

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗАТВЕРДЖУЮ

Декан факультету комп'ютерних
інформаційних технологій



Ігор ЯКИМЕНКО

« 29 » 08 2025 р.

ЗАТВЕРДЖУЮ

Проректор з науково-педагогічної
роботи



Віктор ОСТРОВЕРХОВ

2025 р.

РОБОЧА ПРОГРАМА

з дисципліни

«ТЕХНОЛОГІЇ ПРОГРАМУВАННЯ»

Ступінь вищої освіти – **бакалавр**

Галузь знань – **12 Інформаційні технології**

Спеціальність – **125 Кібербезпека та захист інформації**

Освітньо-професійна програма – **Кібербезпека**

кафедра кібербезпеки

Форма навчання	Курс	Семестр	Лекції (год.)	Лабор. (год.)	ІРС (год.)	Тренінг (год.)	СРС (год.)	Разом (год.)	Залік (сем)
Денна	2	4	30	30	4	8	78	150	4

29.08.2015

Тернопіль - 2025

Робочу програму склав доцент кафедри кібербезпеки, к.т.н., доцент Івасьєв Степан Володимирович.

Робоча програма затверджена на засіданні кафедри кібербезпеки, протокол № 1 від 26.08.2025 р.

Завідувач кафедри
кібербезпеки



Василь ЯЦКІВ

Гарант ОП



Михайло КАСЯНЧУК

СТРУКТУРА РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1. Опис дисципліни «Технології програмування»

Дисципліна «Технології програмування»	Галузь знань, спеціальність, СВО	Характеристика навчальної дисципліни
Кількість кредитів – 5	Галузь знань 12 - Інформаційні технології	Статус дисципліни – вибіркова Мова навчання - українська
Кількість залікових модулів – 4	Спеціальність - 125 Кібербезпека та захист інформації	Рік підготовки – 2 Семестр – 4
Кількість змістових модулів – 2	Ступінь вищої освіти – бакалавр	Лекції – 30 год. Лабораторні заняття – 30 год.
Загальна кількість годин – 150		Самостійна робота: 78 год. Тренінг: 8 год. Індивідуальна робота: 4 год.
Тижневих годин: 10 год., з них аудиторних – 4 год.		Вид підсумкового контролю – залік

2. Мета і завдання вивчення дисципліни «Технології програмування»

2.1. Мета вивчення дисципліни.

Метою викладання дисципліни «Технології програмування» є навчити здобувачів ОПІ розробляти програми мовою Python з використанням структурно-модульного методу програмування для кібербезпеки.

2.2 Завдання вивчення дисципліни:

Основне завдання курсу дати здобувачам ОПІ теоретичну та практичну підготовку з основ функціонального та об'єктно-орієнтованого програмування на Python.

Завдання лекційних занять

Мета проведення лекцій полягає у викладанні основних теоретичних положень, синтаксису та алгоритмічних конструкцій мовою програмування python. На лекційних заняттях будуть розглянуті найпоширеніші практики для опрацювання складних структур даних та побудови багатомодульного програмного засобу.

Завдання проведення лабораторних робіт

Проведення лабораторних занять, як одна з основних форм навчального процесу, передбачає поглиблення розуміння і застосування на практиці основних базових конструкцій та структур даних, методів побудови програмних засобів та реалізації повноцінного програмного засобу.

3. Програма навчальної дисципліни «Технології програмування»

Змістовий модуль 1. Основи мови Python.

Тема 1. Типи об'єктів мови Python.

Числа. Базові числові типи. Рядки. Літерали рядків. Рядки в дії. Строкові методи. Вирази форматування рядків. Метод форматування рядків. Загальні категорії типів Списки. Словники. Кортежі. Файли. Ієрархії типів даних в мові Python. Інші базові типи.

Тема 2. Введення в інструкції мови Python.

Структура програми на мові Python. Інструкції присвоювання. Інструкції виразів. Операція print.

Тема 3. Умовна інструкція if і синтаксичні правила.

Умовні інструкції if. Синтаксичні правила мови Python. Перевірка істинності. Тримісний вираз if/ else.

Тема 4. Цикли while і for.

Цикли while, break, continue, pass і else. Цикл for. Правила програмування циклів.

Тема 5. Ітерації і генератори.

Ітератори. Генератори списків. Нові об'єкти в Python 3.0 .

Тема 6. Основи функцій.

Створення функцій. Визначення та виклики функцій.

Тема 7. Області видимості.

Області видимості в мові Python. Інструкція `global`. Області видимості і вкладені функції. Інструкція `nonlocal`.

Тема 8. Аргументи.

Передача аргументів. Спеціальні режими зіставлення аргументів. Функція пошуку мінімуму. Універсальні функції для роботи з множинами. Імітація функції `print` в Python 3.0

Змістовий модуль 2. Модулі в мові Python.

Тема 9. Розширені можливості функцій.

Концепції проектування функцій. Рекурсивні функції. Функції - це об'єкти: атрибути і анотації. Анонімні функції: `lambda`. Відображення функцій на послідовності: `map`. Засоби функціонального програмування: `filter` і `reduce`.

Тема 10. Ітератори і генератори. Генератори списків: функціональні інструменти. Синтаксис генераторів в 3.0. Хронометраж ітераційних альтернатив. Типові помилки при роботі з функціями.

Тема 11. Основи програмування модулів.

Навіщо потрібні модулі? Як працює імпорт. Шлях пошуку модулів. Створення модуля. Використання модулів. Простори назв модулів. Перезавантаження модулів.

Тема 12. Пакети модулів.

Основи імпортування пакетів. Приклад імпортування пакета. Коли використовується операція імпортування пакетів? Імпорт пакету.

Тема 13. Додаткові можливості модулів.

Приховування даних в модулях. Змішані режими використання: `__name__` і `__main__`. Зміна шляху пошуку модулів. Розширення `as` для інструкцій `import` і `from`. Імпорт модулів по імені у вигляді рядка. Перезавантаження модулів. Концепції проектування модулів. Типові проблеми при роботі з модулями.

Тема 14. Основи об'єктно-орієнтованого програмування. Навіщо потрібні класи? Класи генерують безліч екземплярів об'єктів. Класи адаптуються за допомогою успадкування. Класи можуть перевизначати оператори мови Python.

Тема 15. Програмування класів.

Інструкція `class`. Методи. Спадкування. Простори назв. Класи і модулі.

4. Структура залікового кредиту з дисципліни «Технології програмування»

	Кількість годин					
	Лекції	Лаборат. заняття	ІР С	СРС	Тренінг	Контрольні заходи
Змістовий модуль 1. Основи мови Python						
Тема 1. Типи об'єктів мови Python.	2	2	2	5	4	Поточне опитування
Тема 2. Введення в інструкції мови Python	2	2		5		
Тема 3. Умовна інструкція if і синтаксичні правила.	2	2		5		
Тема 4. Цикли while і for.	2	2		5		
Тема 5. Ітерації і генератори.	2	2		6		
Тема 6. Основи функцій.	2	2		5		
Тема 7. Області видимості.	2	2		5		
Тема 8. Аргументи.	2	2		5		
Змістовий модуль 2. Модулі в мові Python						
Тема 9. Розширені можливості функцій	2	2	2	5	4	Поточне опитування
Тема 10. Ітератори і генератори.	2	2		5		
Тема 11. Основи програмування модулів.	2	2		5		
Тема 12. Пакети модулів.	2	2		5		
Тема 13. Додаткові можливості модулів.	2	2		5		

Тема 14. Основи об'єктно-орієнтованого програмування.	2	2		6		
Тема 15. Програмування класів.	2	2		6		
Разом	30	30	4	78	8	

5. Тематика лабораторних робіт

Лабораторна робота № 1.

Тема: Змінні, значення і їх типи. Присвоєння значення. Введення значень з клавіатури.

Мета: Отримати практичні навички роботи зі змінними, значеннями та їх типами.

Питання для обговорення: 1. Змінні, значення і їх типи. 2. Присвоєння значення. 3. Введення значень з клавіатури.

Лабораторна робота № 2.

Тема: Варіанти виводу тексту на екран.

Мета: Отримати практичні навички роботи з операторами введення- виведення.

Питання для обговорення: 1. Форматований ввід/вивід. 2. Псевдографіка. 3. Робота зі стрічками. 4. Парсинг тексту.

Лабораторна робота № 3.

Тема: Вбудовані операції і функції. Основні алгоритмічні конструкції.

Мета: Отримати практичні навички роботи із функціями та вбудованими операціями.

Питання для обговорення: 1. Зарезервовані слова. 2. Алгоритмічні конструкції. 3. Побудова алгоритмів.

Лабораторна робота № 4.

Тема: Оператори умов. Умови рівності / нерівності.

Мета: Отримати практичні навички роботи із конструкціями переходу.

Питання для обговорення: 1. Логічні вирази. Операції над логічними виразами. 2. Конструкції переходу. 3. Безумовний перехід.

Лабораторна робота № 5.

Тема: Цикли і лічильники. Використання команд break, continue.

Мета: Отримати практичні навички роботи із лічильниками, спеціальними операторами циклу.

Питання для обговорення: 1. Поняття лічильника. 2. Види циклів. 3. Вихід з циклу. 4. Вічні цикли.

Лабораторна робота № 6.

Тема: Визначення функцій. Установки та аргументи. Виклики функцій. Оператор повернення return.

Мета: Отримати практичні навички роботи із користувацькими функціями.

Питання для обговорення: 1. Аргументи функції. 2. Область видимості функції. 3. Оператор повернення.

Лабораторна робота № 7.

Тема: Основні стандартні модулі і пакети в Python.

Мета: Отримати практичні навички роботи із модулями та пакетами в Python.

Питання для обговорення: 1. Основні модулі. 2. Пакети в Python. 3. Підключення пакетів.

Лабораторна робота № 8.

Тема: Імпорт модулів. Створення власних модулів і їх імпортування. Спеціалізовані модулі та додатки.

Мета: Отримати практичні навички створення модулів та їхній імпорт.

Питання для обговорення: 1. Можливості створення власних модулів. 2. Можливості імпорту модулів. 3. Набір спеціалізованих модулів.

Лабораторна робота № 9.

Тема: Різні типи послідовностей і загальні для них оператори.

Мета: Отримати практичні навички створення модулів та їхній імпорт.

Питання для обговорення: 1. Різні види послідовностей. 2. Оператори для роботи з послідовностями. 3. Оператори порівняння послідовностей. 4. Оператори друку.

Лабораторна робота № 10.

Тема: Робота зі списками. Вкладені списки. Спеціальні оператори, функції для них.

Мета: Отримати практичні навички роботи зі списками.

Питання для обговорення: 1. Списки. 2. Робота зі списками. 3. Вкладені списки. 4. Спеціальні оператори.

Лабораторна робота № 11.

Тема: Робота зі словниками. Методи словників. Матриці.

Мета: Отримати практичні навички роботи зі словником та матрицями.

Питання для обговорення: 1. Створення словників. 2. Опрацювання словників. 3. Словники, як аргументи.

Лабораторна робота № 12.

Тема: Мережні додатки на Python. Робота із сокетом. Модулі для клієнта WWW.

Мета: Отримати практичні навички роботи із мережевими можливостями мови.

Питання для обговорення: 1. Можливості мережевої взаємодії. 2. Побудова клієнта. 3. Побудова сервера.

6. Тренінг з дисциплін.

Порядок проведення тренінгу:

1. Вступна частина проводиться з метою ознайомлення здобувачів з темою тренінгу.
2. Організаційна частина полягає у створенні робочого настрою у колективі.
3. Практична частина реалізується шляхом виконання завдань з певних проблемних питань теми тренінгу.
4. Підведення підсумків. Обговорення результатів виконаних завдань. Обмін думками з питань, що виносились на тренінг.

Тематика тренінгу: Використання спеціалізованих бібліотек та побудова багатомодульних програмних засобів.

Мета тренінгу: забезпечення здобувачів ОПІ теоретичними знаннями та практичними навичками використання спеціалізованих бібліотек та побудови багатомодульних програмних засобів.

Здобувачі обирають одну із запропонованих тем або формують власну тему дослідження у межах проблематики.

Завдання тренінгу: презентація результатів, за орієнтовною тематикою:

1. Створення словника використовуючи XML.
2. Приклади роботи з послідовностями.
3. Реалізація перевантаження функцій.
4. Управління контекстом виконання.
5. Генератор чисел Фібоначчі.
6. Альтернативний синтаксис доступу до елементів словника.
7. Функтор з генерацією байтокода.
8. Транспонування матриці.
9. Знаходження Факторіал.
10. Рішення квадратного рівняння.
11. Шифрування Цезаря.
12. Шифрування Віженера.
13. Шифрування заміною.
14. Хешування файлів.

7 Самостійна робота

Самостійна робота з курсу «Технології програмування» включає вивчення наведених тем та оцінюється у вигляді тесту:

1. Основні стандартні модулі Python. Поняття модуля. Модулі в Python.
2. Основні стандартні модулі Python. Поняття модуля. Модулі в Python.
3. Стандартні бібліотеки Python.
4. Елементи функціонального програмування. Що таке функціональне програмування? Функціональна програма.

5. Функція: визначення і виклик. Рекурсія.
6. Чисельні алгоритми. Матричні обчислення. Модуль Numeric.
7. Створення масиву. Зрізи. Універсальні функції. Функції модуля Numeric.
8. Функції для роботи з масивами.
9. Обробка текстів. Рядки. Кодування Python-програми.
10. Рядкові літерали. Операції над рядками. Модуль string. Методи рядків.
11. Регулярні вирази. Синтаксис регулярного виразу.
12. Методи об'єкта-шаблону. Приклади шаблонів. Налаштування регулярних виразів.

Приклади застосування регулярного виразу.

13. Об'єктно-орієнтоване програмування. Основні поняття.
14. Абстракція і декомпозиція. Об'єкти. Типи і класи. Інкапсуляція.
15. Відносини між класами. Спадкування.
16. Статичний метод. Метод класу. Метакласи. Мультиметоди.

8. Методи навчання.

У навчальному процесі застосовуються: лекції, в тому числі з використання мультимедійного проектора та інших ТЗН; практичні роботи, індивідуальні заняття; робота в Інтернет.

9. Засоби оцінювання та методи демонстрування результатів навчання

У процесі вивчення дисципліни «Технології програмування» використовуються наступні методи оцінювання навчальної роботи:

- поточне тестування та опитування;
- підсумкове тестування за кожним змістовним модулем;
- оцінювання виконання лабораторних робіт;
- оцінювання тренінгів;
- оцінювання результатів самостійної роботи.

10. Політика оцінювання

- *Політика щодо дедлайнів і перескладання.* Для всіх видів навчальних завдань і контрольних заходів встановлюються чіткі дедлайни. Роботи, здані із порушенням термінів без поважних причин, оцінюються на нижчу оцінку (-10 балів). Перескладання проводиться у встановленому порядку.
- *Політика щодо академічної доброчесності.* Здобувач ОПП зобов'язаний виконувати усі роботи та завдання самостійно. Під час контрольного заходу він може користуватися лише дозволеними допоміжними матеріалами або засобами; йому забороняється в будь-якій формі обмінюватися інформацією з іншими здобувачами ОПП, а також використовувати, розповсюджувати або збирати варіанти чужих робіт чи контрольних завдань..
- *Політика щодо відвідування.* За об'єктивних причин (наприклад, карантин, воєнний стан, хвороба, закордонне стажування) навчання може відбуватись у дистанційній формі за погодженням із керівником курсу та з дозволу дирекції факультету.

11. Критерії, форми поточного та підсумкового контролю

Підсумковий бал (за 100 – бальною шкалою) з дисципліни «Технології програмування» визначається як середньозважена величина, в залежності від питомої ваги кожної складової залікового кредиту %:

Модуль 1		Модуль 2		Модуль 3	Модуль 4	Модуль 5
10%	10%	10%	10%	5%	15%	40%
Поточне оцінювання	Модульний контроль 1	Поточне оцінювання	Модульний контроль 2	Тренінги	Самостійна робота	Екзамен
Оцінка за даний модуль визначається як середнє арифметичне за захист лабораторних робіт №1-6.	Підсумкове модульне тестування за темами №1-8.	Оцінка за даний модуль визначається як середнє арифметичне за захист лабораторних робіт №7-12.	Підсумкове модульне тестування за темами № 9-15.	Визначається як оцінка за виконання та презентацію одного завдання тренінгу.	Підсумкове модульне тестування за результатами опрацювання теоретичного матеріалу. Теоретичні питання: 25 питань по 4 бали - max 100 балів	Теоретичні питання і вигляді тестів: 15 питань по 4 бали - max 60 балів. Практичне завдання: - max 40 балів

Виконання лабораторних робіт:

90-100 балів (Відмінно) - здобувач освіти самостійно, без помилок, виконав усі кроки в рамках лабораторної роботи, правильно задокументував етапи та вільно оперує поняттями та принципами, що відносяться до теми дисципліни.

75-89 балів (Добре) - здобувач освіти виконав завдання лабораторної роботи, проте в процесі виконання допустив кілька дрібних помилок, які не вплинули на кінцевий результат (наприклад, неточна послідовність дій), в процесі роботи виникали додаткові запитання.

60-74 балів (Задовільно) - здобувач освіти виконав завдання лабораторної роботи, але з суттєвими помилками, наприклад, не з першого разу чи не до кінця. Розуміння поставлених у лабораторній роботі завдань є поверхневим та неповним.

1-59 балів (Незадовільно) - здобувач освіти не зміг виконати завдання або результати були повністю невірними. Не продемонстрував базових навичок роботи з програмним забезпеченням.

Підсумкове модульне тестування:

90–100 балів – правильно виконано 90–100% завдань тесту.

75–89 балів – правильно виконано 75–89% завдань тесту.

60–74 бали – правильно виконано 60–74% завдань тесту.

1–59 балів – правильно виконано менше 60% завдань тесту.

Тренінг:

90-100 балів (Відмінно) - здобувач освіти самостійно, без помилок, виконав усі етапи завдання, правильно задокументував усі етапи роботи, та вільно оперує поняттями та принципами дисципліни.

75-89 балів (Добре) - здобувач освіти виконав завдання, але з кількома дрібними помилками, які не вплинули на кінцевий результат, в процесі роботи виникали додаткові запитання.

60-74 балів (Задовільно) - здобувач освіти виконав завдання, але з суттєвими помилками, наприклад, не з першого разу. Розуміння поставлених у тренінгу завдань є поверхневим.

1-59 балів (Незадовільно) - здобувач освіти не зміг виконати завдання або результати були повністю невірними. Не продемонстрував достатній рівень навичок роботи з апаратним та програмним забезпеченням.

Самостійна робота: оцінюється у вигляді тестування за результатами опрацювання теоретичного матеріалу. Теоретичні питання: 25 питань по 4 бали, Максимальна оцінка 100 балів.

Підсумковий екзамен - вид підсумкового контролю, який проводиться з метою оцінювання засвоєння здобувачем вищої освіти теоретичного та практичного матеріалу з дисципліни «Технології програмування». Екзаменаційний білет складається з двох блоків.

Перший блок містить теоретичні питання і вигляді тестів, за кожну правильну відповідь тестування здобувач освіти отримує 4 бали, максимум 60 балів. Виконання практичних завдань передбачається у текстовому вигляді (засобами системи дистанційного навчання) та можуть бути прокоментовані усно для пояснення шляху реалізації поставленого практичного завдання. Максимальна кількість балів за виконання практичного завдання 40 балів.

Шкала оцінювання:

За шкалою університету	За національною шкалою	За шкалою ECTS
90-100	відмінно	A (відмінно)
85-89	добре	B (дуже добре)
75-84		C (добре)
65-74	задовільно	D (задовільно)
60-64		E (достатньо)
35-59	незадовільно	FX (незадовільно з можливістю повторного складання)
1-34		F (незадовільно з обов'язковим повторним курсом)

11. Інструменти, обладнання та програмне забезпечення, використання яких передбачає навчальна дисципліна

№	Найменування	Номер теми
1	Мультимедійний проектор та проєкційний екран	1 -15
2	Персональні комп'ютери	1 -15
3	Комунікаційне програмне забезпечення (Zoom) для проведення занять у режимі он-лайн (за необхідності)	1 -15
4	Комунікаційна навчальна платформа (Moodle) для організації дистанційного навчання (за необхідності)	1 -15
5	Наявність доступу до мережі Інтернет	1 -15
6	Microsoft Windows, Microsoft Office 2013, Mozilla Firefox, FoxitReader, WinZip, DjVu Viewer, Total Commander, SQL Server 2015 Express, Visual Studio Code, Comuniti, Python IDE. PyCharm	1-15

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

1. Програмування в Python. Теорія і практика : навч. посіб. Київ : Видавництво Ліра-К, 2023. 462 с. AloorRavi S. Metaprogramming with Python. Packt , 2022. — 402 p..
2. Програмування числових методів мовою Python : підруч. / А. В. Анісімов, А. Ю. Дорошенко, С. Д. Погорілий, Я. Ю. Дорогий ; за ред. А. В. Анісімова. — К. : Видавничо-поліграфічний центр "Київський університет", 2014. — 640 с.
3. Чичкарьов Є.А., Зінченко О.В., Єльченко С.В.. «Прикладне програмування на Python. Частина 1. Основи програмування на Python». - 2022.
4. Amos David et al. Python Basics: A Practical Introduction to Python 3. 4th Edition. — David Amos, Dan Bader, Joanna Jablonski, Fletcher Heisler. — Real Python, 2021. — 643 p.
5. Balti Haythem. Job Ready Python. John Wiley & Sons, 2022. — 714 p.
6. Beazley David M. Python Distilled. Pearson, 2022. — 352 p.
7. Behrman Kennedy. Foundational Python for Data Science. Addison-Wesley/Pearson Education, 2021. — 256 p.
8. Brian O. Python Testing with pytest: Simple, Rapid, Effective, and Scalable. 2nd Edition. — Pragmatic Bookshelf, 2022. — 274 p.
9. Gaddis Tony. Starting out with Python. 5th Edition, Global Edition. — Pearson, 2022. — 895 p.
10. David Michael. Hand-On Python Tutorial: Project-Based Introduction to Programming For Beginners and Professionals. Independently published, 2021. — 208 p.
11. Mastrodomenico Rob. The Python Book. Wiley, 2022. — 258 p.