

Міністерство освіти і науки України
Національний аерокосмічний університет
«Харківський авіаційний інститут»

Кафедра кібербезпеки та інтелектуальних інформаційних технологій (№ 503)

ЗАТВЕРДЖУЮ

Гарант освітньої програми



(підпис)

Сергій САЄНКО

(ім'я та ПРІЗВИЩЕ)

« 29 » серпня 2025 р.

СИЛАБУС ОBOB'ЯЗКОВОЇ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Основи програмування

(назва навчальної дисципліни)

Галузь знань: 13 Механічна інженерія

(шифр і найменування галузі знань)

Спеціальність: 133 Галузеве машинобудування

(код і найменування спеціальності)

Освітня програма: «Комп'ютерний дизайн та 3D-моделювання»

(найменування освітньої програми)

Рівень вищої освіти: перший (бакалаврський)

Силабус введено в дію з 01.09.2025 року

Харків – 2025 р.

Розробник: Лейченко К. М., доц. каф. 503, д-філ (PhD).

(прізвище та ініціали, посада, науковий ступінь та вчене звання)



(підпис)

Силабус розглянуто на засіданні кафедри _____
кібербезпеки та інтелектуальних інформаційних технологій

(назва кафедри)

Протокол № 1 від « 29 » 08 2025 р.

Завідувач кафедри Д.Т.Н., професор

(науковий ступінь та вчене звання)



(підпис)

Вячеслав ХАРЧЕНКО

(ім'я та ПРІЗВИЩЕ)

Представник здобувачів освіти:

студент гр. 449



(підпис)

Олександр РИДА

(ім'я та ПРІЗВИЩЕ)

1. Загальна інформація про викладача



ПІБ: Лейченко Кирило Миколайович

Посада: доцент

Науковий ступінь: доктор філософії (PhD)

Вчене звання:

Перелік дисциплін, які викладає:

- комп'ютерні мережі;
- основи програмування;
- технології Data Science;
- виробнича практика.

Напрями наукових досліджень:

Прокладання та розміщення безпілотних інтелектуальних систем, моніторинг об'єктів критичної інфраструктури, великі дані.

Контактна інформація:

k.leychenko@csn.khai.edu

2. Опис навчальної дисципліни

Форма здобуття освіти	<i>Денна</i>
Семестр	4 семестр
Мова викладання	Українська
Тип дисципліни	<i>Обов'язкова</i>
Обсяг дисципліни: кредити ЄКТС/ кількість годин	<i>денна: 4,5 кредити ЄКТС / 135 годин (64 аудиторних, з яких: лекції – 16, лабораторні – 16, практичні – 32 ; СРЗ – 71)</i>
Види навчальної діяльності	<i>Лекції, лабораторні та практичні заняття, самостійна робота</i>
Види контролю	<i>Поточний контроль, модульний контроль, семестровий контроль – залік</i>
Пререквізити	<i>Лінійна алгебра та аналітична геометрія, Математичний аналіз, Мовні компетентності Іноземна мова</i>

3. Мета та завдання навчальної дисципліни, переліки компетентностей та очікуваних результатів навчання

Мета – надання студентам знань і навичок з основних положень структурного принципу при створенні комп'ютерних програм, вивчення мови програмування високого рівня та стандартних бібліотек функцій.

Завдання: придбання студентами необхідних знань та вмінь в сфері засобів і основних принципів побудови алгоритмів, створення та використання структур даних, вивчення синтаксису мови програмування Python, а також:

- придбання знань про базові елементи мови Python та її філософію;
- придбання знань про характер мови та етапи виконання Python-скрипту;
- придбання знань про структуру програми на мові Python;
- придбання знань про типи даних в мові Python;
- придбання знань про змінні, правила їх іменування та динамічну типізацію;
- придбання знань про рядки та основні методи роботи з ними;
- придбання знань про списки, кортежі, множини та словники, та їхні методи;
- придбання знань про оператори та вирази;
- придбання знань про оператори переходу в мові Python;
- придбання знань про введення/виведення даних;
- придбання знань про умовні оператори, оператори циклу та оператори переходу;
- вміти аналізувати поставлені задачі;
- вміти розробити алгоритм (виконати синтез) поставленої задачі;
- вміти використовувати сучасні графічні редактори для оформлення алгоритму;
- вміти оформлювати звіт та отримані результати по вимогах кафедри.

Компетентності, які набуваються:

Загальні компетентності:

- ЗК1. Здатність до абстрактного мислення;
- ЗК2. Здатність застосовувати знання у практичних ситуаціях;
- ЗК3. Здатність планувати та управляти часом;
- ЗК4. Здатність до пошуку, оброблення та аналізу інформації з різних джерел;
- ЗК5. Здатність генерувати нові ідеї (креативність);
- ЗК10. Навички використання інформаційних і комунікаційних технологій;
- ЗК14. Здатність ухвалювати рішення та діяти, дотримуючись принципу неприпустимості корупції та будь-яких інших проявів не доброчесності.

Фахові компетентності:

ФК1. Здатність застосовувати типові аналітичні методи та комп'ютерні програмні засоби для розв'язування інженерних завдань галузевого машинобудування, ефективні кількісні методи математики, фізики, інженерних наук, а також відповідне комп'ютерне програмне забезпечення для розв'язувань

інженерних задач галузевого машинобудування;

ФК2. Здатність застосовувати фундаментальні наукові факти, концепції, теорії, принципи для розв'язування професійних задач і практичних проблем галузевого машинобудування.

ФК5. Здатність застосовувати комп'ютеризовані системи проектування та спеціалізоване прикладне програмне забезпечення для вирішення інженерних завдань в галузі машинобудування.

Програмні результати навчання:

ПРН5 Аналізувати інженерні об'єкти, процеси та методи;

ПРН16 Застосовувати сучасне загальне та спеціалізоване програмне забезпечення у професійній діяльності.

4. Зміст навчальної дисципліни

МОДУЛЬ 1

Змістовий модуль 1. Основи програмування

Тема 1. Вступ. Знайомство з мовою програмування Python

Стисла анотація: Філософія та особливості Python (читабельність, PEP 8). Інтерпретований та динамічно типізований характер мови. Налаштування середовища (IDE, віртуальні середовища). Виконання простого скрипту. Базові операції введення/виведення (input(), print()). Умовні конструкції (if, elif, else) та оператори порівняння.

Лекція 1: Вступ. Знайомство з мовою програмування

Лабораторна робота 1: Операції з числами та виразами в Python

Практична робота 1: Опрацювання теми лекції згідно поточної теми.

Live-кодінг.

Практична робота 2: Обговорення та розгляд практичних питань згідно теми

Самостійна робота: Опрацювання матеріалу лекції.

Тема 2. Числа, рядки та функції у Python.

Стисла анотація: Базові типи даних (int, float, bool). Арифметичні та логічні оператори. Рядки (str): індексація, зрізи, ключові методи роботи та форматування (f-рядки). Вступ до функцій: оголошення, параметри, повернення значень.

Лекція 2: Числа, рядки та функції.

Лабораторна робота 2: Python: Числа, Рядки та Функції.

Практична робота 3: Опрацювання теми лекції згідно поточної теми. Live-кодінг.

Практична робота 4: Обговорення та розгляд практичних питань згідно теми.

Самостійна робота: Опрацювання матеріалу лекції.

Тема 3. Складні структури даних у Python

Стисла анотація: Колекції в Python: детальний огляд. Списки (list): методи, модифікація, використання. Кортежі (tuple): незмінність та застосування. Словники (dict): робота з парами "ключ-значення". Множини (set)

та операції з ними. Робота з операторами циклу (for, while) та ітерація по колекціях.

Лекція 3: Складні структури даних у Python

Лабораторна робота 3: Складні типи даних

Практична робота 5: Опрацювання теми лекції згідно поточної теми. Live-кодинг.

Практична робота 6: Обговорення та розгляд практичних питань згідно теми

Самостійна робота: Опрацювання матеріалу лекції.

Тема 4. Функції та модулі у Python

*Стисла анотація: Розширена робота з функціями: аргументи за замовчуванням, ключові аргументи, *args та **kwargs. Поняття області видимості (LEGB). Модулі та пакети: створення, імпорт та використання стандартних бібліотек (math, random)..*

Лекція 4: Функції та модулі у Python

Лабораторна робота 4: Розробка та використання функцій та модулів у Python для вирішення математичних та текстових задач

Практична робота 7: Опрацювання теми лекції згідно поточної теми. Live-кодинг.

Практична робота 8: Обговорення та розгляд практичних питань згідно теми

Самостійна робота: Опрацювання матеріалу лекції. Підготовка до модульного контролю.

Модульний контроль 1.

Змістовий модуль 2

Тема 5. Потоки та робота з файлами в Python

Стисла анотація: Огляд потоків: помилки та винятки. Обробка винятків за допомогою блоків try-except-finally. Робота з файловою системою: режими відкриття файлів. Читання та запис даних (текстові та бінарні файли). Сериалізація даних (формати JSON, pickle).

Лекція 5: Потоки та робота з файлами у Python

Лабораторна робота 5: Аналіз успішності студентів з використанням файлів та потоків

Практична робота 9: Опрацювання теми лекції згідно поточної теми. Live-кодинг.

Практична робота 10: Обговорення та розгляд практичних питань згідно теми.

Самостійна робота: Опрацювання матеріалу лекції.

Тема 6. Об'єктно-орієнтоване програмування в Python

Стисла анотація: Основні принципи ООП: інкапсуляція, успадкування, поліморфізм. Оголошення класів та об'єктів. Метод `init` (конструктор). Атрибути класу та екземпляра. Методи успадкування (`Super`). Поняття `Mixin` класів.

Лекція 6: Об'єктно-орієнтоване програмування на Python.

Лабораторна робота 6: Розробка системи "Бібліотека" з використанням об'єктно-орієнтованого програмування в Python.

Практична робота 11: Опрацювання теми лекції згідно поточної теми. Live-кодінг.

Практична робота 12: Обговорення та розгляд практичних питань згідно теми

Самостійна робота: Опрацювання матеріалу лекції.

Тема 7. Основи GUI програмування на Python

Стисла анотація: Вступ до створення графічних інтерфейсів. Огляд популярних GUI-бібліотек (`Tkinter`, `PyQt/PySide`). Створення основного вікна (віджету). Обробка подій (натискання кнопок, введення тексту). Базові елементи дизайну та менеджери компоновання.

Лекція 7: Основи протоколів мережного рівня.

Лабораторна робота 7: Розробка графічного інтерфейсу для системи "Бібліотека" з використанням `Tkinter`.

Практична робота 13: Опрацювання теми лекції згідно поточної теми. Live-кодінг.

Практична робота 14: Обговорення та розгляд практичних питань згідно теми.

Самостійна робота: Опрацювання матеріалу лекції.

Тема 8. Python для аналізу даних

Стисла анотація: Огляд екосистеми для Data Science. Вступ до бібліотеки `NumPy` (робота з масивами). Вступ до `Pandas` (структури `Series` та `DataFrame`). Базові операції з даними: завантаження, фільтрація, групування. Коротка демонстрація візуалізації (`Matplotlib/Seaborn`).

Лекція 8: Python для аналізу даних.

Практична робота 15: Опрацювання теми лекції згідно поточної теми. Live-кодінг.

Практична робота 16: Обговорення та розгляд практичних питань згідно теми.

Самостійна робота: Опрацювання матеріалу лекції. Підготовка до модульного контролю.

Модульний контроль 2.

5. Індивідуальні завдання

Розробити консольну програму-калькулятор на Python, що складається з двох модулів: 1) Калькулятора Індексів М'язів Тіла (ІМТ), який класифікує вагу користувача за допомогою умовних операторів; та 2) Аналізатора Чисел, який використовує цикл для розрахунку суми та кількості парних/непарних чисел у заданому діапазоні.

6. Методи навчання

Проведення аудиторних лекцій, практичних та лабораторних занять, консультацій, а також самостійна робота студентів за матеріалами, опублікованими кафедрою.

7. Методи контролю

Проведення поточного контролю під час проведення лабораторних занять, модульний контроль у вигляді тесту, підсумковий контроль у вигляді тесту, семестровий контроль у письмово-усній формі під час заліку.

8. Критерії оцінювання та розподіл балів, які отримують здобувачі освіти

Таблиця 8.1 – Розподіл балів, які отримують здобувачі освіти

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовий модуль 1			
Робота на лекціях	0...1	4	0...4
Виконання і захист лабораторних (практичних) робіт	0...4	6	0...24
Модульний контроль	0...22	1	0...22
Змістовий модуль 2			
Робота на лекціях	0...1	4	0...4
Виконання і захист лабораторних (практичних) робіт	0...4	6	0...24
Модульний контроль	0...22	1	0...22
Усього за семестр			60...100

Семестровий контроль у вигляді заліку проводиться у разі відмови студента від балів поточного тестування й за наявності допуску до заліку. Під час складання семестрового заліку студент має можливість отримати максимум 100 балів.

Білет для заліку складається із двох теоретичних та одного практичного запитання, максимальна кількість балів за кожне теоретичне запитання, складає 34 балів, а за практичне – 32 балів.

Таблиця 8.2 – Шкали оцінювання: бальна та традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	

60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

Критерії оцінювання роботи здобувач освіти протягом семестру

Задовільно (60-74). Показати мінімум знань та умінь. Захистити не менше 80% від усіх завдань практичних занять. Уміти використовувати методологію розробки програмного забезпечення на мові Python.

Добре (75-89). Твердо знати необхідний обсяг знань для одержання позитивної оцінки, захистити не менше 90% завдань практичних занять. Уміти використовувати практичні навички розробки програмного забезпечення на мові Python.

Відмінно (90-100). Здати всі контрольні точки з оцінкою «відмінно». Досконально знати всі теми та уміти їх застосовувати. Уміти застосовувати навички розробки програмного забезпечення на мові Python.

9. Політика навчального курсу

Відвідування занять. Регуляція пропусків. Характер курсу передбачає необхідність відвідування занять. Здобувачі освіти, які за певних обставин не можуть відвідувати заняття регулярно, повинні протягом тижня узгодити із викладачем графік індивідуального відпрацювання пропущених занять. Окремі пропущені заняття мають бути відпрацьовані на найближчій консультації протягом тижня після їх пропуску. Відпрацювання занять здійснюється усно у формі співбесіди за питаннями, визначеними планом заняття. В окремих випадках дозволяється письмове відпрацювання пропущених занять шляхом виконання індивідуального письмового завдання.

Дотримання вимог академічної доброчесності здобувачами освіти під час вивчення навчальної дисципліни. Під час вивчення навчальної дисципліни здобувані освіти мають дотримуватися загальноприйнятих морально-етичних норм і правил поведінки, вимог академічної доброчесності, передбачених Положенням про академічну доброчесність Національного аерокосмічного університету «Харківський авіаційний інститут» (<https://khai.edu/assets/files/polozhennya/polozhennya-pro-akademichnu-dobrochesnist.pdf>).

Очікується, що роботи здобувачів освіти будуть їх оригінальними дослідженнями або міркуваннями. Списування, втручання в роботу інших здобувачів освіти становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі здобувача освіти є підставою для її незарахування викладачем незалежно від масштабів плагіату чи обману.

Вирішення конфліктів. Порядок і процедури врегулювання конфліктів, пов'язаних із корупційними діями, зіткненням інтересів, різними формами дискримінації, сексуальними домаганнями, міжособистісними стосунками та іншими ситуаціями, що можуть виникнути під час навчання, а також правила етичної поведінки регламентуються Кодексом етичної поведінки в Національному аерокосмічному університеті «Харківський авіаційний інститут»

[\(https://khai.edu.ua/university/normativna-baza/ustanovchi-dokumenti/kodeks-etichnoi-povedinki/\)](https://khai.edu.ua/university/normativna-baza/ustanovchi-dokumenti/kodeks-etichnoi-povedinki/).

10. Методичне забезпечення

Сторінка дисципліни у системі дистанційного навчання «Ментор» [Ел. ресурс]. URL: <https://mentor.khai.edu/course/view.php?id=9538>.

11. Рекомендована література

Базова

1. Свехарт, Ел. Автоматизація рутинних завдань за допомогою Python: Практичний посібник для початківців (Automate the Boring Stuff with Python: Practical Programming for Total Beginners). No Starch Press, 2019.

2. Метъес, Ерік. Python. Вивчаємо програмування (Python Crash Course: A Hands-On, Project-Based Introduction to Programming). 2-ге видання, No Starch Press, 2019.

3. Лутц, Марк. Вивчення Python (Learning Python). 5-те видання, O'Reilly Media, 2013 (або новіше).

Допоміжна

1. Гріффітс, Д. та Баррі, Д. *Head First Python* (A Brain-Friendly Guide). O'Reilly Media, 2016.

2. Доусон, Майкл. *Програмуємо на Python. Для початківців і не тільки* (Python Programming for the Absolute Beginner). 3-тє видання, Course Technology PTR, 2013.

3. Грінфельд, Бруно. *Освоєння Python. Програмування. Створення програм. Обробка даних* (Programming Python). O'Reilly Media (для поглибленого розуміння стандартної бібліотеки та ООП).

4. Шоу, Зед А. *Навчись програмувати, використовуючи Python* (Learn Python the Hard Way). 3-тє видання, Addison-Wesley Professional, 2017.

12. Інформаційні ресурси

1. **Python Documentation** [Електрон. ресурс]. Режим доступу: <https://docs.python.org/3/>

2. **PEP 8 – Style Guide for Python Code** [Електрон. ресурс]. Режим доступу: <https://www.python.org/dev/peps/pep-0008/>

3. **Real Python:** Навчальні статті та туторіали [Електрон. ресурс]. Режим доступу: <https://realpython.com/>

4. **GitHub:** Платформа для розміщення коду та колективної роботи [Електрон. ресурс]. Режим доступу: <https://github.com/>

5. **Stack Overflow:** Форум запитань та відповідей для програмістів [Електрон. ресурс]. Режим доступу: <https://stackoverflow.com/>

6. Сайт Національного аерокосмічного університету «Харківський авіаційний інститут» [Електрон. ресурс]. Режим доступу: <https://www.khai.edu>