція*	*2*	

	Розробка графічного інтерфейсу для десктопного застосунку	*лабораторне заняття*	*4*	*2 тижні*
	Використання Streamlit для розробки простих вебінтерфейсів [22]	*самостійна робота*	*4*	