

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

Кафедра інформаційно-комунікаційних технологій
ім. О. О. Зеленського (№ 504)

ЗАТВЕРДЖУЮ

Гарант освітньої програми



(підпис)

Олег ЄРЕМЕЄВ

(ім'я та прізвище)

28 серпня 2025 р.

СИЛАБУС ОBOB'ЯЗKОВОЇ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Програмування на Python

(назва навчальної дисципліни)

Галузь знань: F «Інформаційні технології»
(шифр і найменування галузі знань)

Спеціальність: F6 «Інформаційні системи і технології»
(код і найменування спеціальності)

Освітня програма: «Штучний інтелект та інформаційні системи»
(найменування освітньої програми)

Рівень вищої освіти: перший (бакалаврський)

Силабус введено в дію з 01.09.2025

Харків – 2025 р.

Розробник: Рубель О. С., доцент, к.т.н.

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



(підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри _____
інформаційно-комунікаційних технологій ім. О. О. Зеленського (№ 504)

(назва кафедри)

Протокол № 1 від « 28 » серпня 2025 р.

Завідувач кафедри д.т.н., професор

(науковий ступінь і вчене звання)



(підпис)

Володимир ЛУКІН

(ім'я та прізвище)

Погоджено з представником здобувачів освіти:

студент групи 518ст



(підпис)

Вероніка Швець

(ім'я та прізвище)

1. Загальна інформація про викладача



ПІБ: Рубель Олексій Сергійович

Посада: доцент

Науковий ступінь: к.т.н.

Вчене звання: -

Перелік дисциплін, які викладає:

Об'єктно-орієнтоване програмування та ефективні практики, Програмування на Python

Напрями наукових досліджень:

цифрова обробка зображень, машинне навчання

Контактна інформація: o.rubel@khai.edu

2. Опис навчальної дисципліни

Форма здобуття освіти	Денна
Семестр	1, 2-й
Мова викладання	Українська
Тип дисципліни	Обов'язкова
Обсяг дисципліни: кредити ЄКТС/ кількість годин	9 (4,5 у 1-й семестр, 4,5 у 2-й семестр) / 9,5* (5 у 1-й семестр, 4,5 у 2-й семестр) кредити ЄКТС / 270/285* годин (128 аудиторних, з яких: лекції – 64, практичні – 64; СРЗ – 142/157*)
Види навчальної діяльності	Лекції, практичні заняття, самостійна робота
Види контролю	Поточний контроль, модульний контроль, семестровий контроль 1-й семестр – залік, 2-й семестр - іспит
Пререквізити	«Дискретна математика», «Основи інформаційних систем»

* Для здобувачів, які навчаються за скороченим терміном

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

Мета – набуття студентами знань і вмінь з основ програмування, використання їх у своїй практичній роботі, пов'язаній з розробкою програмного забезпечення за допомогою мови програмування Python, використанням усіх елементів імперативної, об'єктно-орієнтованої і функціональної парадигми програмування у ході розробки програмного забезпечення інформаційно-комунікаційних систем і сервісів.

Завдання – вивчення сучасних засобів і інструментів для розробки програмного забезпечення, мови програмування Python, імперативного, об'єктно-орієнтованого і функціонального програмування.

Компетентності, які набуваються:

Інтегральна компетентність:

Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми в області інформаційних систем та технологій, або в процесі навчання, що характеризуються комплексністю та невизначеністю умов, які потребують застосування теорії та методів інформаційних технологій.

Загальні компетентності

Після закінчення цієї програми здобувач освіти буде здатен:

- Здатність до абстрактного мислення, аналізу та синтезу;
- Здатність застосовувати знання у практичних ситуаціях;
- Здатність спілкуватися іноземною мовою;
- Здатність вчитися і оволодівати сучасними знаннями;
- Здатність до пошуку, оброблення та узагальнення інформації з різних джерел;
- Здатність ухвалювати рішення та діяти, дотримуючись принципу неприпустимості корупції та будь-яких інших проявів недоброчесності.

Спеціальні компетентності

Після закінчення цієї програми здобувач освіти буде здатен:

- Здатність проектувати, розробляти та використовувати засоби реалізації інформаційних систем, технологій та інфокомунікацій (методичні, інформаційні, алгоритмічні, технічні, програмні та інші);
- Здатність управляти та користуватися сучасними інформаційно-комунікаційними системами та технологіями (у тому числі такими, що базуються на використанні Інтернет).

Програмні результати навчання:

- Застосовувати знання фундаментальних і природничих наук, системного аналізу та технологій моделювання, стандартних алгоритмів та дискретного аналізу при розв'язанні задач проектування і використання інформаційних систем та технологій;
- Використовувати базові знання інформатики й сучасних інформаційних систем та технологій, навички програмування, технології безпечної роботи в комп'ютерних мережах, методи створення баз даних та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій;
- Демонструвати знання сучасного рівня технологій інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп'ютерних систем та середовищ з метою їх запровадження у професійній діяльності.

4. Зміст навчальної дисципліни

Осінній семестр

МОДУЛЬ 1

Змістовний модуль 1. Основи програмування на Python

Тема 1. Базова програма на Python. Основні відомості, історія виникнення і розвитку Python. Прості програми на Python. Робота програм: написання коду, інтерпретація і виконання. Процес інтерпретації, Python Virtual Machine. Середовище інтегрованої розробки програм (IDE) PyCharm.

Лекція № 1. Вступ до мови Python. Інтерпретація і виконання програм.

Практична робота «Встановлення і налаштування PyCharm. Перша програма».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання практичної роботи.

Тема 2. Робота з командною строкою на Python. Ввід даних в програмах. Аргументи програм, ввід аргументів в командному рядку. Консоль Python shell: базові команди, виконання програм і особливості роботи.

Лекція № 2. Python IDLE.

Практична робота «Робота з Python IDLE».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання практичної роботи.

Тема 3. Модулі і пакети в Python. Пошук і завантаження модулів, вбудовані модулі. Створення власних модулів. Пакети в Python. Робота з документацією. Установлення нових модулів за допомогою pip. Робота з файлом вимог до пакетів – requirements.txt

Лекція № 3. Модулі і пакети.

Практична робота «Робота з модулями».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання практичної роботи.

Тема 4. Базові типи даних і змінні. Базові типи даних: строки, числа. Вивід базових типів. Базові операції з числовими типами даних, написання складних виразів, пріоритетність операцій. Змінні і робота з ними: декларування, присвоєння значень, найменування. Знайомство з документами PEP (Python Enhancement Proposals), прийняті конвенції.

Лекція № 4. Базові типи даних.

Практична робота «Базові типи і змінні».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання практичної роботи.

Тема 5. Типізація і об'єкти в Python. Ідентичність значень і змінних. Копіювання і порівняння значень та змінних. Динамічна і статична, сильна і слабка типізація. Явний вивід типів. Операції порівняння, складні ланцюги порівнянь. Об'єкти, посилання на них, змінні об'єкти і посилання. Копіювання об'єктів. Порівняння за значенням, типом і хешем.

Лекція № 5. Об'єкти в Python.

Практична робота «Об'єкти в Python».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання практичної роботи.

Тема 6. Рядки. Декларування рядків. Багатолінійні рядки. Екранування символів і послідовностей в строках, спеціальні символи. Булеві типи і логіка. Конвертація типів даних в Python у Булев тип. Робота з типом даних bytes, конвертація строк: функції chr() і ord(). Порівняння строк. Створення байтових послідовностей. Конвертація і докодування байтів.

Лекція № 6. Рядки в Python.

Практична робота «Робота з рядками».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання практичної роботи.

Тема 7. Функції в Python. Декларування і виклик функцій. Вбудовані функції. Аргументи, результати роботи функцій. Виконання і повернення вихідних значень. Декомпозиція функцій. Області видимості: глобальні і локальні змінні, правило LEGB, нелокальні змінні. Ефективні практики використання областей видимості. Правильний стиль написання коду – документ PEP8.

Лекція № 7. Функції в Python.

Практична робота «Створення примітивного чат-боту».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання практичної роботи.

Тема 8. Умовні вирази і списки в Python. Умовний вираз if, вбудовані вирази if, конструкції з else і elif – створення складних умовних виразів. Поширений вид колекцій – списки. Робота зі списками– індексація, базові операції, зрізи, діапазони. Вбудовані списки. Сортуювання списків. Виведення списків: з умовними виразом і без, складні вираз виведення. Функції all і any, map і filter.

Лекція № 8. Умовні вирази і списки.

Практична робота «Робота зі списками».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання практичної роботи, підготовка до модульної контрольної роботи.

Модульний контроль 1 «Основи програмування на Python».

Змістовний модуль 2. Робота з даними і об'єктами в Python

Тема 9. Цикли і інші колекції в Python. Умовний цикл while: безкінечний цикл і робота з ним. Цикл for: обробка списків, функція range, обробка введених даних, вбудовані цикли. Переривання і контроль виконання циклів. Інші види колекцій: кортежі, словники і множини. Операції з кортежами, словниками і множинами. Відмінності між видами колекцій. Конвертація об'єктів колекцій між собою.

Лекція № 9. Колекції.

Практична робота «Робота з колекціями».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання практичної роботи.

Тема 10. Стилї оформлення, робота з виключеннями. Розширений розгляд документів PEP і детальний аналіз PEP8. Коментарі, ефективні практики з роботи з коментарями. Документація програм. Помилки в роботі програм. Виключення і їх обробка. Вбудовані виключення і їх створення програмістом.

Лекція № 10. Виключення.

Практична робота «Робота з виключеннями».

Самостійна робота здобувача освіти: опрацювання матеріалу лекції, виконання практичної роботи.

Тема 11. Робота зі строками, генератори і ітератори. Базові строкові методи, пошук в строках, форматування строк, функції split і join. Генератори: функції, вирази, використання на практиці. Ітератори, функції zip і enumerate. Модуль itertools: функції chain, product, combinations.

Лекція № 11. Генератори і ітератори.

Практична робота «Робота з генераторами і ітераторами».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 12. Робота з файлами. Файли в Python. Зчитування і запис файлів. Контекстний менеджер. Робота з CSV-файлами, бібліотека csv. Словники Shelve і операції з ними. Модуль os: робота з директоріями, обробка вмісту, доступ, шляхи файлів і їх валідація.

Лекція № 12. Файли в Python.

Практична робота «Робота з файлами».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 13. Робота зі скриптами. Модуль Argparse: додавання аргументів, розбір аргументів, розпакування, коректна робота з командною строкою. Вивід даних при роботі скриптів, запис логів. Ефективні практики і рекомендації при створенні скриптів.

Лекція № 13. Модуль Argparse.

Практична робота «Робота з Argparse».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 14. Базові елементи функціонального програмування. Лямбда-функції: декларування, виклик, практики з використання. Аргументи в функціях: позиційні аргументи Args, розпакування, значення аргументів за замовчуванням, іменовані аргументи Kwargs і їх розпакування. Декоратори функцій. Рекурсія і функції в Python.

Лекція № 14. Базові елементи функціонального програмування.

Практична робота «Лямбда-функції».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 15. Математичні функції і робота з часом. Математичні функції: модуль math, арифметичні і геометричні методи. Модуль random: робота з випадковими числами і розподілами випадкових величин. Константи, спеціальні типи даних і робота з ними. Огляд Scientific Python Ecosystem. Модуль statistics: статистичні параметри і функції, їх розрахунок. Модулі Time і Datetime: вимірювання часу, обчислення різниці у часі, часові зони, формати запису, розбір часових записів

Лекція № 15. Огляд Scientific Python Ecosystem.

Практична робота «Робота з математичними функціями».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 16. Робота в мережі і регулярні вирази. Модуль json, робота з JSON об'єктами, кодування та декодування. Робота з XML: формат даних, деревоподібна структура, атрибути, конвертація XML. Модулі requests і client: основні запити, обробка запитів, використання мережевих форматів передачі даних. Модуль BeautifulSoup, веб-скрапінг, парсинг даних отриманих з мережі, пошук та виділення потрібних об'єктів чи інформації. Регулярні вирази в Python: екранування, набори і діапазони, виключення, квантори, скорочення, повтори

Лекція № 16. Робота в мережі і регулярні вирази.

Практична робота «Робота зі http запитами».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи, підготовка до модульної контрольної роботи.

Модульний контроль 2 «Робота з даними і об'єктами в Python».

Весняний семестр

Змістовний модуль 3. Об'єктно-орієнтоване програмування на Python

Тема 17. Класи в Python. Декларування класів, атрибути класів, сутності класів. Робота з сутностями класів: створення, ініціалізація, порівняння, посилання на сутність self, атрибути сутностей. Порівняння класів і сутностей: зміна атрибутів, атрибути класів і сутностей, додавання атрибутів.

Лекція № 17. Класи в Python.

Практична робота «Робота з класами».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 18. Методи класів. Декларування методів, виклик, порівняння методів і функцій, повернення результатів роботи. Методи і атрибути: створення атрибутів в методах, їх модифікація, рекомендація з роботи з ними. Магічні методи, математичні магічні методи.

Лекція № 18. Методи класів.

Практична робота «Проект з імітації роботи пристрою з інтерфейсом для користувача».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 19. Розширена робота з класами. Механізм спадкування: основні відомості, об'єкт класу, одиничне і множинне спадкування, перевірка типів, підкласів і сутностей. Перегрузка методів, метод super і робота з класами-предками, розв'язання колізій успадкування при множинному спадкуванні. Абстрактні класи, підкласи, механізм спадкування абстрактних класів. Декоратори в Python: статичні методи, методи класів, методи-властивості.

Лекція № 19. Спадкування класів. Абстрактні класи.

Практична робота «Спадкування класів».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 20. Детальні відомості про об'єктно-орієнтоване програмування. Ієрархії класів, поліморфізм, спадкування. Механізми рефлексії: детектування анотацій і декораторів, маніпуляція методами і атрибутами, робота з сутностями і модифікаторами. Процес серіалізації: основи, підготовка до серіалізації, байтові потоки для об'єктів, серіалізація і десеріалізація об'єктів, створення власних механізмів серіалізації.

Лекція № 20. Ієрархія класів. Робота з сутностями і модифікаторами доступу.

Практична робота «Модифікатори доступу».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 21. Тестування і налагодження в Python. Налагодження і зневадження програм або процес «дебагінгу». Налагодження роботи скриптів, налагодження в командній строці. Основні операції і рекомендації на практиці, процес налагодження в IDE PyCharm, практичні приклади. Використання підтвердження assert, тестування даних при вводі користувачем, блок try-ехсерт, вбудовані методи перевірки.

Лекція № 21. Тестування і налагодження в Python.

Практична робота «Тестування і налагодження».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 22. Unit-тестування на Python. Unit-тести і основні відомості про них. Інструменти unittest, nose, pytest і doctest. Методи підтвердження assert, запуск тестів, вивід даних тестування. Концепція розробки програмного забезпечення test-driven development (TDD), методи setUp і tearDown, використання в командній строці. Аналіз черги викликів (traceback), правильна інтерпретація, типові помилки і їх обробка.

Лекція № 22. Unit-тести.

Практична робота «Unit-тести».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 23. Шаблони проектування. Шаблони ітераторів. Шаблони декоратор, спостерігач, стратегія, стан, одинак, заготовка, адаптер, фасад, команда, абстрактна фабрика, пристосуванець, компоновщик. Практичні приклади, рекомендації і ефективні практики використання.

Лекція № 23. Шаблони проектування.

Практична робота «Шаблони проектування».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 24. Паралельні обчислення і потоки Python. Потоки в Python: основні відомості і проблеми, спільне використання пам'яті, накладні витрати на потоки, блокування. Багатопроцесорна обробка, пули, черги, проблеми використання. Модуль AsyncIO: потоки, виконавці, клієнти.

Лекція № 24. Потоки в Python.

Практична робота «Робота з модулем AsyncIO».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи, підготовка до модульної контрольної роботи.

Модульний контроль «Об'єктно-орієнтоване програмування на Python»

Змістовний модуль 4. Розробка інформаційних сервісів на Python

Тема 25. Алгоритми і бібліотека NumPy. Базові алгоритми в Python: пошук, сортування, реалізація стеків і черг. Масиви і індекси, операції з масивами, типи даних в NumPy. Булеві операції з масивами, робота з декількома масивами, підмодуль linalg.

Лекція № 25. Бібліотека NumPy.

Практична робота «Робота з NumPy».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 26. Робота з базами даних SQLite. Основні відомості про системи керування базами даних, СУБД SQLite. Основи мови запитів SQL: типи даних, літерали, вирази, робота з часом і датами. Запити Data Modification Language: створення, модифікація і видалення записів. Запити Data Definition Language: створення схем і таблиць, обмеження. Запити з завантаженням даних. Використання плагіну з роботи з базами даних у PyCharm і IDE DataGrip.

Лекція № 26. Робота з базами даних SQLite.

Практична робота «Робота з базами даних SQLite».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 27. Об'єктно-реляційні моделі при роботі з базами даних. Бібліотека SQLAlchemy: установка і використання. Об'єктно-реляційна модель SQLAlchemy: з'єднання з базою даних, створення сесії, створення таблиць, декларативне відображення записів, створення базових операцій CRUD. Відміна змін і відкочування до попередніх версій. Міграція баз даних – базові інструменти.

Лекція № 27. Бібліотека SQLAlchemy.

Практична робота «Робота з SQLAlchemy».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 28. Створення простих інформаційних сервісів. Мікрофреймворк Flask: установка, доступний сервер і його налаштування, режим налагодження. Маршрутизація: використання змінних, унікальні посилання і переадресація. Генерація посилань. Статичні файли і шаблони.

Лекція № 28. Фреймворк Flask.

Практична робота «Робота з фреймворком Flask».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 29. Основи фреймворку Django. Установка фреймворка Django. Контролери, додатки, маршрути і маршрутизатори. Консоль Django. Робота з моделями та шаблонами, параметри полів і моделей, редактор моделей. Налагоджувальний веб-сервер і адміністративний веб-сайт Django.

Лекція № 29. Основи фреймворку Django.

Практична робота «Початок роботи з Django».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 30. Проєкти в Django. Зв'язки між моделями, строкове, URL-параметри і параметризовані запити. Форми, контролери-класи, наслідування шаблонів, статичні файли. Налаштування проєкту: основні налаштування, параметри баз даних, списки зареєстрованих додатків і посередників. Мовні налаштування.

Лекція № 30. Форми, контролери-класи, наслідування шаблонів у Django.

Практична робота «Наслідування шаблонів у Django».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 31. Моделі в Django. Створення моделей і полів. Параметри, класи моделей. Створення зв'язків між моделями: «один-до-багатьох», «один-до-одного», «багато-до-багатьох». Валідатори, валідація моделей. Вилучення записів з полів записів, доступ до зв'язаних записів. Вибірка записів: усіх, вилучення одного, набір, пошук, фільтрація, порівняння. Агрегатні обчислення, об'єднання наборів записів.

Лекція № 31. Моделі в Django.

Практична робота «Створення моделей і полів у Django».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи.

Тема 32. Контролери, форми і маршрутизація. Базові класи-контролери, класи для виведення записів, роботи з формами, для виведення хронологічних списків. Вивід даних і директиви, теги шаблонізаторів, фільтри, обробка

статичних файлів, пагінатори. Створення форм зв'язаних з моделями, елементів керування. Обробка форм, вивід форм на екран, валідація, набори форм зв'язаних з моделями. Передача даних до контролерів. Іменовані маршрути, псевдоніми додатків, контролери-функції, формування відповідей на запити, спеціальні відповіді.

Лекція № 32. Контролери, форми і маршрутизація Django.

Практична робота «Проект з реалізації банківського сервісу».

Самостійна робота здобувача освіти: опрацювання матеріалу лекцій, виконання практичної роботи, підготовка до модульної контрольної роботи.

Модульний контроль «Розробка інформаційних сервісів на Python»

5. Індивідуальні завдання

Не передбачені навчальним планом.

6. Методи навчання

Використовуються наступні методи навчання: словесні, наочні та практичні, а саме: проведення лекцій (із застосуванням пояснювально-ілюстративного та проблемного викладання навчального матеріалу), робота на практичних заняттях (із застосуванням як репродуктивного методу, так і проблемного викладання), консультації протягом семестру, самостійна робота за матеріалами, опублікованими кафедрою, використання мережі Internet.

7. Методи контролю

Поточний контроль: виконання практичних завдань, контроль засвоєння навчального матеріалу, запланованого на самостійне опрацювання студентом.

Контроль:

- поточний контроль (оцінювання роботи студентів на практичних заняттях);
- модульний контроль за змістовними модулями;
- семестровий контроль у вигляді заліку (1-й семестр) і семестровий контроль у вигляді іспиту (2-й семестр).

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
Осінній семестр			
Змістовний модуль 1			
Модульний контроль (теми 1–8)	0...30	1	0...30
Виконання і захист практичних робіт	0...5	4	0...20
Змістовний модуль 2			
Модульний контроль (теми 9–16)	0...30	1	0...30
Виконання і захист практичних робіт	0...5	4	0...20
Усього за 1 семестр			0...100
Весняний семестр			
Змістовний модуль 3			
Модульний контроль (теми 17–24)	0...30	1	0...30
Виконання і захист практичних робіт	0...5	4	0...20
Змістовний модуль 4			
Модульний контроль (теми 25–32)	0...30	1	0...30
Виконання і захист практичних робіт	0...5	4	0...20
Усього за 2 семестр			0...100

Семестровий контроль проводиться у разі відмови здобувача освіти від балів підсумкового контролю й за наявності допуску до заліку та іспиту.

Білет для заліку та іспиту складається з тридцяти тестових теоретичних і практичних питань. Максимальна сума балів – 100 балів.

Таблиця 8.2 – Шкали оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційний залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

Критерії оцінювання роботи здобувача освіти протягом семестру

Задовільно (60-74). Показати мінімум знань та умінь. Вміти викладати отримані знання в усній чи письмовій формі; при цьому неповний обсяг засвоєного навчального матеріалу не повинен перешкоджати засвоєнню наступного програмного матеріалу; допускаються окремі істотні помилки, виправлені за допомогою викладача. Захистити всі практичні завдання і здати тестування. Знати синтаксис мови програмування Python і базові принципи програмування. Мати уявлення про об'єктно-орієнтоване програмування.

Добре (75-89). Твердо знати мінімум, захистити всі практичні завдання, здати тестування і поза аудиторну самостійну роботу. Уміти: окрім наведених базових знань застосовувати шаблони проектування для написання гнучкого і придатного для масштабування програмного забезпечення, знати основні елементи і прийоми об'єктно-орієнтованого програмування.

Відмінно (90-100). Здати всі контрольні точки з оцінкою «відмінно». Досконально знати всі теми та уміти застосовувати їх. Відповідати на теоретичні питання на високому рівні в межах конспекту лекцій, рекомендованих підручників та додаткової літератури, вміти аналізувати надану інформацію та пропонувати нестандартні рішення, вміти їх обґрунтовувати. Досконально знати всі теми та уміти застосовувати їх на практиці.

9. Політика навчального курсу

Відвідування занять. Регуляція пропусків. Відвідування практичних занять є обов'язковим. Допускаються пропуски занять з таких поважних причин, як хвороба, сімейні обставини, участь в олімпіаді, творчому конкурсі тощо за попереднього домовленістю та згодою викладача за умови дозволу деканату (викладачу надаються копії довідок, документів або інших матеріалів, які підтверджують причину пропуску). За умов дистанційного навчання поважними причинами пропусків можуть бути порушення у функціонуванні електромереж і збої в роботі Інтернету. Відпрацювання пропущених занять є обов'язковим незалежно від причини пропущеного заняття. Відпрацювання практичного заняття полягає в самостійному виконанні завдань пропущеної практичної роботи.

Дотримання вимог академічної доброчесності здобувачами освіти під час вивчення навчальної дисципліни. Під час вивчення навчальної дисципліни здобувачі 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. Відсутність посилань на використані джерела, фабрикавання джерел, списування, втручання в роботу

інших здобувачів освіти становлять, але не обмежують, приклади можливої академічної недобросовісності. Виявлення ознак академічної недобросовісності в письмовій роботі здобувача освіти є підставою для її незарахування викладачем незалежно від масштабів плагіату чи обману. У разі виявлення плагіату (шляхом порівняння виконаних практичних робіт) робота оцінюватиметься у «0» балів (з можливістю перескладання).

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/>).

10. Методичне забезпечення

1. Навчально-методичне забезпечення дисципліни "Програмування на PYTHON" для бакалаврів / М-во освіти і науки України, Нац. аерокосм. ун-т ім. М. Є. Жуковського "Харків. авіац. ін-т", Каф. вищ. математики та систем. аналізу (№ 405) ; розроб. В. О. Макарічев. - Харків. - Нац. аерокосм. ун-т ім. М. Є. Жуковського "Харків. авіац. ін-т", 2020. - 133 с. - <http://library.khai.edu/library/fulltexts/2021/complex/>

2. Сторінка дисципліни у системі дистанційного навчання *Ментор* знаходиться за посиланням: <https://mentor.khai.edu/course/view.php?id=4148>

3. Алгоритмізація та програмування : конспект лекцій / А. В. Погудін, О. К. Погудіна ; М-во освіти і науки України, Нац. аерокосм. ун-т ім. М. Є. Жуковського "Харків. авіац. ін-т". - Харків. - Нац. аерокосм. ун-т ім. М. Є. Жуковського "Харків. авіац. ін-т", 2020. - 43 с.

4. Основи програмування : навч. посіб. до виконання лаб. робіт , Ч. 1 / Є. В. Соколова, О. В. Лучшева, І. Б. Туркін ; М-во освіти і науки України, Нац. аерокосм. ун-т ім. М. Є. Жуковського "Харків. авіац. ін-т". - Харків. - Нац. аерокосм. ун-т ім. М. Є. Жуковського "Харків. авіац. ін-т", 2018. - 77 с. - <http://library.khai.edu/library/fulltexts/metod/>

11. Рекомендована література

Базова

1. Hillard D. Practices of the Python Pro. / D. Hillard. – NY: Manning Publishing, 2020. – 305 p.
2. Slatkin B. Effective Python: 90 specific ways to write better Python. / B. Slatkin. – Boston: Addison-Wesley Professional, 2020. – 444 p.
3. Lutz M. Learning Python, 5th Edition. / M. Lutz. – O'Reilly Media, 2018. – 1648 p.

Допоміжна

1. Durr C. Competitive Programming in Python: 128 Algorithms to Develop your Coding Skills. / C. Durr, J.J. Vie. – UK: Cambridge University Press, 2021. – 264 p.
2. Lott S.F. Modern Python Cookbook: 133 recipes to develop flawless and expressive programs in Python 3.8. / S.F. Lott. – UK: Packt Publishing, 2020. – 822 p.

12. Інформаційні ресурси

1. <https://www.python.org/>
2. <https://pythoninstitute.org/>
3. <https://pypi.org/>
4. <https://docs.python.org/>
5. <https://openedg.org/openedg-python-institute-certifications/>