

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

ЗАТВЕРДЖУЮ
Декан факультету комп'ютерних
інформаційних технологій
Ігор ЯКИМЕНКО
“29” вересня 2025 р.

ЗАТВЕРДЖУЮ
Проректор з науково-
педагогічної роботи
Віктор ОСТРОВЕРХОВ
“29” вересня 2025 р.

ЗАТВЕРДЖУЮ
Директор навчально-наукового
інституту новітніх освітніх технологій
Святослав ПИТЕЛЬ
“29” вересня 2025 р.

РОБОЧА ПРОГРАМА

з дисципліни « Micropython для одноплатних мікрокомп'ютерів »

ступінь вищої освіти – бакалавр

галузь знань – 12 “Інформаційні технології”

спеціальність – 125 “Кібербезпека”

освітньо-професійна програма – „Кібербезпека”

Кафедра комп'ютерної інженерії

Форма навчання	Курс	Семестр	Лекції (год.)	Лабораторії (год.)	ІРС (год.)	Тренінг (год.)	Самост. робота студ. (год.)	Разом (год.)	Залік сем.)	Екз. (сем.)
Денна	4	7	30	14	3	10	93	150	7	-
Заочна	4	7	8	4	0	0	138	150	7	-

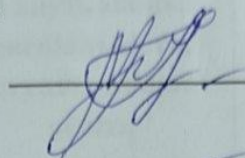
Тернопіль – ЗУНУ
2025

Робочу програму склав к.т.н., доцент кафедри КІ

Юрій БАТЬКО

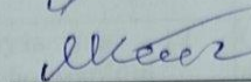
Робоча програма затверджена на засіданні кафедри комп'ютерної інженерії,
протокол №1 від 26 серпня 2025 р.

Завідувач кафедри



Леся ДУБЧАК

Гарант ОП



Михайло КАСЯНЧУК

Спеціальність – 127	Рівень підготовки:
Комп'ютерна інженерія	Діагн. – 4
	Діагн. – 4
	Семестр – 7
	Діагн. – 7
	Зачисл. – 7
Ступінь вищої освіти – бакалавр	Діагн. – 30 год.
	Зачисл. – 8 год.
	Лабораторні заняття:
	Діагн. – 14 год.
	Зачисл. – 4 год.
Заступник керівника групи –	Самостійна робота:
Діагн. – 150 год.	Діагн. – 53 год.
Зачисл. – 150 год.	Зачисл. – 123 год.
	Греши – 10 год.
	Індивідуальна робота:
	Діагн. – 3 год.
Гарант ОП	Вид шк. якого контролює:
Діагн. 7 семестр – 11,5 год.	Діагн. 7 семестр – 11,5 год.
Зачисл. 7 семестр – 3 год.	Зачисл. 7 семестр – 3 год.

СТРУКТУРА РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
" MICROPYTHON ДЛЯ ОДНОПЛАТНИХ МІКРОКОМП'ЮТЕРІВ "

1. Опис дисципліни " Micropython для одноплатних мікрокомп'ютерів "

Дисципліна - «Micropython для одноплатних мікрокомп'ютерів»	Галузь знань, спеціальність, ступінь вищої освіти	Характеристика навчальної дисципліни
Кількість кредитів – <i>Денна – 5,</i> <i>Заочна – 5</i>	галузь знань – 12 „Інформаційні технології”	Статус дисципліни – вибіркова Мова навчання – українська
Кількість залікових модулів: 3	Спеціальність – 123 „Комп'ютерна інженерія”	Рік підготовки: <i>Денна - 4,</i> <i>Заочна - 4</i> Семестр: <i>Денна – 7</i> <i>Заочна – 7</i>
Кількість змістових модулів – 2	Ступінь вищої освіти - бакалавр	Лекції: <i>Денна - 30 год.,</i> <i>Заочна – 8 год.</i> Лабораторні заняття: <i>Денна - 14 год.</i> <i>Заочна – 4 год.</i>
Загальна кількість годин – <i>Денна – 150 год.,</i> <i>Заочна – 150 год.</i>		Самостійна робота: <i>Денна - 93 год.</i> <i>Заочна – 138 год.</i> Тренінг – 10 год. Індивідуальна робота: <i>Денна -3 год.</i>
Тижневих годин: <i>Денна: 7 семестр – 11,5 год.,</i> з них аудиторних - 3 год.		Вид підсумкового контролю <i>Денна: 7 семестр – залік</i> <i>Заочна: 7 семестр – залік</i>

2. Мета й завдання дисципліни " Micropython для одноплатних мікрокомп'ютерів "

2.1. Мета вивчення дисципліни

Програма та тематичний план дисципліни „Micropython для одноплатних мікрокомп'ютерів ” орієнтовані на глибоке та ґрунтовне засвоєння студентами систематичних знань та практичних навичок використання сучасних одноплатних комп'ютерів для проектування, реалізації та впровадження програмно-апаратних комплексів під час розв'язання актуальних теоретичних задач та практичних завдань.

Метою дисципліни є формування у студентів теоретичних знань та практичних навичок з проектування та реалізації програмно-апаратних систем на основі одноплатних мікрокомп'ютерів.

2.2. Завдання вивчення дисципліни

Завданням вивчення дисципліни є вивчення науково-практичного інструментарію організації та проектування завдань автоматизованого тестування програмних, апаратних і програмно-апаратних систем та їх складових за допомогою сучасних наборів практик для автоматизації процесів створення та реалізації складних систем.

В результаті вивчення дисципліни студенти мають знати:

- сучасні методи і та технології для розроблення автоматизованих тестових наборів для програмних, апаратних та програмно-апаратних систем.
- нові технології автоматизації процесів тестування, брати участь розробці та модернізації тестових стратегій з врахуванням технічних вимог, інтегрувати елементи автоматизації в процеси тестування під час створення програмно-апаратних комплексів.

В результаті вивчення дисципліни студенти мають вміти:

- розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності;
- застосовувати знання функціональних особливостей та технічних характеристик програмно-апаратних комплексів для автоматизації процесу тестування програмних, апаратних і програмно-апаратних систем в процесі вирішення технічних задач спеціальності.
- розробляти планування тестування, розробляти тестові сценарії та процедури, проводити аналіз особливостей розробки й генерації тестів
- здійснювати пошук інформації в різних джерелах для розв'язання задач зі спеціальності.

3. Програма навчальної дисципліни „ Micropython для одноплатних мікрокомп'ютерів ”

Змістовий модуль 1. Основи мови програмування Python

Теми 1. Вступ в технологію програмування одноплатних мікрокомп'ютерів.

1. Завдання та основні визначення дисципліни. 2. Етапи розвитку одноплатних мікрокомп'ютерів. 3. Технічні характеристики та можливості. 4. Сфери застосування одноплатних мікрокомп'ютерів. 5. Переваги та недоліки одноплатних мікрокомп'ютерів.

Література: 1, 2, 3.

Тема 2. MicroPython для одноплатних мікрокомп'ютерів.

1. Особливості мови Python. 2. Переваги та недоліки мови Python. 3. Середовища розробки та тестування. 4. Структура програми на Python.

Література: 4, 7.

Тема 3. Типи даних, операції, оператори.

1. Основні типи даних: рядки, списки, кортежі, словники. 2. Особливості введення/виведення. 3. Основні математичні операції. 4. Основні логічні операції. 5. Оператори розгалуження.

Література: 4, 7.

Тема 4. Складні структури даних та операції з ними.

1. Масиви, стрічки, списки. 2. Циклічні оператори. 3. Поняття масиву та операції над ним. 4. Стрічки та алгоритми їх обробки. 5. Списки механізми генерації та обробки. 6. Кортежі, списки, основні методи роботи з кортежами та списками.

Література: 4, 7.

Змістовий модуль 2. Проектування програмно-апаратних систем на основі одноплатних мікрокомп'ютерів та мови python.

Тема 5. Функціональне програмування та механізм рекурсії.

1. Поняття функції. 2. Локальні та глобальні змінні. 3. Повернення значень. 4. Рекурсія та особливості її використання.

Література: 2, 5.

Тема 6. Робота з файлами на мові python.

1. Поняття файлу. 2. Файлове введення-виведення. 3. Читання рядків за допомогою файлів ітераторів. 4. Робота із двійковими файлами.

Література: 5, 8.

Тема 7. Додаткові модулі для розширення функціональних можливостей мови python.

1. Бібліотека math. 2. Модуль NumPy для роботи з математичними операціями. 3. Модуль Matplotlib та особливості його застосування. 4. Модуль SciPy та чисельні алгоритми.

Література: 4, 9.

Тема 8. Обробка та візуалізація табличних даних.

1.Формування та обробка табличних даних. 2. Способи візуалізації табличних даних. 3. Модуль Pandas, робота з таблицями та аналіз даних. 4. Обробка файлів формату csv.

Література: 1, 3.

Тема 9. Проектування та реалізації графічного інтерфейсу користувача.

1. Поняття графічного інтерфейсу користувача. 2.Особливості проектування та реалізації програм із GUI. 3. Огляд графічних бібліотек: Tkinter, PyQt. 4. Приклади реалізації програмних додатків з графічним інтерфейсом на мові python.

Література: 2, 5.

Тема 10. Одноплатний мікрокомп'ютер Raspberry Pi.

1. Raspberry Pi основні характеристики. 2.Дистрибутиви Raspberry Pi. 3. Встановлення ОС за допомогою NOOBS. 4. Встановлення дистрибутива Raspbian за допомогою завантажувальної картки.

Література: 2, 3.

Тема 11. Об'єктно-орієнтований підхід для одноплатних мікрокомп'ютерів.

1. Особливості об'єктно-орієнтованого підходу для одноплатних мікрокомп'ютерів. 2. Об'єктно-орієнтоване програмування у Python. 3. Створення та ініціалізація класів. 4. Декоратори.

Література: 1, 3, 6, 7.

Тема 12. Управління програмними проектами.

1. Розробка програмно-апаратних системи Raspberry Pi + Python. 2. Файловий менеджер. 3. Створення скріншотів. 4. Інсталяція пакета Samba.

Література: 1, 2, 3.

Тема 13. Програмні проекти на основі Raspberry Pi + Python.

1. Охоронна система на основі відеопотоку. 2. FTP сервер. 3. Веб-сервер. 4. Торрент-клієнт. 5. Відеотрансляція за допомогою веб-камери.

Література: 1, 2, 3.

**4. Структура залікового кредиту дисципліни
„Micropython для одноплатних мікрокомп'ютерів”.**

(денна форма навчання)

	Кількість годин					
	Лекції	Лабораторні заняття	Самостійна робота	Індивід. робота	Тренінг	Контрольні заходи
Змістовий модуль 1						
Теми 1. Вступ в технологію	2	-	7	1	5	Опитування

програмування одноплатних мікрокомп'ютерів						
Тема 2. MicroPython для одноплатних мікрокомп'ютерів.	2	-	7			Опитування
Тема 3. Типи даних, операції, оператори	2	2	7			Опитування
Тема 4. Складні структури даних та операції з ними	2	4	7			Опитування
Змістовий модуль 2						
Тема 5. Функціональне програмування та механізм рекурсії	2	4	7	2	5	Опитування
Тема 6. Робота з файлами на мові python	2	-	7			Опитування
Тема 7. Додаткові модулі для розширення функціональних можливостей мови python.	2	-	7			Опитування
Тема 8. Обробка та візуалізація табличних даних	2	-	7			Опитування
Тема 9. Проектування та реалізації графічного інтерфейсу користувача.	2	-	9			Опитування
Тема 10. Одноплатний мікрокомп'ютер Raspberry Pi.	2	-	9			Опитування
Тема 11. Об'єктно-орієнтований підхід для одноплатних мікрокомп'ютерів	4	4	7			Опитування
Тема 12. Управління програмними проектами	2	-	7			Опитування
Тема 13. Програмні проекти на основі Raspberry Pi + Python	6		7			Опитування
Разом	30	14	93	3	10	

(заочна форма навчання)

	Кількість годин		
	Лекції	Лабораторні заняття	Самостійна робота
Змістовий модуль 1			
Теми 1. Вступ в технологію програмування одноплатних мікрокомп'ютерів	-	-	8
Тема 2. MicroPython для одноплатних мікрокомп'ютерів.	-	-	10
Тема 3. Типи даних, операції, оператори	2	2	10
Тема 4. Складні структури даних та операції з ними	2	-	10
Змістовий модуль 2			
Тема 5. Функціональне програмування та механізм рекурсії	-	-	10
Тема 6. Робота з файлами на мові python	-	-	10
Тема 7. Додаткові модулі для розширення функціональних можливостей мови python.	-	-	10
Тема 8. Обробка та візуалізація	2	-	10

табличних даних			
Тема 9. Проектування та реалізації графічного інтерфейсу користувача.	-	-	10
Тема 10. Одноплатний мікрокомп'ютер Raspberry Pi.	-	-	10
Тема 11. Об'єктно-орієнтований підхід для одноплатних мікрокомп'ютерів	2	2	10
Тема 12. Управління програмними проектами	-	-	15
Тема 13. Програмні проекти на основі Raspberry Pi + Python	-	-	15
Разом	8	4	138

5. Тематика лабораторних занять

Лабораторна робота №1.

Тема: Основні поняття та принципи побудови програм на мові python.

Мета: Отримати теоретичні знання та практичні навички по розробці програмних додатків на мові програмування python.

1. Середовище розробки PyCharm;
2. Типи даних на мові python
3. Оператори вводу/виводу на мові python

Література: 2, 4.

Лабораторна робота №2.

Тема: Математичні операції та алгоритми розгалуження на мові python

Мета: Отримати практичні навички для розв'язання прикладних задач з елементами механізмів розгалуження та складних математичних виразів.

Питання для обговорення:

1. Функції для виконання математичних операцій;
2. Функції для реалізації механізмів розгалуження в програмі
3. Генерація випадкових чисел

Література: 2, 4, 5.

Лабораторна робота №3.

Тема: Циклічні алгоритми та масиви даних

Мета: Отримати теоретичні знання та практичні навички при реалізації програмних додатків обробки одновимірних та багатовимірних масивів

Питання для обговорення:

1. Функції для реалізації циклічних алгоритмів;
2. Одновимірні масиви на мові python;
3. Багатовимірні масиви на мові python.

Література: 2, 5.

Лабораторна робота №4.

Тема: Одноплатні комп'ютери Raspberry Pi

Мета: Отримати теоретичні знання та практичні навички по розробці та реалізації системи на основі одноплатного комп'ютера Raspberry Pi

Питання для обговорення:

1. Технічні характеристики Raspberry Pi;
2. Особливості написання програмного коду для Raspberry Pi;
3. Завантаження керуючого програмного коду.

Література: 2, 5.

6. Тренінг з дисципліни «Micropython для одноплатних мікрокомп'ютерів»

№п/п	Вид роботи	Порядок проведення тренінгу
1	Огляд сучасних одноплатних комп'ютерів	1. аналіз та встановлення операційних систем для функціонування одноплатних комп'ютерів; 2. дослідження технічних можливостей Raspberry Pi
2	Аналіз та проектування апаратної системи	• постановка задачі; • опис технічного завдання; • моделювання програмно-апаратної розробки
3	Програмна реалізація проектованої системи	1. реалізація керуючого блоку апаратної системи; 2. реалізація графічного інтерфейсу користувача.
4	Тестування розробленої системи на основі Raspberry Pi	– вибір та обґрунтування тестової вибірки; – опис вхідних та вихідних даних; – перевірка правильності роботи реалізованого програмного додатку

7. Самостійна робота студентів (денна форма навчання)

№ п/п	Тематика	Завдання
1	Проектний документ Python Enhancement Proposal	Вибір та обґрунтування принципів проектування та реалізації програмного додатку
2	Межі видимості об'єктів в Python	
3	Типи даних List та Tuples для зберігання колекцій об'єктів у Python.	
4	Модулі та пакети в Python	Проектування та структури програмної частини проекту
5	Відмінності між масивом та списком у Python	
6	Негативне значення індексів у Python	
7	Управління пам'яттю в Python	
8	Перевантаження операторів у Python	Реалізація програмної частини проекту
9	Особливості реалізації специфікаторів доступу в Python	
10	Ламбда-функція в Python	
11	Декоратори в Python	Реалізація програмно-апаратного проекту
12	Одноплатні комп'ютери	
13	Операційні системи одноплатних комп'ютерів	
14	Особливості реалізації програмних проектів на мові Python для одноплатних комп'ютерів	

(заочна форма навчання)

№ п/п	Тематика
1	Проектний документ Python Enhancement Proposal
2	Межі видимості об'єктів в Python
3	Типи даних List та Tuples для зберігання колекцій об'єктів у Python.
4	Поняття Pass - операції в Python
5	Модулі та пакети в Python
6	Відмінності між масивом та списком у Python
7	Негативне значення індексів у Python
8	Управління пам'яттю в Python
9	Перевантаження операторів у Python
10	Передача аргументів за значенням та посиланням
11	Спеціальні синтаксиси *args та *kwargs
12	Особливості реалізації специфікаторів доступу в Python
13	Ламбда-функція в Python
14	Декоратори в Python
15	Global Interpreter Lock

8. Засоби оцінювання та методи демонстрування результатів навчання

У навчальному процесі застосовуються: лекції, в тому числі з використанням мультимедіапроектора та інших ТЗН; практичні заняття; індивідуальні заняття, самостійна робота студента, робота в Інтернет.

У процесі вивчення дисципліни „Micropython для одноплатних мікрокомп'ютерів” використовуються наступні засоби оцінювання та методи демонстрування результатів навчання:

- поточні опитування;
- модульне тестування;
- презентації результатів виконання завдань та досліджень;
- оцінювання результатів виконання тренінгів;
- оцінювання результатів самостійної роботи;
- залік.

9. Критерії, форми поточного та підсумкового контролю

Підсумковий бал (за 100-бальною шкалою) з дисципліни «Micropython для одноплатних мікрокомп'ютерів» визначається як середньозважена величина, в залежності від питомої ваги кожної складової залікового кредиту:

7 семестр

Модуль 1		Модуль 2	Модуль 3
40%	40%	5%	15%
Поточне оцінювання	Модульний контроль 1	Тренінги	Самостійна робота
Середнє	Тестові завдання	Виконання 4 завдань	Виконання наскрізного

арифметичне за 4 лабораторні роботи			проекту із 3 завдань
--	--	--	----------------------

Оцінка за «Поточне оцінювання» визначається як середнє арифметичне оцінок, отриманих студентом при виконанні лабораторних завдань. Кожне завдання оцінюється за 100 бальною шкалою на основі якості виконання та відповідності поставленим завданням. Бали нараховуються відповідно до таких критеріїв:

- 15 балів – розуміння теоретичних відомостей (розкриття понятійного апарату відповідно до теми лабораторного заняття);
- 25 балів – правильність розв’язування (коректність реалізації, правильний вибір підходу);
- 10 балів – якість виконання (унікальність рішення, стиль програмування, якість програмного коду);
- 10 балів – результативність / тестування (програма видає коректний результат при тестуванні);
- 10 балів – аналіз і висновки (опис отриманих результатів, засоби їх досягнення);
- 30 балів – захист практичної роботи (відповідь на запитання викладача).

Модульний контроль проводиться у формі виконання тестових завдань з однією правильною відповіддю. Тестовий контроль складається з теоретичних питань і одного практичного завдання, що передбачає проведення практичних дій перед вибором варіанту відповіді. Частка виконання практичного завдання становить 40% (40 балів при правильній відповіді) у підсумку. Відповідно частка теоретичної частини складає 60% (60 балів). Максимальний бал за виконання теоретичних питань модульного контролю залежить від середнього арифметичного правильних відповідей (наприклад, якщо тестів 20 і студент відповів на 15 правильно, то результат рівний $15/20 = 0.75$, тобто 75 % від максимально можливого балу). Підсумкове значення виконання тестового контролю визначається сумою балів отриманих при виконанні тестових і практичних завдань.

Тренінг оцінюється з повнотою реалізації, якістю програмного коду, тестуванням та аналізом ефективності виконання завдань відповідно до п.7 даної робочої програми.

Оцінювання рівня виконання завдань тренінгу проводиться викладачем за 100-бальною шкалою згідно таблиці 1.

Таблиця 1 – Критерії оцінювання тренінгу

Завдання	Макс. бал	Високий рівень	Достатній рівень	Середній рівень	Низький рівень
1. Огляд сучасних одноплатних комп'ютерів	20	16-20 ґрунтовний огляд з наведеним порівняльним аналізом	11-15 достатньо аргументований, але без огляду аналогів	6-10 поверхневий огляд	0-5 відсутній або невідповідне
2. Аналіз та проектування апаратної системи	20	16-20 аргументоване, з порівняльним аналізом	11-15 достатньо аргументоване, але без альтернатив	6-10 поверхневий аналіз	0-5 відсутнє або невідповідне
3. Програмна реалізація проекрованої системи	30	21-30 реалізація у повній мірі	11-20 реалізація з незначними недоліками	6-10 передбачено лише часткові випадки	0-5 відсутнє або невідповідне
4. Тестування розробленої системи на основі Raspberry Pi	30	21-30 комплексне тестування	11-20 тестування, часткове	6-10 часткова відповідність, наявність помилок	0-5 відсутнє або невідповідне
Разом	100	-	-	-	-

Модуль «Самостійна робота» оцінюється викладачем за якістю виконаного індивідуального завдання та звіту відповідно до критеріїв, визначених у розділі 6 «Самостійна робота» цієї робочої програми. Підсумковий бал розраховується як сума оцінок за окремими критеріями у таблиці 2.

Таблиця 2 – Критерії оцінювання самостійної роботи

Завдання	Макс. бал	Високий рівень	Достатній рівень	Середній рівень	Низький рівень
Проектування структури та алгоритму роботи	30	22-30 алгоритм	15-21 алгоритм	10-14 поверховий	0-9 відсутнє або

одноплатного комп'ютера згідно індивідуального завдання		реалізовано у повній мірі	реалізований в неповній мірі	аналіз	невідповідне
Розробка програмної частини системи обробки даних на основі одноплатного комп'ютера	30	22-30 універсальний підхід	15-21 загальний, не універсальний підхід	10-14 типові рішення	0-9 відсутнє або невідповідне
Тестування програмної та апаратної складової системи обробки даних на основі одноплатного комп'ютера	40	26-40 комплексне тестування	17-25 тестування, часткове	10-14 часткова відповідність, наявність помилок	0-9 відсутнє або невідповідне
Разом	100	-	-	-	-

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.	PyCharm Community Edition	1-13
2.	Raspberry Pi OS	10-13
3.	Raspberry Pi	10-13

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

1. Al Sweigart Beyond the Basic Stuff with Python: Best Practices for Writing Clean Code, No Starch Press (December 16, 2020); eBook (Read Online). 2020. 384.
2. Tomas Beuzen, Tiffany Timbers Python Packages. Chapman and Hall/CRC; 1st edition (April 21, 2022); eBook (Open Source Book). 2022. 222.

3. Harry Percival, Bob Gregory Francois Chollet Deep Learning with Python, Second Edition. Manning; 2nd edition (December 21, 2021). 2021. 504.
4. Architecture Patterns with Python: Enabling Test-Driven Development, Domain-Driven Design, and Event-Driven Microservices. O'Reilly Media; 1st edition (2020); eBook (Creative Commons Licensed). 2020. 304.
5. Leonardo Giordani Clean Architectures in Python: A Practical Approach to Better Software Design. The Digital Cat Books/Leanpub (2022). 2022. 422.
6. Yasoob Khalid Practical Python Projects. Self-Publishing, 2022. 2022.
7. Lee Vaughan Impractical Python Projects: Playful Programming Activities to Make You Smarter Illustrated Edition. No Starch Press; Illustrated edition (November 27, 2018). 2018. 424.
8. . Al Sweigart The Big Book of Small Python Projects: 81 Easy Practice Programs No Starch Press (June 25, 2021); eBook (Read Online). 2021. 432.
9. Gareth Halfacre The Official Raspberry Pi Beginner's Guide Raspberry Pi Press (2019); eBook (4th Edition, 2020, Creative Commons Licensed). 2019. 241.
10. Rob Zwetsloot The Official Raspberry Pi Handbook Raspberry Pi Press (2022); eBook (Creative Commons Licensed). 2022. 241.
11. Основи програмування мовою Python: навчальний посібник. Ч.: ФОП Баликіна С.М., 2020. 180 с
12. Замуруєва О. В., Кримусь А. С., Ольхова Н. В. Об'єктно-орієнтоване програмування в Python : курс лекцій. Луцьк : Вежа-Друк, 2018. – 64 с.