

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВЕТЕРИНАРНОЇ МЕДИЦИНИ ТА
БІОТЕХНОЛОГІЙ ІМЕНІ С.З. ГИЖИЦЬКОГО

Факультет механіки, енергетики та інформаційних технологій
Кафедра інформаційних технологій

ПОГОДЖЕНО

Гарант ОПП «Комп'ютерні науки»

Вадим ПТАШНИК

(ім'я та прізвище, підпис)

«27» серпня 2025 року

ЗАТВЕРДЖЕНО

Декан факультету механіки, енергетики та
інформаційних технологій

Степан КОВАЛИШИН

(ім'я та прізвище, підпис)

«28» серпня 2025 року



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«НАВЧАЛЬНА ПРАКТИКА (2 КУРС)»

(назва навчальної дисципліни)

рівень вищої освіти	<u>перший (бакалаврський)</u> (назва освітнього рівня)
галузь знань	<u>12 «Інформаційні технології»</u> (назва галузі знань)
спеціальність	<u>122 «Комп'ютерні науки»</u> (назва спеціальності)
освітня програма	<u>«Комп'ютерні науки»</u> (назва)
вид дисципліни	<u>обов'язкова</u> (обов'язкова / за вибором)

2025-2026 навчальний рік

Робоча програма «Навчальна практика (2 курс)»
(назва навчальної дисципліни)

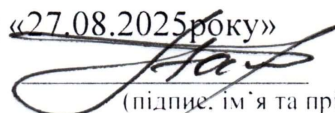
Укладачі: к.т.н., доцент Луб П.М., ст. викладач Махно І.Ю.
(вказати укладачів, їхні посади, наукові ступені та вчені звання)

Робочу програму схвалено на засіданні кафедри інформаційних технологій.

Протокол № 1 від 26.08.2025 року.

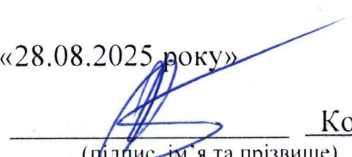
Завідувач кафедри  Тригуба А.М.
(підпис, ім'я та прізвище)

Погоджено навчально-методичною комісією спеціальності
«Комп'ютерні науки»
(назва спеціальності)

Протокол № 8/1 від «27.08.2025 року»
Голова НМКС  Пташник В.В.
(підпис, ім'я та прізвище)

Схвалено рішенням навчально-методичної ради факультету ФМЕІТ
(назва факультету)

Протокол № 1 від «28.08.2025 року»

Голова НМРФ  Ковалишин С.Й.
(підпис, ім'я та прізвище)

Ухвалено вченою радою ФМЕІТ протокол №1 від 28.08.2025 року.

ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	Всього годин
	денна форма здобуття освіти
Семестр	4
Кількість кредитів/годин	6/180
Форма контролю	Захист звіту з практики

1. МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Метою вивчення освітньої компоненти «Навчальна практика (2 курс)» є отримання студентами практичних навичок роботи з сучасними інформаційно-технічними засобами, знайомство з технологіями розробки, упровадження та експлуатації програмного та інформаційного забезпечення, закріплення та поглиблення теоретичних та практичних знань отриманих студентами при вивченні спеціальних дисциплін, набуття навиків для самостійного написання програм відповідної структури.

Завдання навчальної дисципліни передбачають: основні поняття програмування, принципи розробки програми, типові алгоритмічні конструкції, основні структури даних; основи структурного та функційного програмування; основи об'єктно-орієнтованого програмування. В результаті вивчення дисципліни студенти розвивають уміння аналізувати поставлену задачу, обирати доцільний метод її розв'язання; складати алгоритм вирішення задачі; писати та відлагоджувати програми, володіти сучасними засобами розроблення програмного забезпечення.

Після завершення цього курсу студент буде :

Знати:

- головні принципи програмування мовою Python та JS;
- проектування та застосування типів даних, структур даних, списків і кортежів, словників і множин, вказівники повторення, функції, модульні пакети, файли;
- зміст і можливості технології об'єктно-орієнтованого програмування, об'єкти та класи;
- практичну і теоретичну методику оцінювання складності алгоритмів і програм;
- процедури тестування і налагодження програм;

Вміти:

- застосовувати типів даних, структур даних, списків і кортежів, словників і множин, вказівники повторення, функції, модульні пакети, файли до розв'язування задач;
- проектувати модулі та функції загального призначення;
- проектувати класи і використовувати технологію об'єктно-орієнтованого програмування;
- оцінювати складність алгоритмів і програм;
- реалізовувати класичні алгоритми мовами Python та JS;
- тестувати програми; застосовувати системні та програмовані методи налагодження програм.

Пререквізити: для успішного опанування освітньої компоненти необхідно володіти знаннями з інформатики та іноземної (англійської) мови, вищої математики, чисельних методів, архітектури комп'ютерних систем, алгоритмів та структур даних, основ програмування які формуються під час вивчення дисциплін «Іноземна мова», «Вища математика», «Дискретна математика», «Чисельні методи», «Архітектура комп'ютерних систем», «Алгоритми та структури даних» та «Програмування».

Постреквізити: освітньої компоненти створює підґрунтя для опанування наступних дисциплін освітньої програми. Здобуті знання безпосередньо використовуються у курсах «Об'єктно-орієнтоване програмування», «Бази даних», «Якість програмного забезпечення та тестування», «Моделювання систем», «Інформаційна безпека», «Клієнт-серверне програмування»

та «Основи штучного інтелекту», де вони застосовуються для створення програмних рішень, побудови ефективних інформаційних систем, забезпечення їх надійності, безпеки та оптимальної взаємодії між компонентами тощо.

Результати проходження «Навчальної практики (2 курс)» полягають у тому, що студенти отримають знання з основ структурного, функційного та об'єктно-орієнтованого програмування. Вони зможуть пояснювати принципи програмування мовами Python та JavaScript, характеризувати основні типи та структури даних, описувати алгоритмічні конструкції та методи проектування алгоритмів, а також володітимуть теоретичними основами оцінки складності програмних рішень. Окрему увагу буде приділено розумінню процесів тестування, налагодження та забезпечення якості програмного забезпечення.

Студенти набудуть умінь аналізувати поставлені завдання та обирати найбільш доцільні методи їх розв'язання. Вони зможуть розробляти алгоритми і реалізовувати їх засобами мов програмування Python та JavaScript, проектувати й застосовувати функції, модулі, класи та об'єкти у практичних завданнях. Важливим результатом стане здатність проводити тестування програмного коду, здійснювати його налагодження та забезпечувати працездатність програмних рішень.

У процесі навчальної практики студенти розвинути навички роботи із сучасними інструментами розроблення програмного забезпечення. Вони навчатися створювати прикладні програми із застосуванням структурного та об'єктно-орієнтованого підходів, працювати з файлами та модульними пакетами, а також систематично розвиватимуть алгоритмічне мислення. Практичний досвід дозволить їм упевнено застосовувати здобуті знання у майбутніх навчальних дисциплінах і професійній діяльності.

Відповідно до освітньо-професійної програми «Комп'ютерні науки» вивчення дисципліни забезпечує набуття здобувачами таких компетентностей та програмних результатів навчання:

Загальні компетентності	●ЗК2 Здатність застосовувати знання у практичних ситуаціях. ●ЗК3 Знання та розуміння предметної області та розуміння професійної діяльності ●ЗК7 Здатність до пошуку, оброблення та аналізу інформації з різних джерел.
Фахові (спеціальні) компетентності	●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
Програмні результати навчання	●ПРН5 Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій. ●ПРН9 Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук. ●ПРН14 Знати мережні технології, архітектури комп'ютерних мереж, мати практичні навички технології адміністрування комп'ютерних мереж та їх програмного забезпечення.

2. ЗМІСТ ПРАКТИКИ

1. Організаційний етап.

1.1. Проведення загальних організаційних зборів практикантів та первинного інструктажу з техніки безпеки.

1.2. Узгодження, видача завдань та календарних термінів проходження практики.

2. Практичний етап.

2.1. Ознайомлення із обладнанням, документацією, режимами виконання завдання.

2.2. Виконання індивідуального завдання.

3. Завершальний етап.

3.1.Обговорення результатів проведення Навчальної практики.

3.2. Підготовка звіту і щоденника за результатами проходження Навчальної практики.

3.3. Представлення та корегування розробленого звіту.

3.4. Захист звіту із проходження практики.

Зміст індивідуальних завдань конкретизується під час проходження Навчальної практики.

Виконані завдання мають бути використані для оформлення звіту, доповідей тощо.

Індивідуальні завдання можуть бути орієнтовані на повторення базових тем, поглиблення їх, вивчення особливостей мов програмування, фреймворків, інформаційних систем тощо.

Індивідуальні завдання видаються керівником практики перед її початком та оформлюються окремою сторінкою у звіті з проходження практики із підписом керівника.

Тривалість практики – 4 тижні.

3. ТЕМИ ЗАНЯТЬ

Тема	Назва теми (виконуваних робіт)	К-ть годин
Інтерпретована об'єктно-орієнтована мова програмування високого рівня із суворою динамічною типізацією – <i>Python</i>		
Тема 1. Прості типи даних	Об'єкти та змінні; Числа; Цілі числа; Перетворення типів: функція int(); Пріоритет операторів; Числа з плаваючою крапкою; Перетворення типів: функція float(); Математичні функції	6
Тема 2. Рядки	Створення рядків і функція print(); Стил ь форматування: функція format(), f-рядки; Перетворення типів: функція str(); Керуючі символи; Конкатенація рядків; Дублювання рядків; Доступ до елемента рядка за індексом; Зрізи: функція slice[start: end: step]; Довжина рядка: функція len(); Розділення рядка: функція split(); Об'єднання рядків: функція join(); Регістр і вирівнювання; Заміна символів: функція replace(); Введення даних з клавіатури; Функція input()	6
Тема 3. Списки і кортежі	Створення списків; Довжина списку; Перетворення типів: функція list(); Доступ до елементів списку; Списки списків; Зміна елементів списку; Розділення і зворотний порядок елементів списку; Додавання елемента у список	6
Тема 4. Об'єднання списків	Видалення елементів зі списку; Чи є елемент у списку?; Кількість значень у списку; Сорт ування списків; Список у зворотному порядку; Створення числових списків; Кортежі	8
Тема 5. Словники і множини	Створення словника; Перетворення типів: функція dict(); Додавання і зміна елементів словника; Об'єднання словників; Видалення елементів із словника; Чи є ключ у словнику?; Значення із словника; Ключі і значення словника; Обробка відсутніх ключів словника: функція setdefault(); Множини; Складені структури даних	8
Тема 6. Структури коду	Створення і перевірка умов; Булеві значення; Оператори порівнювання; Булеві оператори; Поєднання булевих значень, операторів порівнювання, булевих операторів; Вказівка розгалуження; Що істина, а що хибність?; Команда if; Команда else; Команда elif; Конструкція if/elif/else	8

Тема 8. Фундаментальні операції повторення	Команди while; виведення інформації про виконання коду; Методи роботи з масивами та словниками; Використання функцій map(), filter() та reduce(); Функція enumerate(); Цикли for...of та for...in; Методи роботи з об'єктами; Методы работы с массивами и объектами; Базові синтаксисні конструкції	8
Тема 9. Модулі і пакети	Модуль random: правила імпорту; Модуль math: що всередині?; Стандартна бібліотека Python: короткий огляд модулів; Модулі string і inspect, функції capwords(), isinstance(), getmembers() і repr(); Модуль collections, функція OrderedDict(); Модуль pprint, функція pprint(); Модуль decimal, функції Decimal(), getcontext(); Модуль __name__; Аргументи командного рядка; Пакети; Створення власних пакетів; Менеджер пакетів pip; Віртуальні середовища	8
Тема 10. Файли	Відкриття текстового файлу; Запис даних у текстові файли; Зчитування даних з текстових файлів; Інструкція with; Бінарні файли; Структуровані текстові файли; CSV; XML; JSON; Файли бази даних; PythonDatabase API; Мова запитів SQL; SQLite	8
Тема 11. Класи та об'єкти	Створення і використання класу; Створення класу; Створення екземпляру класу; Доступ до атрибутів; Виклик методів; Створення декількох екземплярів; Робота з класами та екземплярами; Присвоювання атрибуту значення за замовчуванням; Зміна значень атрибутів; Наслідування; Перевизначення методу; Екземпляри як атрибути	6
Динамічна, об'єктно-орієнтована прототипна мова програмування JavaScript		
Тема 1. Основи JavaScript	Структура коду; Сучасний режим, "usestrict"; Змінні; Типи даних; Взаємодія: alert, prompt, confirm; Перетворення типу; Базові оператори, математика; Оператори порівняння; Умовні розгалуження: if, "?"; Логічні оператори; Оператор об'єднання з null "??"; Цикли: while і for; Конструкція "switch"; Функції; Функціональні вирази; Стрілкові функції, основи	6
Тема 2. Якість коду	Налагодження в браузері; Стандарт оформлення коду; Коментарі; Ніндзя-код; Автоматичне тестування за допомогою Mocha; Поліфіли та транспілятори	6
Тема 3. Об'єкти: основи	Об'єкти; Копіювання об'єктів та посилання; Збирання сміття; Методи об'єкта, "this"; Конструктори, оператор "new"; Опціональний ланцюжок "?. "; Тип даних Символ; Перетворення об'єктів в примітиви	8
Тема 4. Типи даних	Методи примітивів; Числа; Рядки; Масиви; Методи масивів; Ітеративні об'єкти; Map та Set; WeakMap та WeakSet; Object.keys, values, entries; Деструктуроване присвоєння; Дата і час; Методи JSON, toJSON	8
Тема 5. Розширена робота з функціями	Рекурсія та стек; Залишкові параметри та синтаксис поширення; Область видимості змінної, замикання; Застаріле ключове слово "var"; Глобальний об'єкт; Об'єкт функції, NFE; Синтаксис "newFunction"; Планування: setTimeout та setInterval; Декоратори та переадресація виклику, call/apply; Прив'язка контексту до функції; Повторення стрілкових функцій	8

Тема 6. Властивості об'єкта, їх конфігурація	Прапори та дескриптори властивостей; Гетери і сетери властивостей	8
Тема 7. Прототипи, наслідування	Успадкування через прототипи; F.prototype; Вбудовані прототипи; Методи прототипів, об'єкти без __proto__	8
Тема 8. Класи	Базовий синтаксис класу; Наслідування класу; Статичні властивості та методи; Приватні та захищені властивості та методи; Розширення вбудованих класів; Перевірка класу: "instanceof"; Міксини	8
Тема 9. Обробка помилок	Робота з помилками, "try...catch"; Нестандартні помилки, розширення Error	8
Тема 10. Проміси, async/await	Введення: колбеки; Проміси; Ланцюжок промісів; Проміси: обробка помилок; Promise API; Промісифікація; Мікрозадачі; Async/await	8
Тема 11. Генератори, розширена ітерація	Генератори; Асинхронні ітератори та генератори	8
Тема 12. Модулі	Вступ до модулів; Експорт та імпорт; Динамічні імпорти	6
	Підготовка та оформлення звіту	10
	Усього годин	180

4. МЕТОДИ НАВЧАННЯ

Навчальна практика здійснюється із застосуванням сучасних інтерактивних та практикоорієнтованих методів, що поєднують словесні (лекція, пояснення, дискусія), наочні (демонстрація, робота з мультимедійними матеріалами) та активні форми (групові проєкти, семінари-дискусії, моделювання ситуацій, аналіз кейсів).

Використання методів мозкового штурму, проблемно-орієнтованих і дослідницьких підходів сприяє розвитку критичного та креативного мислення, уміння працювати в команді й приймати ефективні управлінські рішення. Ефективність забезпечується залученням сучасних цифрових інструментів, програмних середовищ для створення, відлагодження та редагування програм, а також роботи з професійною літературою та науковими публікаціями.

5. МЕТОДИ КОНТРОЛЮ

Після закінчення практики студент зобов'язаний представити звіт про проходження навчальної практики у вигляді реферативної роботи, згідно із отриманим індивідуальним завданням.

Звіт про проходження навчальної практики оформляється на стандартних аркушах формату А4. У звіт включається: титульна сторінка, детальний опис виконуваних робіт, задач, а також фотокопії робочого вікна (скріншоти) із результатом роботи програм тощо.

Після представлення студентом звіту про проходження навчальної практики викладач – керівник практики повинен в 5-ти денний термін прорецензувати звіт, вказати недоліки, що мають місце, оцінити звіт за відповідною шкалою.

Звіт з навчальної практики захищається студентом в 10-ти денний термін після закінчення практики і оцінюється викладачем який керує практикою. За результатами проходження навчальної практики та захисту звіту виставляється кількість балів та оцінка за кредитно-модульною системою.

6. КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ ЗДОБУВАЧІВ ВИЩОЇ ОСВІТИ

Успішність студента оцінюється шляхом проведення поточного контролю.

Максимальна кількість балів з освітньої компоненти «Навчальна практика (2 курс)», яку може отримати студент за всі види роботи, становить 100 балів, визначається як сумарна величина за кожною складовою:

Поточна та самостійна робота (50%)	50 балів.
Оформлення звіту (20%)	20 балів.
Захист звіту (30%)	30 балів.
Разом (підсумковий бал – 100%)	100 балів.

Результати **Поточної та самостійної роботи (ПК)** оцінюються за чотирибальною («2», «3», «4», «5») шкалою. В кінці практики обчислюється середнє арифметичне значення (САЗ) усіх отриманих студентом оцінок з наступним переведенням його у 50-ти бальну шкалу за формулою:

$$ПК = 10 \cdot САЗ$$

Критерії оцінювання знань студентів

Оцінка	Критерії оцінювання
5 («відмінно»)	У повному обсязі володіє навчальним матеріалом, вільно, самостійно та аргументовано його викладає, глибоко і всебічно розкриває зміст, використовуючи обов'язкову та додаткову літературу. Правильно вирішив 90% завдань.
4 («добре»)	Достатньо повно володіє навчальним матеріалом, обґрунтовано його викладає, в основному розкриває зміст завдань, використовуючи обов'язкову літературу. При викладанні окремих питань не вистачає достатньої глибини та аргументації, допускаються несуттєві неточності й незначні помилки. Правильно вирішив більшість завдань.
3 («задовільно»)	У цілому володіє навчальним матеріалом, викладає його основний зміст, але без глибокого всебічного аналізу, обґрунтування та аргументації, допускаючи окремі суттєві неточності та помилки. Правильно вирішив близько половини завдань.
2 («незадовільно»)	Не в повному обсязі володіє навчальним матеріалом. Викладає матеріал фрагментарно та поверхово, без аргументації й обґрунтування, недостатньо розкриває зміст теоретичних і практичних завдань, допускає суттєві неточності. Правильно вирішив меншість завдань.

Переведення підсумкових рейтингових оцінок з дисципліни, виражених у балах за 100-бальною шкалою, у оцінки за національною шкалою та шкалою ECTS.

Таблиця 1 – Шкала оцінювання: національна та ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, диференційованого заліку, курсового проєкту (роботи), практики	для заліку
90–100	A	відмінно	зараховано
82–89	B	добре	
74–81	C		
64–73	D	задовільно	
60–63	E		
35–59	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання

0–34	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни
------	---	--	---

7. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

Підручники і навчальні посібники; електронні видання; інструктивно-методичні матеріали до занять; індивідуальні завдання; електронні варіанти завдань тощо.

Посилання на матеріали розміщені в Мудл та в Інтернеті.

8. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Олексій Васильєв. Програмування мовою Python. Київ: Навчальна книга. Богдан, 2019. 504 с.
2. Lambert K. A. Fundamentals of Python: First Programs, 2nd Edition. Cengage, 2019. 498 p.
3. Підручник мови Python. URL: https://uk.wikibooks.org/wiki/Підручник_мови_Python 9. <https://docs.python.org/3/library/index.html>
4. Путівник мовою програмування Python. URL: https://pythonguide.rozh2sch.org.ua/#_%D0%BF%D1%80%D0%BE%D1%81%D1%82%D1%96_%D1%82%D0%B8%D0%BF%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85
5. Чураков, А.Я. Архітектура ЕОМ [Текст]: посібник/ А.Я. Чураков, С.В. Шаров, О.В. Строкань. Мелітополь: РВЦ МДПУ, 2012. 195 с.
6. Мова програмування JavaScript. URL: <https://uk.javascript.info/getting-started>

9. ІНФОРМАЦІЙНІ РЕСУРСИ

1. Бібліотечно-інформаційні ресурси

- Бібліотека ЛНУВМБ: м. Дубляни, вул. В.Великого, 1; тел. 22-45-915
- Львівська наукова бібліотека ім. Стефаника НАН України: вул. Стефаника, 2; тел. 74-43-72
- Львівська обласна наукова бібліотека: просп. Шевченка, 13; тел. 74-02-26
- Наукова бібліотека ЛНУ ім. Франка, метод. відділ: вул. Драгоманова, 17; тел. 296-42-41
- Центральна міська бібліотека ім. Лесі Українки: вул. Мулярська, 2а; тел. 72-05-81

2. Інформаційні ресурси в Інтернеті

1. Електронний ресурс стосовно ІТ компаній Львівського регіону: http://www.invest-lvivregion.com/it-компанії_ua_285cms.htm
2. Портал об'єктно-орієнтованого програмування: <http://oop.in.ua/tag/FAQ/>
3. Книжки з програмування: як читати і що саме: <https://dou.ua/lenta/articles/programming-books/>
4. Українська технічна література. Програмування: <https://ukrtechlibrary.wordpress.com/tag/програмування/>
5. Задачі програмування із прикладами розв'язку: <http://purecodecpp.com/uk/archives/433>
6. Задачі програмування із прикладами розв'язку: <http://library.nuft.edu.ua/ebook/datathree.php?ID=138>
7. Задачі програмування із прикладами розв'язку: <http://abramov.org.ua/blog/category/opp/obchislennya-%D1%96z-zber%D1%96gannyam-posl%D1%96dovnosti/>
8. Електронне навчання у віртуальному навчальному середовищі ЛНУВМБТ (<https://moodle.lnup.edu.ua/>).