



## Силабус освітнього компонента

Програма навчальної дисципліни



# Поглиблений курс програмування Python

### Шифр та назва спеціальності

121 – Інженерія програмного забезпечення  
122 – Комп'ютерні науки

### Інститут

ННІ Комп'ютерних наук та інформаційних технологій

### Освітня програма

Інженерія програмного забезпечення  
Комп'ютерні науки та інтелектуальні системи

### Кафедра

Програмна інженерія та інтелектуальні технології управління (321)

### Рівень освіти

Бакалавр

### Тип дисципліни

Спеціальна (фахова), Вибіркова

### Семестр

5

### Мова викладання

Українська, англійська

## Викладачі, розробники



### Коваленко Світлана Миколаївна

Svitlana.Kovalenko@khpri.edu.ua

канд.техн.наук, доцент, доцент кафедри програмної інженерії та інтелектуальних технологій управління

Google Scholar:

<https://scholar.google.com/citations?user=jeD1w74AAAAJ&hl>

ORCID: <https://orcid.org/0000-0001-6770-6778>

Scopus: <https://www.scopus.com/authid/detail.uri?authorId=57212035934>

Web of Science: <https://www.webofscience.com/wos/author/record/F-8252-2017>

[Детальніше про викладача на сайті кафедри](#)

## Загальна інформація

### Анотація

Завданням дисципліни є в подальшому розширенні та поглибленні знань і навичок у програмуванні мовою Python; розширення розуміння концепцій програмування та розвиток вмінь розробки складних програмних рішень з використанням різноманітних технологій та інструментів., подальше здобуття навичок проектування, розробки та тестування програмного забезпечення

### Мета та цілі дисципліни

Формування у студентів теоретичних та практичних знань з роботи з файлами та папками, використання сторонніх бібліотек Python для наукових розрахунків та візуалізації даних; роботи з базами даних на мові Python; створення GUI додатків за допомогою бібліотек Tkinter.

### Формат занять

Лекції, лабораторні роботи, самостійна робота, консультації. Підсумковий контроль – залік.

## Компетентності

- 121-K01. Здатність до абстрактного мислення, аналізу та синтезу.
- 121-K02. Здатність застосовувати знання у практичних ситуаціях.
- 121-K05. Здатність вчитися і оволодівати сучасними знаннями.
- 121-K06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.
- 121-K13. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.
- 121-K15. Здатність розробляти архітектури, модулі та компоненти програмних систем.
- 121-K19. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.
- 121-K22. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.
- 121-K23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.
- 121-K25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.
- 121-K26. Здатність до алгоритмічного та логічного мислення.
- 122-ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.
- 122-ЗК2. Здатність застосовувати знання у практичних ситуаціях.
- 122-ЗК6. Здатність вчитися й оволодівати сучасними знаннями.
- 122-ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.
- 122-ЗК8. Здатність генерувати нові ідеї (креативність).
- 122-ЗК11. Здатність приймати обґрунтовані рішення.
- 122-СК8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.
- 122-СК9. Здатність реалізувати багаторівневу обчислювальну модель на основі архітектури клієнт-сервер, включаючи бази даних, знань і сховища даних, виконувати розподілену обробку великих наборів даних на кластерах стандартних серверів для забезпечення обчислювальних потреб користувачів, у тому числі на хмарних сервісах.
- 122-СК10. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника.
- 122-СК12. Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення.

## Результати навчання

- 121-ПР01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.
- 121-ПР02. Знати кодекс професійної етики, розуміти соціальну значимість та культурні аспекти інженерії програмного забезпечення і дотримуватись їх в професійній діяльності.
- 121-ПР08. Вміти розробляти людино-машинний інтерфейс.
- 121-ПР12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.
- 121-ПР13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.
- 121-ПР15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.
- 121-ПР18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.
- 121-ПР23. Вміти документувати та презентувати результати розробки програмного забезпечення.

122-ПР1. Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.

122-ПР4. Використовувати методи обчислювального інтелекту, машинного навчання, нейромережевої та нечіткої обробки даних, генетичного та еволюційного програмування для розв'язання задач розпізнавання, прогнозування, класифікації, ідентифікації об'єктів керування тощо.

122-ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проектувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування.

122-ПР11. Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проектну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).

122-ПР20. Розробляти архітектуру програмних систем та їх окремих компонент при побудові інтелектуальних систем управління у різних галузях, а також управляти процесами життєвого циклу програмного забезпечення інтелектуальних систем управління.

## Обсяг дисципліни

Загальний обсяг дисципліни 150 год. (5 кредитів ECTS): лекції – 32 год., лабораторні роботи – 32 год., самостійна робота – 86 год.

## Передумови вивчення дисципліни (пререквізити)

Вивчення цієї дисципліни безпосередньо спирається на: "Об'єктно-орієнтоване програмування", "Базовий курс програмування Python"

## Особливості дисципліни, методи та технології навчання

### Методи викладання та навчання:

інтерактивні лекції з презентаціями, дискусії, лабораторні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання.

### Форми оцінювання:

письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування (CAS), онлайн-тести (CAS), підсумковий/семестровий контроль у формі заліку, відповідно до графіку навчального процесу (FAS).

## Програма навчальної дисципліни

### Теми лекційних занять

#### Тема 1. Введення та виведення в/із файлів

Різниця між бінарними та текстовими файлами. Відкриття та закриття текстових файлів. Менеджер контенту with. Режими відкриття файлів. Читання, запис та модифікація даних у файлі.

#### Тема 2. Робота з файлами та папками

Модулі os, pathlib, glob. Отримання контенту папки. Отримання атрибутів файлу. Створення та видалення папок. Пошук файлів по масці. Копіювання, переміщення та перейменування файлів та папок. Модуль zipfile та робота з архівами.

#### Тема 3. Менеджер пакетів та Jupyter Notebook

Модулі стандартної бібліотеки та third-party modules. Pip та його найуживаніші команди. Jupyter Notebook: встановлення, налагодження, використання. Google Colaboratory. Як поділитися Jupyter Notebook. Git та GitHub: встановлення, основи роботи, створення віддаленого репозиторію, рендеринг та поширення ноутбуків.

#### Тема 4. Обробка дати та часу в Python

Команда import та її варіанти використання. Вбудовані модулі datetime та time. Поняття часової мітки та timedelta. Операції з датами та часом. Методи класів Date(), Time() та Datetime().

## Тема 5. Форматування коду Python. PEP8. Лінтери та форматери

PEP8 та правила форматування коду Python для підвищення читабельності. Модуль black. Лінтери. Встановлення та використання форматерів та лінтерів в різних IDE.

## Тема 6. Python декоратори

Поняття об'єктів першого класу. Виклик функції із іншої функції. Замикання. Внутрішні та зовнішні функції. Створення декоратора. \*args, \*\*kwargs. Створення та використання декораторів-класів.

## Тема 7. Python для наукових розрахунків.

Модуль NumPy та історія його створення. ndarrays та їх відмінність від списків Python. Стандартні типи даних NumPy. Атрибути масивів ndarrays. Створення та операції над масивами. Індесування, слайсинг та розширене індексуювання. Підмасиви як відображення та створення копій масивів. Зміна форми масивів: методи reshape() та resize(). Об'єднання та розділення масивів. Розрахунки на ndarrays та універсальні функції. Час виконання розрахунків, магичні команди %timeit і %time. Функції агрегації. Бродкастинг масивів ndarrays. Порівняння на масивах, маски та булева логіка.

## Тема 8. Введення у візуалізацію в Python

Модуль Matplotlib. Стили. Створення графіків в Jupyter Notebook та інших IDE. Збереження графіку у файл, типи графічних файлів. Процедурний та об'єктно-орієнтований інтерфейси Matplotlib. Класи Figure() та Axes(). Простий лінійний графік: створення та налагодження. Діаграми розсіювання (Scatter plots): особливості, створення та налагодження. Бібліотеки seaborn та plotly.

## Тема 9. Створення GUI з TkInter

Поняття GUI та візуального програмування. Класична та подійно-залежна парадигма. Tk та TkInter. Менеджери геометрії. Якорі. Налаштування параметрів вікна. Основні віджети TkInter та робота з ними. Події та їх обробка. Змінні TkInter. Створення меню. Діалогові вікна. Pyinstaller та створення .exe файлу. Створення пакетних файлів для запуску GUI-застосунків.

## Тема 10. Використання баз даних в Python

Python та підтримка баз даних. Бібліотеки для роботи з базами даних. SQLite та її основні характеристики. З'єднання з базою. Створення таблиць. Додавання даних в таблицю. Отримання даних, fetchall(). Фільтрація даних. Видалення та оновлення даних.

## Тема 11. Тестування програмного забезпечення

Основи тестування програмного забезпечення. Автоматичне та ручне тестування. Unit та інтеграційне тестування. Бібліотеки для тестування. Структура тесту. Тестування за допомогою pytest. Методи оцінювання у unittest. Запуск тестів. Розуміння результатів тестів.

## Теми практичних занять

Практичні заняття в рамках дисципліни не передбачені.

## Теми лабораторних робіт

Тема 1. Робота з файлами та папками

Тема 2. Дослідження середовища Jupyter Notebook

Тема 3. Використання бібліотеки NumPy

Тема 4. Візуалізація в Python

Тема 5. Створення GUI-застосунку за допомогою TkInter

Тема 6. Тестування програмного забезпечення

## Самостійна робота

Студентам рекомендуються додаткові матеріали (відео, статті) для самостійного вивчення та опрацювання.

## Література та навчальні матеріали

### Основна література

1. Mariano Anaya. Clean Code in Python: Develop maintainable and efficient code, 2nd Edition, Packt Publishing, 2021, 422p.

2. Robert Johansson. Numerical Python. Scientific Computing and Data. Science Applications with NumPy, SciPy and Matplotlib. Second Edition, 2019. 685 p.

3. Eric Matthes Python Crash Course, 2nd Edition, No Starch Press, 2019, 548p.
4. John Grayson. Python and Tkinter Programming, Manning, 2020, 688p.
5. Agus Kurniawan. Python and SQLite Development, PE Press. 2021, 54p.

## Додаткова література

1. Dusty Phillips, Steven F. Lott Python Object-Oriented Programming, 4th edition, Packt, 2019, 714p
2. Luciano Ramalho. Fluent Python, O'Reilly Media, 2022, 980p
3. Al Sweigart. Automate the Boring Stuff with Python, 2nd edition: Practical Programming for Total Beginners, No Starch Press, 2020, 901p.
4. B. Stephenson: The Python Workbook: A Brief Introduction with Exercises and Solutions, 2nd ed. // Springer, 2019
5. John Canning, Alan Broder, Robert Lafore. Data Structures & Algorithms in Python, Pearson Education

## Система оцінювання

### Критерії оцінювання успішності студента та розподіл балів

100% підсумкової оцінки складаються з результатів оцінювання у вигляді заліку (40%) та поточного оцінювання (60%):

- 6 лабораторних робіт (по 7%);
- 2 контрольні роботи (по 9%).

### Шкала оцінювання

Сума балів	Національна оцінка	ECTS
90–100	Відмінно	A
82–89	Добре	B
75–81	Добре	C
64–74	Задовільно	D
60–63	Задовільно	E
35–59	Незадовільно (потрібне додаткове вивчення)	FX
1–34	Незадовільно (потрібне повторне вивчення)	F

## Норми академічної етики і політика курсу

Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХП»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність. Конфліктні ситуації повинні відкрито обговорюватися в навчальних групах з викладачем, а при неможливості вирішення конфлікту – доводитися до відома співробітників дирекції інституту. Нормативно-правове забезпечення впровадження принципів академічної доброчесності НТУ «ХП» розміщено на сайті: <http://blogs.kpi.kharkov.ua/v2/nv/akademichna-dobrochesnist/>

## Погодження

Силабус погоджено	08.06.2023	Завідувач кафедри Ігор ГАМАЮН
	08.06.2023	Гаранти ОП Андрій КОПП Юлія ЛІТВІНОВА