

СИЛАБУС

Базова інформація про дисципліну	
Назва дисципліни	SE023 Тренінг-курс з основ програмування / Training Course on Programming Fundamentals
Рівень фахової передвищої освіти	Фаховий молодший бакалавр
Галузь знань	F «Інформаційні технології»
Спеціальність	F2 «Інженерія програмного забезпечення»
Освітня програма	Інженерія програмного забезпечення
Семестр	2 семестр (на базі повної загальної середньої освіти) 4 семестр (на базі базової середньої освіти)
Курс	1 курс (на базі повної загальної середньої освіти) 2 курс (на базі базової середньої освіти)
Анотація курсу	Предмет передбачає ознайомлення та закріплення знань і вмінь з розширеної тематики основ програмування. Він має на меті вивчення базових інструментів управління кодом, вступ до різних парадигм програмування та основи обробки даних. Основною мовою програмування є Python, також задіюються система контролю версій Git, інтерактивне середовище Jupyter Notebook / Google Colab та профільні бібліотеки, зокрема numpy, pandas та ін.
Сторінка курсу в MOODLE	http://78.137.2.119:2929/course/view.php?id=563
Мова викладання	українська
Лектор курсу	Марченко Станіслав Віталійович, спеціаліст першої категорії канали комунікації: СДН «Moodle»: повідомлення в чаті E-mail: sv.marchenko1989@gmail.com
Місце дисципліни в освітній програмі	
Інтегральна компетентність	Здатність вирішувати типові спеціалізовані задачі інженерії програмного забезпечення, що вимагає застосування положень і методів відповідних наук (математики, інформатики, інформаційних технологій, тощо) та може характеризуватися певною невизначеністю умов; нести відповідальність за результати своєї діяльності; здійснювати контроль інших осіб у визначених ситуаціях.

Базова інформація про дисципліну	
Назва дисципліни	SE023 Тренінг-курс з основ програмування / Training Course on Programming Fundamentals
Рівень фахової передвищої освіти	Фаховий молодший бакалавр
Галузь знань	F «Інформаційні технології»
Спеціальність	F2 «Інженерія програмного забезпечення»
Перелік загальних компетентностей (ЗК)	ЗК05. Знання та розуміння предметної області та розуміння професійної діяльності. ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел. ЗК07. Здатність застосовувати знання у практичних ситуаціях.
Перелік спеціальних компетентностей (СК)	СК01. Здатність алгоритмічно та логічно мислити. СК06. Здатність брати участь у проектуванні програмного забезпечення.
Перелік програмних результатів навчання	РН03. Застосовувати спеціалізовані емпіричні та теоретичні знання у сфері інженерії програмного забезпечення. РН07. Застосовувати стандарти, специфікації в процесах життєвого циклу програмного забезпечення. РН08. Аналізувати вимоги до програмного забезпечення. РН14. Розуміти предметну область, застосовувати знання у професійній діяльності. РН15. Аналізувати та узагальнювати необхідну інформацію з різних джерел та ресурсів для розв'язання професійних задач з урахуванням сучасних досягнень інформаційних технологій.
Опис дисципліни	
Структура навантаження на студента	Загальна кількість годин – 180 Кількість кредитів – 6 Кількість лекційних годин – 0 Кількість практичних занять – 45 Кількість годин для самостійної роботи студентів – 135 Форма підсумкового контролю – залік
Методи навчання	За подачею навчального матеріалу: методи готових знань, дослідницький метод. З огляду на мету навчання: методи здобуття нових знань, метод формування умінь і навичок, метод застосування знань на практиці, методи закріплення знань, умінь і навичок, методи перевірки і оцінювання знань, умінь і навичок.
Зміст дисципліни	

Базова інформація про дисципліну	
Назва дисципліни	SE023 Тренінг-курс з основ програмування / Training Course on Programming Fundamentals
Рівень фахової передвищої освіти	Фаховий молодший бакалавр
Галузь знань	F «Інформаційні технології»
Спеціальність	F2 «Інженерія програмного забезпечення»
Тема 1. Налаштування середовища для розробки мовою Python.	<p>Теоретичний огляд. Інтерфейс командного рядка та види терміналів. Базові команди Bash-терміналу. Структурна організація програмних проєктів. Запуск додатку з консолі. Організація вводу-виводу даних.</p> <p>Самостійне опрацювання. Різновиди Python-додатків та їх мінімальні реалізації. Менеджери пакетів та організація коду Python-проєкту.</p> <p>Практичні завдання. Ініціалізація Python-проєкту та керування залежностями (pip). Отримання та аналіз інформації про середовище виконання. Альтернативні менеджери залежностей. Запуск застосунку з використанням Python-фреймворку.</p>
Тема 2. Програми та їх представлення в пам'яті	<p>Теоретичний огляд. Представлення інформації в пам'яті комп'ютера. Типи даних та представлення програмних об'єктів. Структура коду різними мовами програмування.</p> <p>Самостійне опрацювання. Трансляція коду різними мовами. Байткод та його дизасемблювання.</p> <p>Практичні завдання. Арифметичні операції з двійковими цілими числами. Дробові числа (одинарна та подвійна точність). Текст у двійковій системі. Зчитування та нормалізація аудіо даних (модуль wave). Структура об'єкта в пам'яті. Абстрактне синтаксичне дерево (модуль ast). Байткод Python</p>
Тема 3. Аналіз постановки та хід розв'язування задачі	<p>Теоретичний огляд. Аналіз постановки задачі. Алгоритми: поняття та аналіз. Стандарти форматування коду. Стилiстичне оформлення та документування Python-коду.</p> <p>Самостійне опрацювання. Стратегії та інструменти налагодження коду. Елементарні техніки розробки тестових сценаріїв. Принципи модульного тестування коду. Основи роботи з фреймворком unittest.</p> <p>Практичні завдання. Аналіз постановки задачі. Задачі з прийняттям рішень. Задачі з циклічним повторенням. Дизайн тестів. Еквівалентне розбиття та аналіз граничних значень. Таблиця прийняття рішень. Тестування засобами unittest</p>

Базова інформація про дисципліну	
Назва дисципліни	SE023 Тренінг-курс з основ програмування / Training Course on Programming Fundamentals
Рівень фахової передвищої освіти	Фаховий молодший бакалавр
Галузь знань	F «Інформаційні технології»
Спеціальність	F2 «Інженерія програмного забезпечення»
Тема 4. Функції та елементи функціонального програмування	<p>Теоретичний огляд. Організація Python-коду за допомогою функцій. Виконання програми в функціональному стилі. Функції, ітератори, генератори. Функції вищого порядку.</p> <p>Самостійне опрацювання. Проблематика роботи з датами й часом. Python-бібліотеки для роботи з датами й часом. Функції для роботи з текстом. Регулярні вирази.</p> <p>Практичні завдання. Задачі на створення функцій. Базова робота з часом. Робота з календарем. Обробка рядків. Регулярні вирази. Лямбда-вирази та декоратори. Функції вищого порядку</p>
Тема 5. Вступ до об'єктно-орієнтованого програмування мовою Python	<p>Теоретичний огляд. Інкапсуляція, зв'язки між класами. Робота з вбудованими структурами даних.</p> <p>Самостійне опрацювання. Серіалізація об'єктів у мові Python. Консервування об'єктів. Робота з серіалізаційним представленням json. Робота з серіалізаційним представленням csv.</p> <p>Практичні завдання. Створення власного типу даних. Зв'язки між типами даних. Вбудовані структури даних. Текстові файли. Файлова система. Відкриті формати даних.</p>
Тема 6. Конкурентне виконання коду та елементи мережевого програмування	<p>Теоретичний огляд. Різновиди конкурентного виконання коду. Багатопотокове виконання коду. Бібліотека threading. Паралелізм та мультіпроцесорність. Корутини та асинхронне виконання коду.</p> <p>Самостійне опрацювання. Створення HTTP-сервера. Вебсокети. HTTP-запити. Робота з API. Створення простого сайту.</p> <p>Практичні завдання. Робота з потоками (бібліотека threading). Паралелізм та мультіпроцесорність. Корутини та асинхронне виконання коду. HTTP та вебсокети.</p>
Тема 7. Основні сценарії роботи з локальним Git-репозиторієм	<p>Теоретичний огляд. Принципи роботи системи контролю версій Git. Створення репозиторію в командному рядку. Вітки та галуження в Git. Злиття віток та пов'язані з ним конфлікти. Скасування змін та переміщення по таймлайну розробки.</p> <p>Самостійне опрацювання. Візуальні операції з Git.</p>

Базова інформація про дисципліну	
Назва дисципліни	SE023 Тренінг-курс з основ програмування / Training Course on Programming Fundamentals
Рівень фахової передвищої освіти	Фаховий молодший бакалавр
Галузь знань	F «Інформаційні технології»
Спеціальність	F2 «Інженерія програмного забезпечення»
	Практичні завдання. Конфігурація Git. Базова робота з репозиторієм. Галуження. Злиття віток. Стешинг. Скасування змін
Тема 8. Основи командної роботи з віддаленим репозиторієм	<p>Теоретичний огляд. Базова взаємодія з сервісом GitHub. Оформлення репозиторію. Робочий процес GitHub.</p> <p>Самостійне опрацювання. Редактори коду та інтеграція з GitHub.</p> <p>Практичні завдання. Основи роботи з GitHub. Оформлення репозиторію. Приклади конфліктів при командній роботі. Побудова GitHub Flow.</p>
Тема 9. Використання ШІ-інструментів для створення коду	<p>Теоретичний огляд. Принципи побудови промптів для генерування коду: інструкції, контекст, вхідні дані, індикатор виходу. ШІ-інструменти для генерування коду.</p> <p>Самостійне опрацювання. Генерування Python-коду засобами Claude Code.</p> <p>Практичні завдання. Сертифікат курсу «Claude Code in Action».</p>
Політика дисципліни	
Політика відвідування	<p>Регулярне відвідування всіх видів занять, своєчасність виконання самостійної роботи.</p> <p>За об'єктивних причин (наприклад, хвороба, міжнародне стажування) навчання зорганізується в он-лайн формі за погодженням із керівником курсу.</p>
Політика щодо дедлайнів та перескладання	Роботи, які здаються із порушенням термінів без поважних причин, оцінюються на нижчу оцінку.
Академічна доброчесність	У випадку недотримання політики академічної доброчесності (плагіат, самоплагіат, фабрикація, фальсифікація, списування, обман, хабарництво) передбачено повторне проходження оцінювання.
Використання ШІ	Використання ШІ під час виконання завдань регламентується Положенням «Про використання ШІ в освітньому процесі ЧДБК». Завдання містять маркування щодо регламенту використання ШІ.
Підсумковий контроль	Диференційний залік у кінці семестру за результатами поточної успішності (у формі підсумкового модульного контролю)

Система оцінювання

Система оцінювання підсумкової успішності студентів поділяється на **поточний контроль** та **семестровий контроль**.

Поточний контроль здійснюється протягом семестру і охоплює всі види аудиторної роботи (практичні, семінарські заняття) та виконання індивідуальних завдань. Максимальна кількість балів, яку студент може набрати за цей вид контролю, становить 100.

Підсумковий контроль

Відбувається в формі диференційного заліку (у формі підсумкового модульного контролю).

Розрахунок підсумкової оцінки

Підсумкова оцінка базується виключно на балах, накопичених протягом семестру (**S**). Ваговий коефіцієнт у цьому випадку становить **1**.

Формула: $O=S \times 1$

Види навчальної роботи	Загальна кількість балів
Практичні роботи за темами 1, 7, 8 по 5 балів	15
2, 4 по 10 балів	20
3, 5, 6 по 8 балів	24
9 по 3 бала	3
Модульні контрольні (2 к.р. по 3 бали, 1 к.р. по 2 бали)	8
Індивідуальна самостійна робота	30
Разом	100

Критерії оцінювання для кожного виду навчальної

Критерії оцінювання практичних робіт

100% – звіт з практичної роботи подано вчасно з дотриманням умов академічної доброчесності, термінів звітування, вимог до оформлення та коректності документування, всі завдання розв'язано, захищено усно (якщо передбачено умовою завдань).

90-99% – звіт з практичної роботи подано вчасно з дотриманням умов академічної доброчесності, термінів звітування, наявністю зауважень до оформлення та коректності документування, всі завдання розв'язано, захищено усно (якщо передбачено умовою завдань).

80-89% – звіт з практичної роботи подано вчасно з дотриманням умов академічної доброчесності, термінів звітування, наявністю значних зауважень до оформлення та коректності документування, всі завдання розв'язано, захищено усно (якщо передбачено умовою завдань).

50-79% – звіт з практичної роботи подано вчасно з дотриманням умов академічної доброчесності, термінів звітування, вимог до оформлення та коректності документування, всі завдання розв'язано, проте не захищено усно (якщо це передбачено умовою завдань).

Відсотковий показник визначається стосовно сумарної кількості балів, отриманої за практичну роботу за результатами звітування. Порухення умов академічної доброчесності та/або визначених термінів звітування може позбавити здобувача освіти права подачі звіту за відповідною темою.

Критерії оцінювання модульних робіт

Модульні контрольні роботи складаються з набору закритих тестових завдань та автоматично перевіряються засобами СДН Moodle.

Критерії оцінювання індивідуальних робіт (проектів)

Індивідуальні роботи охоплюють:

- 1) підготовку матеріалів для доповіді за заданою тематикою (10 балів);
- 2) розв'язування та усний захист задач для підкріплення навичок програмування мовою Python (12 балів);
- 3) тренування навичок вайбкодингу в міні-командах з демонстрацією отриманих результатів групової роботи (8 балів).

Критерії оцінювання першої частини індивідуальних робіт поділяються на дві категорії: реферативне повідомлення (6 балів) та усна доповідь (4 бали).

Реферативне повідомлення:

6б. Реферат включає матеріали принаймні з 3 книг та 10 джерел загалом, мінімум одне джерело – англomовне. На кожне з джерел у тексті реферату необхідно мати принаймні одне посилання. Обсяг

реферату повинен бути не меншим за 15 сторінок. Реферат містить демонстрацію практичної частини та розглянутих технологій, наочні візуалізації, оформлений відповідно до вимог стандарту ДСТУ 3008:2015.

4-56. Реферат включає матеріали принаймні з 3 книг та 10 джерел загалом, мінімум одне джерело – англомовне. На кожне з джерел у тексті реферату необхідно мати принаймні одне посилання. Обсяг реферату повинен бути не меншим за 15 сторінок. Реферат містить наочні візуалізації, оформлений відповідно до вимог стандарту ДСТУ 3008:2015.

1-36. Реферат включає матеріали принаймні з 3 книг та 10 джерел загалом, мінімум одне джерело – англомовне. Обсяг реферату повинен бути не меншим за 15 сторінок. Реферат має недоліки оформлення, деякі джерела не мають посилань у тексті. Практична частина відсутня.

06. Завдання не виконано.

Усна доповідь:

46. Здобувач освіти вільно орієнтується в темі доповіді, відповідає на додаткові завдання. Презентаційні матеріали характеризуються повнотою, логічністю викладу, наочністю та інтерактивністю.

36. Здобувач освіти вільно орієнтується в темі доповіді, відповідає на додаткові завдання. Презентаційні матеріали характеризуються повнотою, логічністю викладу та наочністю.

26. Здобувач освіти переважно орієнтується в темі доповіді, частково відповідає на додаткові завдання. Презентаційні матеріали характеризуються логічністю викладу та наочністю, проте можуть бути неповними.

16. Здобувач освіти слабо орієнтується в темі доповіді. Презентаційні матеріали характеризуються логічністю викладу, проте можуть бути неповними або з низькою наочністю.

06. Завдання не виконано.

Критерії оцінювання другої частини індивідуальних робіт відповідають критеріям оцінювання практичних робіт.

Критерії оцінювання третьої частини індивідуальних робіт.

8 б. Здобувач освіти особисто чи в складі мінігрупи (до 3 осіб) з дотриманням умов академічної доброчесності, термінів звітування та чіткого розмежування обов'язків у мінігрупі коректно виконав завдання за обраною тематикою, продемонстрував командну діяльність на прикладі репозиторію з програмним кодом і обґрунтованим базовим описом проекту (основні вимоги, структура коду, процедура запуску), відповів на всі додаткові запитання та загалом показав високий ступінь обізнаності в предметній області.

5-7 б. Здобувач освіти особисто чи в складі мінігрупи (до 3 осіб) з дотриманням умов академічної доброчесності, термінів звітування та чіткого розмежування обов'язків у мінігрупі коректно виконав завдання за обраною тематикою, продемонстрував командну діяльність на прикладі репозиторію з програмним кодом і базовим описом проекту (основні вимоги, структура коду, процедура запуску), частково відповів на додаткові запитання.

1-4 б. Здобувач освіти особисто чи в складі мінігрупи (до 3 осіб) з дотриманням умов академічної доброчесності, термінів звітування та чіткого розмежування обов'язків у мінігрупі коректно виконав завдання за обраною тематикою, продемонстрував командну діяльність на прикладі репозиторію з програмним кодом.

0 б. Завдання не було виконано.

Шкала оцінювання		
ECTS	Бали	Зміст
A	90-100	Бездоганна підготовка в широкому контексті
B	80-89	Повні знання, міцні вміння
C	70-79	Хороші знання та вміння
D	65-69	Задовільні знання, стереотипні вміння

E	60-64	Виконання мінімальних вимог діяльності в стандартних умовах
FX	35-59	Слабкі знання, відсутність умінь
F	1-34	Необхідний повторний курс

Список рекомендованих джерел

1. Костюченко А.О. Основи програмування мовою Python: навчальний посібник. Чернігів: ФОП Баликіна С.М., 2020. 180 с.
2. Трофименко О. Г., Прокоп Ю. В., Задерейко О. В. Алгоритмізація та програмування: навчально-методичний посібник. Одеса : Фенікс, 2020. 310 с.
3. Barrett D. J. Efficient Linux at the Command Line. Sebastopol: O'Reilly Media, 2022. 232p.
4. Chacon S., Straub B. Pro Git (v. 2.1.1440). NY: Apress, 2024. 496p.
5. Tsitoara M. Beginning Git and GitHub. A Comprehensive Guide to Version Control, Project Management, and Teamwork for the New Developer. New York: Apress, 2020. 290p.
6. Lutz M. Learning Python, 6th edition. Sebastopol: O'Reilly Media, 2025. 1172p.
7. Lott S., Phillips D. Python Object-Oriented Programming: Learn how and when to apply OOP principles to build scalable and maintainable Python applications, 5th edition. Birmingham: PACKT Publishing, 2025. 544p.
8. Lott S. Functional Python Programming, 3rd edition: Use a functional approach to write succinct, expressive, and efficient Python code, Third Edition. Birmingham: PACKT Publishing, 2022. 576p.
9. Farrel D. The Well-Grounded Python Developer: How the Pros Use Python and Flask. Shelter Island, NY: Manning Publications, 2023. 272p.
10. Hillard D. Publishing Python Packages: Test, Share, and Automate Your Projects. Shelter Island, NY: Manning Publications, 2023. 226p.

Інтернет ресурси

1. Путівник мовою програмування Python [Електронний ресурс]. URL: <https://pythonguide.rozh2sch.org.ua/> (дата звернення: 29.08.2025р.).
2. Підручник з Python [Електронний ресурс]. URL: <https://docs.python.org/uk/3/tutorial/index.html> (дата звернення: 29.08.2025р.).
3. Програмування для всіх (початок роботи з Python) [Електронний ресурс]. URL: <https://www.coursera.org/learn/python> (дата звернення: 29.08.2025р.).
4. Pythonic way. Don't learn to code. Code to learn! [Електронний ресурс]. URL: <http://pythonicway.com/> (дата звернення: 29.08.2025р.).
5. Python Tutorials [Електронний ресурс]. URL: <https://www.youtube.com/playlist?list=PL-osiE80TeTt2d9bfVyTiXJA-UTHn6WwU> (дата звернення: 29.08.2025р.).
6. Уроки Python для початківців / Програмування на Пітон з нуля [Електронний ресурс]. URL: <https://www.youtube.com/playlist?list=PL7vq4D0vOpQb-nious5dTzUF9GEMp6jiL> (дата звернення: 29.08.2025р.).
7. Chacon S., Straub B. Pro Git (переклад українською) [Електронний ресурс]. URL: <https://git-scm.com/book/uk/v2> (дата звернення: 29.08.2025р.).
8. Вивчай Git Branching [Електронний ресурс]. URL: <https://learnitbranching.js.org/?locale=uk> (дата звернення: 29.08.2025р.).

9. Version Control with Git. URL: <https://www.coursera.org/learn/version-control-with-git> (дата звернення: 29.08.2024р.).
10. Спеціалізація Python for Everybody. URL: <https://www.coursera.org/specializations/python> (дата звернення: 29.08.2025р.).