

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
МІЖНАРОДНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ  
Кафедра інформаційних технологій

**Мірошник М.А., Горбачов В.Е., Григор'єва Т.І.**

## **ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ ТЕХНОЛОГІЇ**

---

Методичні рекомендації для самостійної роботи здобувачів

Одеса 2024

Затверджено Вченою Радою Міжнародного гуманітарного університету  
(протокол № 13 від 16.08.2024)

**Мірошник М.А., Горбачов В.Е., Григор'єва Т.І.**

Інформаційно-комунікаційні технології: методичні рекомендації для самостійної роботи здобувачів [Електронне видання]. / Мірошник М.А., Горбачов В.Е., Григор'єва Т.І. Кафедра інформаційних технологій Міжнародного гуманітарного університету. Одеса, 2024. – 25 с.

Методичні рекомендації з курсу «Інформаційно-комунікаційні технології» розроблено відповідно до навчального плану, вони складаються з навчальної програми курсу, методичних рекомендацій з проведення практичних занять і завдань для самостійної роботи здобувачів, списку рекомендованої літератури. Матеріали призначено для здобувачів другого (магістерського) рівня вищої освіти спеціальності 014.09 – «Середня освіта (Інформатика)».

Вивчення дисципліни «Інформаційно-комунікаційні технології» сприятиме залученню здобувачів до науково-дослідницької діяльності та підготовки ними публікацій, кваліфікаційних й інших наукових робіт.

## 1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
Кількість кредитів – 6, загальна кількість годин – 180	Галузь – 01 «Освіта/Педагогіка»  Спеціальність – 014.09 «Середня освіта (Інформатика)»	Денна форма	Заочна форма
		обов'язкова	
		Рік підготовки:	
		1	
Мова навчання – українська	Рівень вищої освіти – другий (магістерський)	1	
		Лекційні заняття:	
		28	6
		Практичні заняття:	
		18	6
		Самостійна робота:	
		134	168
		Вид контролю:	
залік			

У дисципліні «Інформаційно-комунікаційні технології» вивчаються питання розроблення стратегій зберігання, архівування та резервного копіювання даних IoT для довгострокового аналізу та звітності. Визначення стратегій масштабування та розширення IoT-інфраструктури для відповіді на зростання обсягу даних та потреб користувачів. Важливі для розгляду є такі аспекти IoT, як інфраструктури для збору, зберігання, обробки та аналізу даних, які надходять від сенсорів та пристроїв IoT. Ця інфраструктура може включати в себе хмарні платформи, централізовані сервери, бази даних та інші технології.

**Мета дисципліни** «Інформаційно-комунікаційні технології» – формування компетентностей щодо застосування основ побудови індустріальних рішень за

методологіями IoT, поглиблення знань з мережних технологій з точки зору їх застосування в області Інтернету речей та використання широкого спектру апаратно-програмних засобів збирання, передавання та аналізу даних.

## **ЗАПЛАНОВАНІ РЕЗУЛЬТАТИ НАВЧАННЯ ЗА НАВЧАЛЬНОЮ ДИСЦИПЛІНОЮ**

### **Знання:**

- концептуальні та теоретичні знання у сфері інженерії програмного забезпечення;
- методологічні знання в плані застосування сучасних методів та технологій для розробки програмного забезпечення;
- мережні технології і «Інтернет Речей»;
- програмне забезпечення для реалізації алгоритмів IoT;
- вимоги по оформленню результатів наукових досліджень;
- методи наукового дослідження;
- методи моделювання; математичне моделювання;
- аналіз та синтез в процедурах моделювання.

### **Вміння:**

- розробляти програмне забезпечення;
- використовувати підходи адаптації програмного забезпечення до змін;
- застосовувати сучасні інструменти для тестування програмного забезпечення;
- ефективно взаємодіяти з командою при груповій розробці програмного забезпечення;
- виконувати інформаційний пошук, накопичування та обробляти наукову інформацію;
- застосовувати на практиці сучасні прийоми та методи наукових досліджень;
- знаходити точний або наближений розв'язок математичної задачі;

- досліджувати нестандартні задачі класифікації, аналізу складних систем процесів;
- використовувати математичні методи у наукових дослідженнях;
- написати наукову статтю, тези доповідей конференції, науковий звіт.

## **2. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

### **Тема 1. Вступ до дисципліни**

Інтернету речей (IoT): основні терміни і поняття. IoT у сучасному світі. Загальні принципи побудови та архітектура IoT. Класифікація систем IoT. Складові блоки системи IoT.

### **Тема 2. Процеси в керованих системах**

Склад і структура системи керування. Основні елементи та порядок функціонування системи керування. Моделі комунікації в IoT. Комунікації в різних системах IoT. Еталонна модель IoT. Інструментальні засоби моделювання систем «Інтернету Речей». Моделювання рішень Інтернету речей в рамках сценаріїв Цифровізації. Інструментарій додатку Cisco Packet Tracer для інтернету речей. Створення «розумних речей» засобами Packet Tracer.

### **Тема 3. Мережні технології і «Інтернет Речей»**

Огляд основних стандартних протоколів Інтернету, що застосовуються в IoT. Пристрої мережевого з'єднання та маршрутизація. Протоколи IoT. Підключення речей до мережі. Хмарні обчислення в IoT. Зв'язок даних та мережеві з'єднання, з'єднання додатків. Технічні засоби в IoT. Загальні відомості про датчики. Основні характеристики датчиків. Класифікація датчиків. Технологія MEMS (MicroElectro-Mechanical Systems). Актуатори. Мікроконтролери. Способи підключення датчиків і актуаторів до мікроконтролерів. Різниця між мікропроцесорами, мікроконтролерами і мікрокомп'ютерами.

### **Тема 4. Програмне забезпечення для реалізації алгоритмів IoT**

Основні концепції програмування. Системне ПЗ, прикладні програми, мови програмування. Базові структури програм. Одноплатний комп'ютер (SBC) Raspberry Pi. Використання операційної системи Linux. Програмування за допомогою Python.

Бездротові технології в IoT. Бездротові технології WiFi, ZigBee, Bluetooth, 4G/5G, та LoRaWAN. Безпека бездротових мереж.

### 3. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Назви змістових модулів і тем	Кількість годин							
	Денна форма				Заочна форма			
	Загалом	У тому числі			Загалом	У тому числі		
		Лекц.	Практ.	Сам.		Лекц.	Практ.	Сам.
Тема 1. Вступ до дисципліни	40	6	4	30	40	2	0	38
Тема 2. Процеси в керованих системах	40	6	4	30	40	2	2	36
Тема 3. Мережні технології і «Інтернет Речей»	40	8	4	28	40	0	2	38
Тема 4. Програмне забезпечення для реалізації алгоритмів IoT	60	8	6	46	60	2	2	56
<b>Загалом</b>	<b>180</b>	<b>28</b>	<b>18</b>	<b>134</b>	<b>180</b>	<b>6</b>	<b>6</b>	<b>168</b>
<b>Підсумковий контроль – залік</b>								

### 4. ПИТАННЯ ДО ПРАКТИЧНИХ ЗАНЯТЬ

#### Тема 1. Вступ до дисципліни

Інтернету речей (IoT): основні терміни і поняття. IoT у сучасному світі. Загальні принципи побудови та архітектура IoT. Класифікація систем IoT. Складові блоки системи IoT.

#### Тема 2. Процеси в керованих системах

Склад і структура системи керування. Основні елементи та порядок функціонування системи керування. Моделі комунікації в IoT. Комунікації в різних системах IoT. Еталонна модель IoT. Інструментальні засоби моделювання систем «Інтернету Речей».

Моделювання рішень Інтернету речей в рамках сценаріїв Цифровізації. Інструментарій додатку Cisco Packet Tracer для інтернету речей. Створення «розумних речей» засобами Packet Tracer.

### **Тема 3. Мережні технології і «Інтернет Речей»**

Огляд основних стандартних протоколів Інтернету, що застосовуються в IoT. Пристрої мережевого з'єднання та маршрутизація. Протоколи IoT. Підключення речей до мережі. Хмарні обчислення в IoT. Зв'язок даних та мережеві з'єднання, з'єднання додатків. Технічні засоби в IoT. Загальні відомості про датчики. Основні характеристики датчиків. Класифікація датчиків. Технологія MEMS (MicroElectro-Mechanical Systems). Актуатори. Мікроконтролери. Способи підключення датчиків і актуаторів до мікроконтролерів. Різниця між мікропроцесорами, мікроконтролерами і мікрокомп'ютерами.

### **Тема 4. Програмне забезпечення для реалізації алгоритмів IoT**

Основні концепції програмування. Системне ПЗ, прикладні програми, мови програмування. Базові структури програм. Одноплатний комп'ютер (SBC) Raspberry Pi. Використання операційної системи Linux. Програмування за допомогою Python. Бездротові технології в IoT. Бездротові технології WiFi, ZigBee, Bluetooth, 4G/5G, та LoRaWAN. Безпека бездротових мереж. Розроблення інструментів для моніторингу та аналізу продуктивності та стану IoT-пристроїв для підтримки їхньої ефективності та покращення роботи. Як ви зберігаєте та архівуєте дані IoT для подальшого використання та аналізу? Розроблення стратегій зберігання, архівування та резервного копіювання даних IoT для довгострокового аналізу та звітності. Як ви плануєте розширювати та масштабувати вашу IoT-систему в майбутньому? Визначення стратегій масштабування та розширення IoT-інфраструктури для відповіді на зростання обсягу даних та потреб користувачів.

## **5. САМОСТІЙНА РОБОТА**

1. Знайомство з науковою та навчальною літературою відповідно зазначених у програмі тем.
2. Опрацювання лекційного матеріалу.
3. Підготовка до практичних занять.
4. Консультації з викладачем протягом семестру.
5. Самостійне опрацювання окремих питань навчальної дисципліни.
6. Підготовка до підсумкового контролю.

**Тематика та питання до самостійної підготовки та індивідуальних завдань**

### **Тема 1. Вступ до дисципліни**

Інтернету речей (IoT): основні терміни і поняття. IoT у сучасному світі. Загальні

принцип побудови та архітектура IoT. Класифікація систем IoT. Складові блоки системи IoT.

## **Тема 2. Процеси в керованих системах**

Як визначити ціль та завдання керованої системи? Які датчики та сенсори використовуються для моніторингу системи? Як здійснюється моніторинг та аналіз даних для виявлення аномалій та прийняття рішень? Як можна оптимізувати роботу системи для досягнення кращої продуктивності та ресурсощадності? Як здійснюється аналіз ефективності та вдосконалення системи з часом? Як взаємодіють різні компоненти системи між собою? Моделювання рішень Інтернету речей в рамках сценаріїв Цифровізації. Інструментарій додатку Cisco Packet Tracer для інтернету речей. Створення «розумних речей» засобами Packet Tracer.

## **Тема 3. Мережні технології і «Інтернет Речей»**

Які мережні технології використовуються для забезпечення зв'язку в "Інтернеті Речей"? Як забезпечити безпеку в мережах IoT? Як здійснюється збір, обробка та аналіз даних, зібраних від пристроїв IoT? Огляд основних стандартних протоколів Інтернету, що застосовуються в IoT. Протоколи IoT. Хмарні обчислення в IoT. Як ви збираєте та аналізуєте дані про продуктивність і роботу IoT-пристроїв?

## **Тема 4. Програмне забезпечення для реалізації алгоритмів IoT**

Встановлення механізмів комунікації з IoT-пристроями, включаючи використання протоколів зв'язку (наприклад, MQTT, CoAP), підтримку мережевих стандартів (наприклад, Wi-Fi, Bluetooth, LoRa), а також заходи забезпечення безпеки комунікацій. Які алгоритми і логіка ви хочете вбудувати в ваші IoT-пристрої? Визначення конкретних завдань та алгоритмів, які повинні бути реалізовані на пристроях IoT, такі як збір даних, обробка, аналіз та прийняття рішень. Які дані збираються і передаються від сенсорів до центральної системи? Визначення типів даних, які збираються від сенсорів (температура, вологість, рух тощо) та механізмів їх передачі. Як забезпечити безпеку та конфіденційність даних IoT? Розроблення



стратегій шифрування, автентифікації та захисту від несанкціонованого доступу до даних, переданих пристроями IoT. Як здійснюється збір і аналіз даних IoT? Розроблення алгоритмів та інфраструктури для збору, зберігання і аналізу великих обсягів даних, зібраних від IoT-пристроїв. Як забезпечити відмовостійкість і надійність IoT-системи? Розроблення стратегій резервного копіювання, відновлення та моніторингу для підтримки надійності та доступності IoT-пристроїв.

## **Теми доповідей**

### **1. Основи програмування для IoT: огляд мов та інструментів**

Огляд найпопулярніших мов програмування для Інтернет речей (Python, C/C++, Java). Вибір мови програмування в залежності від типу пристрою і його обертальних

### **2. Системне програмне забезпечення для IoT: FreeRTOS, Contiki та інші**

Особливості операційних систем для пристроїв IoT. Порівняння популярних систем: FreeRTOS, Contiki OS, Zephyr

### **3. Raspberry Pi як платформа для реалізації IoT-проектів**

Можливості одноплатних комп'ютерів. Приклади використання Raspberry Pi в системах

### **4. Linux для IoT: можливості, переваги та налаштування**

Використання Linux в розробці IoT-р. Налаштування та управління сенсорами й актуаторами на базі Linux.

### **5. Програмування для IoT на Python: приклади і реалізація**

Програмування пристроїв IoT. Огляд бібліотек для роботи з сенсорами, актуаторами та мережею.

### **6. Бездротові технології для IoT: WiFi, ZigBee, Bluetooth, LoRaWAN**

Особливості та порівняння різних бездротових технологій.

### **7. LoRaWAN як ефективна технологія для IoT: переваги і недоліки**

Переваги використання LoRaWAN. Приклади застосування.

### **8. 4G/5G в IoT: роль мобільних мереж у підключених пристроях**

Використання мобільних мереж для підключення IoT. Переваги 5G для високошвидкісного обміну даними в IoT.

### **9. Моделювання та симуляція IoT-системи за допомогою Packet Tracer**

Використовуйте Cisco Packet Tracer для моделювання IoT-інфраструктури. Приклади симуляції IoT-сценаріїв для навчальних і професійних цілей.

### **10. Огляд основних стандартних протоколів Інтернету, що застосовуються в IoT**

HTTP, MQTT, CoAP: порівняння протоколів передачі даних в IoT. Особливості використання протоколів.

### **11. Пристрої мережевого з'єднання та маршрутизація в системах IoT**

Основні маршрутизатори та шлюзи для підключення пристроїв IoT. Особливості маршрутизації даних у великих IoT-мережах.

### **12. Протоколи IoT: від HTTP до LoRaWAN**

Огляд популярних протоколів IoT: особливості використання та приклади використання. Порівняння низькоенергетичних протоколів передачі даних.

### **13. Підключення «речей» до мережі: особливості та виклики**

Проблеми підключення великої кількості IoT-пристроїв. Технології підключення з низьким енергоспоживанням.

### **14. Хмарні обчислення в IoT: архітектура та переваги**

Як хмарні обчислення інтегруються в IoT-систему. Використання.

### **15. Технічні засоби IoT: сенсори, актуатори та мікроконтролери**

Основні компоненти IoT-системи: сенсори, актуатори, мікроконтролери. Приклади їх використання в різних галузях (розумний дім, промисловість, медицина).

### **16. Загальні відомості про датчики в системах IoT**

Види датчиків, що використовують в IoT: температуру, вологість, тиск тощо. Особливості вибору датчика для конкретних IoT-проектів.

### **17. Основні характеристики датчиків для IoT-проектів**

Ключові параметри датчиків: чутливість, роздільність, точність. Як ці характеристики впливають на ефективність IoT-системи.

## **18.Класифікація датчиків у системах Інтернету речей**

Механічні, електричні, оптичні та інші типи датчиків. Приклади застосування датчиків у різних галузях промисловості (логістика, сільське господарство, медицина).

## **19.Технологія MEMS (MicroElectro-Mechanical Systems) в IoT**

Принципи роботи MEMS-датчиків. Приклади використання MEMS в сучасних пристроях IoT.

## **20.Зв'язок даних та мережеві з'єднання в IoT: від сенсорів до хмари**

Як отримати обмін даними між IoT-пристроями і хмарними платформами. Протоколи та технології, що забезпечують надійний зв'язок.

## **21.Безпека мережі IoT: проблеми і рішення**

Основні загрози для безпеки IoT-мереж. Методи захисту.

## 6. ВИДИ ТА МЕТОДИ КОНТРОЛЮ

Робоча програма навчальної дисципліни передбачає наступні види та методи контролю:

Види контролю	Складові оцінювання
<b>поточний контроль</b> , який здійснюється у ході: проведення практичних занять, виконання індивідуального завдання; проведення консультацій та відпрацювань.	<b>50%</b>
<b>підсумковий контроль</b> , який здійснюється у ході проведення іспиту.	<b>50%</b>

<b>Методи діагностики знань (контролю)</b>	фронтальне опитування; наукова доповідь, усне повідомлення, індивідуальне опитування, робота у групах, розв'язання задач і практичних завдань, іспит
--	--

## 7. Питання для підсумкового контролю

1. Інтернету речей (IoT) у сучасному світі. Загальні принцип побудови та архітектура IoT.
2. Класифікація систем IoT. Складові блоки системи IoT.
3. Склад і структура системи керування.
4. Основні елементи та порядок функціонування системи керування.
5. Моделі комунікації в IoT.
6. Комунікації в різних системах IoT. Еталонна модель IoT.
7. Інструментальні засоби моделювання систем «Інтернету Речей»
8. Моделювання рішень Інтернету речей в рамках сценаріїв цифровізації
9. Інструментарій додатку Cisco Packet Tracer для інтернету речей
10. Створення «розумних речей» засобами Packet Tracer
11. Мережні технології і «Інтернет Речей»
12. Огляд основних стандартних протоколів Інтернету, що застосовуються в IoT
13. Пристрої мережевого з'єднання та маршрутизація
14. Протоколи IoT
15. Підключення речей до мережі. Хмарні обчислення в IoT.
16. Зв'язок даних та мережеві з'єднання, з'єднання додатків.
17. Технічні засоби в IoT. Загальні відомості про датчики. Основні характеристики датчиків. Класифікація датчиків. Технологія MEMS (MicroElectro-

Mechanical Systems).

18. Актуатори. Мікроконтролери. Способи підключення датчиків і актуаторів до мікроконтролерів.

19. Різниця між мікропроцесорами, мікроконтролерами і мікрокомп'ютерами

20. Програмне забезпечення для реалізації алгоритмів IoT

21. Основні концепції програмування. Системне ПЗ, прикладні програми, мови програмування. Базові структури програм

22. Одноплатний комп'ютер (SBC) Raspberry Pi

23. Використання операційної системи Linux

24. Програмування за допомогою Python

25. Бездротові технології в IoT. Бездротові технології WiFi, ZigBee, Bluetooth, 4G/5G, та LoRaWAN.

26. Безпека бездротових мереж

27. Методи і моделі розроблення, впровадження, супроводу програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу.

28. Моделі інформаційних процесів у прикладній області.

29. Інформаційні потреби і дані для проектування програмного забезпечення.

30. Аналіз і синтез при застосовуванні на системному рівні сучасних програмних та апаратних платформ для розв'язання складних задач інженерії програмного забезпечення.

31. Розробка архітектури програмного забезпечення для реалізації вимог замовника.

32. Модифікація існуючих та розробка нових алгоритмічних рішень детального проектування програмного забезпечення.

## **8. КРИТЕРІЇ ПІДСУМКОВОЇ ОЦІНКИ ЗНАТЬ СТУДЕНТІВ**

Рівень знань оцінюється:

- «відмінно» / «зараховано» А - від 90 до 100 балів. Студент виявляє особливі творчі здібності, вміє самостійно знаходити та опрацьовувати необхідну інформацію, демонструє знання матеріалу, проводить узагальнення і висновки. Був присутній на лекціях та семінарських заняттях, під час яких давав вичерпні, обґрунтовані, теоретично і практично правильні відповіді, має конспект з виконаними завданнями до самостійної роботи, проявляє активність і творчість у науково-дослідній роботі;

- «добре» / «зараховано» В - від 82 до 89 балів. Студент володіє знаннями матеріалу, але допускає незначні помилки у формуванні термінів, категорій, проте за допомогою викладача швидко орієнтується і знаходить правильні відповіді. Був присутній на лекціях та семінарських заняттях, має конспект з виконаними завданнями до самостійної роботи, проявляє активність і творчість у науково-дослідній роботі;

- «добре» / «зараховано» С - від 74 до 81 балів. Студент відтворює значну частину теоретичного матеріалу, виявляє знання і розуміння основних положень, з

допомогою викладача може аналізувати навчальний матеріал, але дає недостатньо обґрунтовані, невичерпні відповіді, допускає помилки. При цьому враховується наявність конспекту з виконаними завданнями до самостійної роботи та активність у науково-дослідній роботі;

- «задовільно» / «зараховано» D - від 64 до 73 балів. Студент був присутній не на всіх лекціях та семінарських заняттях, володіє навчальним матеріалом на середньому рівні, допускає помилки, серед яких є значна кількість суттєвих. При цьому враховується наявність конспекту з виконаними завданнями до самостійної роботи;

- «задовільно» / «зараховано» E - від 60 до 63 балів. Студент був присутній не на всіх лекціях та семінарських заняттях, володіє навчальним матеріалом на рівні, вищому за початковий, значну частину його відтворює на репродуктивному рівні, на всі запитання дає необґрунтовані, невичерпні відповіді, допускає помилки, має неповний конспект з завданнями до самостійної роботи.

- «незадовільно з можливістю повторного складання» / «не зараховано» Fx – від 35 до 59 балів. Студент володіє матеріалом на рівні окремих фрагментів, що становлять незначну частину навчального матеріалу.

- «незадовільно з обов'язковим повторним вивченням дисципліни» / «не зараховано» F – від 1 до 34 балів. Студент не володіє навчальним матеріалом.

#### Таблиця відповідності результатів контролю знань за різними шкалами

100-бальною шкалою	Шкала за ECTS	За національною шкалою	
		екзамен	залік
90-100 (10-12)	A	Відмінно	зараховано
82-89 ( 8-9)	B	Добре	
74-81(6-7)	C		
64-73 (5)	D	Задовільно	не зараховано
60-63 (4)	E		
35-59 (3)	Fx	незадовільно	
1-34 (2)	F		

## 9. ОПОРНИЙ КОНСПЕКТ ЛЕКЦІЙ

### Інтернет речей (IoT). Основні терміни і поняття

Інтернет речей (IoT) — це мережа фізичних пристроїв (речей), що оснащена датчиками, програмним забезпеченням та іншими технологіями для підключення та обміну даними з іншими пристроями і системами через Інтернет.

Мета IoT — забезпечити автоматизацію, моніторинг і управління безперебійними системами в реальному часі.

Речі (Things) — це фізичні об'єкти, які можна підключати до мережі для збору, передачі або прийому даних. Це можуть бути датчики, пристрої, машини, транспортні засоби, побутова техніка тощо.

Мережа (Network) — інфраструктура для передачі даних між пристроями IoT. деякі технології технології Wi-Fi, Bluetooth, 4G/5G, LoRa, ZigBee тощо.

Дані (Data) — інформація, зібрана з пристроїв, що надходять до хмарних або локальних систем для обробки, аналізу та прийняття рішень.

Хмара (Cloud) — це інфраструктура для зберігання та обробки даних, яка дозволяє об'єднувати й аналізувати інформацію з різних пристроїв.

Шлюз (Gateway) — це пристрій, який виступає посередником між локальними пристроями IoT та Інтернетом, забезпечуючи зв'язок та обмін даними.

Протокол передачі даних — набір правил, які використовують для передачі інформації між пристроями. Найбільш популярні протоколи в IoT — MQTT, CoAP, HTTP.

### IoT в сучасному світі

IoT стає ключовою технологією в багатьох галузях, таких як:

- Розумні будинки (Smart Homes): Автоматизація та управління побутовими приладами, освітленням, безпекою, клімат-контролем.
- Індустрія 4.0 (Smart Manufacturing): Виробництво з підключеними машинами, що сприяє оптимізації цією процесів та підвищенню ефективності.
- Розумне місто (Smart Cities): Інтелектуальне управління транспортом, енергоспоживанням, безпекою та екологією.
- Охорона здоров'я (Healthcare IoT): Пристрої для моніторингу здоров'я населення, носіїв пристроїв, телемедицини.

### Загальні принципи побудови та архітектури IoT

Архітектура IoT складається з кількох основних рівнів:

1. Фізичний рівень (пристрої). Це сенсори, актори (пристрої, які використовують дію), обладнання для збору даних.
2. Мережевий рівень. Цей рівень відповідає передачі даних із пристроїв до хмарних серверів або локальних баз даних. Включає в себе шлюзи та мережеві протоколи.

3. Рівень управління та обробки даних. Включає системи збору, обробки, зберігання та аналізу даних. Використовуються хмарні обчислення або локальні сервери.

4. Рівень додатків. Це послуги та додатки, які використовуються зібраними даними для управління або аналітики. Наприклад, програми для управління розумним будинком або системи управління виробництвом.

### Класифікація систем IoT

За типом середовища:

- Промислові IoT. Використовуються в промислових додатках, таких як контроль виробництва, моніторинг машини, безпека.
- Споживчі IoT. Застосовуються в побутових пристроях, таких як розумні будинки, носяться пристрої.
- Міські IoT. Використовуються для інтелектуального управління міською інфраструктурою.

За масштабом:

- Локальні системи IoT. Працюють у межах однієї будівлі або об'єкта.
- Глобальні системи IoT. Охоплюють великі регіони або навіть цільову країну.

За призначенням:

- Промислові IoT. орієнтовані на автоматизацію виробничих процесів.
- Споживчі IoT. Розумні прилади, пристрої для дому та особистого користування.

### Процеси в керованих системах

Система керування — це сукупність взаємодіючих елементів, призначених для керування станами іншої системи (об'єкта) на основі зібраних даних і прийнятих рішень.

У контексті IoT керовані системи забезпечують автоматизацію процесів шляхом обміну даними між «розумними» засобами.

### Склад і структура системи керування

Типова система керування включає наступні елементи:

1. Датчики (сенсори). Вимірюють параметри навколишнього середовища або стану об'єкта (температура, тиск, рівень світла тощо).



2. Контролери. Обробляють сигнали з датчиків, приймають рішення на основі запрограмованих алгоритмів.

3. Актуатори (приводи). Виконують дії на основі сигналів від контролерів (відкриття дверей, увімкнення освітлення тощо).

4. Інтерфейси зв'язку. Забезпечують передачу даних між ефективними елементами системи (наприклад, за протоколами Wi-Fi, Bluetooth, ZigBee).

5. Інформаційна система (обчислювальна платформа). Виконує обробку та аналіз зібраних даних, які можуть бути реалізовані у вигляді хмарних сервісів або локальних серверів.

Основні елементи та порядок функціонування системи керування:

1. Датчики збирають дані з об'єкта або середовища.
2. Контролери обробляють дані й приймають рішення про подальші дії.
3. Актуатори забезпечують дії, вносячи зміни в стан об'єкта або системи.
4. Передача даних між датчиками, контролерами та гравцями виконує мережеві протоколи, забезпечуючи безперервну комунікацію.
5. Моніторинг та контроль вивчаються за допомогою програмного забезпечення або інтерфейсів користувача.

### Моделі комунікації в IoT

Комунікації в IoT розглядаються через моделі взаємодії між пристроями:

1. Модель "пристрій-пристрій" (D2D). Пристрої обміну обмінюються даними разом, наприклад, через Bluetooth або ZigBee.
2. Модель "пристрій-хмара" (Device-to-Cloud). Пристрої надсилають дані до хмарних серверів, де ці дані обробляються та аналізуються.
3. Модель "пристрій-шлюз" (Device-to-Gateway). Пристрої передають дані через шлюз (наприклад, локальний сервер) для їх подальшої передачі в хмару або іншу обчислювальну систему.

### Комунікації в різних системах IoT

1. Промислові IoT (IIoT). Використовуються високонадійні та енергоефективні мережі (LoRa, NB-IoT), щоб забезпечити збір даних з великої кількості датчиків на виробництві.

2. Розумні будинки. використання мережі локальних мереж (Wi-Fi, Z-Wave, ZigBee) для обміну інформацією між побутовими пристроями.

3. Медичні системи IoT. Використовуються мережі з низьким енергоспоживанням і високою надійністю, наприклад Bluetooth LE, щоб підключати носії пристроїв для моніторингу здоров'я.

### Еталонна модель IoT

Еталонна модель IoT описує всі додаткові рівні та функціональні компоненти системи IoT. Структура складається з таких рівнів:

1. Фізичний рівень (Пристрої). Датчики, актуатори та інші пристрої, які збирають дані.

2. Мережевий рівень. Відповідає за комунікації між пристроями та сервером.

3. Рівень обробки даних. Містить алгоритми для обробки та аналізу даних у реальному часі.

4. Рівень управління. Контролює роботу всієї системи, приймаючи рішення на основі оброблених даних.

5. Рівень додатків. Відповідає за реалізацію конкретних функцій та інтерфейсів для взаємодії з користувачами.

### Інструментальні засоби моделювання системи «Інтернет Речей»

Cisco Packet Tracer: Використовується для моделювання та симуляції системи IoT. Це програмне забезпечення дозволяє створювати та аналізувати мережі, підключати інші пристрої, а також виконувати симуляцію роботи «розумних» пристроїв.

### Мережні технології і «Інтернет Речей» (IoT)

Інтернет Речей (IoT) — це мережа фізичних пристроїв, які об'єднані між собою та з Інтернетом для обміну даними. IoT дозволяє автоматизувати та оптимізувати різні процеси в промисловості, сільському господарстві, медицині, транспорті та інших галузях.

Огляд основних стандартних протоколів Інтернету, що застосовуються в IoT:

1. IP (Internet Protocol). Основний протокол для передачі даних у мережі Інтернет. Використовується для маршрутизації даних між пристроями.
2. IPv6. Нова версія IP-протоколу, яка забезпечує більшу кількість IP-адрес, деяких для мільярдів IoT-пристроїв.

3. TCP (Transmission Control Protocol). Протокол, який забезпечує надійну передачу даних між пристроями.
4. UDP (протокол дейтаграм користувача). Легкий, але менш надійний протокол передачі даних, який часто використовується для швидкої передачі інформації з меншою затримкою, але без перевірки на доставку (корисно для деяких IoT-застосунків, наприклад, для сенсорів).
5. CoAP (Constrained Application Protocol). Протокол прикладного рівня для пристроїв з обмеженими ресурсами, який використовує IoT. Працює на основі UDP і використовується для комунікації між пристроями IoT.
6. MQTT (Message Queuing Telemetry Transport). Легкий протокол для обміну повідомленнями, який працює на основі моделі «видавець-підписник». Популярний в IoT для обміну даними між сенсорами та хмарними платформами.
7. HTTP/HTTPS. Традиційний протокол для передачі даних в Інтернеті. Використовується для підключення IoT-пристроїв до веб-серверів або хмарних сервісів.

### Пристрої мережевого з'єднання та маршрутизація

IoT-пристрої можуть підключатися до мережі через:

- Шлюзи. Забезпечують зв'язок між пристроями IoT та зовнішніми мережами, такими як Інтернет або локальні мережі.
- Маршрутизатори. Забезпечують маршрутизацію трафіку між широкими мережами та пристроями.
- Комутатори. Використовуються для локального підключення пристроїв IoT, створюючи внутрішню мережу.

### Протоколи IoT

1. ZigBee. Протокол з низьким енергоспоживанням, використання для підключення сенсорів і пристроїв у «розумних будинках» та інших малопотужних мережах.
2. Bluetooth Low Energy (BLE). Забезпечує низьку потужність і обмежену дальність дії, підходить для носіїв пристроїв і малих сенсорних мереж.
3. LoRaWAN. Використовується для мережі широкого охоплення (LPWAN), де потрібно передавати дані на великій відстані з низьким енергоспоживанням, наприклад, у сільському господарстві.
4. NB-IoT (вузькосмуговий IoT). Стандарт, що використовує мережеві рішення з низькою швидкістю передачі даних та енергоефективністю в сфері IoT.

### Підключення речей до мережі

Речі в IoT підключаються до мережі через різні способи комунікації, такі як Wi-Fi, Ethernet, ZigBee, LoRa або мобільні мережі 4G/5G. Вибір залежить від вимог до потужності, дальності та швидкості передачі даних.

## Хмарні обчислення в IoT

Хмарні обчислення забезпечують обробку і зберігання великих обсягів даних, які генеруються IoT-пристроями. Основні хмарні платформи (AWS IoT, Microsoft Azure IoT, Google Cloud IoT) забезпечують:

- Збір даних від пристроїв.
- Зберігання та обробку даних.
- Аналітика і візуалізація даних.
- Можливість управління пристроями на відстані.

### Зв'язок даних та мережеві з'єднання, з'єднання додатків

Дані від IoT-пристроїв передаються через різні мережеві протоколи (CoAP, MQTT, HTTP) до хмарних сервісів або інших візитів для подальшої обробки та адаптації.

## Технічні засоби в IoT

IoT-платформи вибір різноманітних технічних засобів, включаючи:

- Датчики. Пристрої, які вимірюють фізичні параметри (температура, вологість, тиск).
- Актуатори. Пристрої, які забезпечують фізичні дії на основі команди від контролера (наприклад, відкриття дверей).
- Мікроконтролери. Малі обчислювальні пристрої, які керують сенсорами і актуаторами.

### Загальні відомості про датчики

Датчики — це пристрої, які вимірюють фізичні параметри і перетворюють їх у сигнали, які передані далі для обробки.

Основні характеристики датчиків:

- Чутливість. Здатність датчика виявляти зміни вимірювального параметра.
- Точність. Ступінь кількості результату вимірювання до істинного значення.
- Діапазон вимірювань. Інтервал значень, в якому датчик може працювати.
- Швидкість реагування. Час, необхідний для реакції датчика на зміну параметрів.

### Класифікація датчиків

- Датчики температури (термостати, пари).
- Датчики тиску.
- Оптичні датчики (фотодіоди, фототранзистори).
- Магнітні датчики (магнетометри).
- Акселерометри та гіроскопи .

## Технологія MEMS (MicroElectro-Mechanical Systems)

MEMS — це технологія мікроелектромеханічних систем, що дозволяє створювати мікроскопічні датчики та актуатори на основі кремнію, які використовуються в мобільних пристроях, автомобільних системах, медичних пристроях та інших IoT-застосунках.

Актуатори — це пристрої, що перетворюють електричну енергію на фізичну дію (рух, нагрівання, відкриття або закриття). Вони підтримують команди від контролерів і забезпечують механічні дії.

Мікроконтролери — це невеликі обчислювальні пристрої, які керують датчиками та актуаторами. Вони містять процесор, пам'ять і порти для підключення зовнішніх пристроїв. Приклад популярного мікроконтролера — Arduino, який широко використовується в IoT-проектах.

### Способи підключення датчиків і актуаторів до мікроконтролерів

Датчики і актуатори можуть бути підключені до мікроконтролерів за допомогою:

- Цифрових входів/виходів.
- Аналогових входів/виходів.
- Комунікаційних шин (I2C, SPI, UART).

Програмне забезпечення для реалізації алгоритмів IoT охоплює широкий спектр технологій та інструментів, які забезпечують ефективну роботу та безпеку пристроїв, а також можливість інтеграції з хмарними рішеннями для аналітики.

## 10. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

### Основна

1. Wiring the IoT, Connecting Hardware with Raspberry Pi, Node-Red, and MQTT by Dr. Lucy Rogers, Dr. Andy Stanford-Clark, 2017
2. Simone Cirani, Gianluigi Ferrari, Marco Picone, Luca Veltri. Internet of Things. Architectures, Protocols and Standards. – Wiley. – 2019. – 394 p.
3. Мірошник М. А. Автоматизоване проектування програмних засобів систем: Методичні вказівки призначено для студентів напряму 123 «Комп'ютерна інженерія» Харків: УкрДУЗТ, 2020. 59 с.
4. Мірошник М. А. Теорія автоматичного управління, штучний інтелект і автоматизація процесу прийняття рішення: навч. посіб. Харків: УкрДУЗТ, 2018. 144 с.
5. Коцовський В.М. Супровід програмних систем: Методичний посібник для студентів спеціальності "Інженерія програмного забезпечення" / В. М. Коцовський. — Ужгород: Видавництво УжНУ "Говерла", 2016. — 52 с.
6. Петрик М.Р., Петрик О.Ю. Моделювання програмного забезпечення: науково-методичний посібник. Тернопіль: Вид-во ТНТУ імені Івана Пулюя, 2015. 200 с.
7. Мірошник М. А. Інформаційноуправляючі системи та організація паралельних обчислювань : навч. посіб. Харків : «Діса плюс», 2015. 324 с
8. Троцько В.В. Методи штучного інтелекту : навч.-методич. і практ. посіб. / за рец. А.М. Котенка, Ю.Є. Добришина. Київ: Університет економіки та права «КРОК», 2020 86 с. URL: [https://library.krok.edu.ua/media/library/category/navchalni-posibniki/trotsko\\_0001.pdf](https://library.krok.edu.ua/media/library/category/navchalni-posibniki/trotsko_0001.pdf)
9. Гороховатський В.О., Творошенко І.С. Методи інтелектуального аналізу та оброблення даних: навч. посіб. / за рец. В.О. Філатова, О.А. Винокурової. Харків: ХНУРЕ, 2021. 92 с. URL: <https://openarchive.nure.ua/server/api/core/bitstreams/2e55d639-52fd-48d9-b7b7-14989f49f291/content>
10. Болюбаш Н.М. Методичні вказівки до виконання курсової роботи з дисципліни «Інтелектуальний аналіз» даних для студентів спеціальності 122 – Комп'ютерні науки : методичні вказівки / за рец. І.П. Атаманюка. Миколаїв : Вид-в ЧНУ ім. Петра Могили, 2021. 28 с. URL: <https://bit.ly/40sKIGw>

## Допоміжна

1. Павленко П.М. Основи математичного моделювання систем і процесів: Навчальний посібник. – К.: Книжкове вид-во НАУ, 2010.– 198 с.
2. Маценко В.Г. Математичне моделювання: навчальний посібник– Чернівці: Чернівецький національний університет, 2014.–519 с.
3. Пасічник В.В., Виклюк Я.І., Камінський Р.М. Моделювання складних систем. Посібник. Львів: Видавництво "Новий Світ - 2000". 2017. 404с.
4. Raschka S., Liu Y., Mirjalili V. Machine learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python. Birmingham, UK: Packt Publishing Ltd., 2022. 770 p. URL: <https://github.com/rasbt/machine-learning-book>
5. Banachewicz K., Massaron L. The Kaggle book: Data analysis and machine learning for competitive data science. Birmingham, UK: Packt Publishing Ltd., 2022. 531 p. URL: <https://github.com/PacktPublishing/The-Kaggle-Book>
6. Олещенко Л.М. Машинне навчання: Комп'ютерний практикум з дисципліни «Машинне навчання» [Електронний ресурс]: навч. посіб. для студ. спеціальності 121 «Інженерія програмного забезпечення» / за рец. С.М. Чумаченка, А.О. Мошенського. Київ: КПІ ім. Ігоря Сікорського, 2022. 92 с. URL: [https://ela.kpi.ua/bitstream/123456789/48804/1/Mashynne\\_navchannia.pdf](https://ela.kpi.ua/bitstream/123456789/48804/1/Mashynne_navchannia.pdf)

## Інформаційні ресурси

1. Офіційний сайт Python [Електронний ресурс] URL: <https://www.python.org>  
(Дата звернення: 01.09.2023)
2. Schmidt, M. 80 lectures on machine learning. Mark Schmidt, University of British Columbia. 11.01.2016 Web. URL: <https://www.cs.ubc.ca/~schmidtm/Courses/LecturesOnML/>

**ЗМІСТ**

1. Опис навчальної дисципліни.....	3
2. Програма навчальної дисципліни.....	5
3. Структура навчальної дисципліни.....	6
4. Питання до практичних занять.....	6
5. Самостійна робота.....	7
6. Види та методи контролю.....	12
7. Питання до іспиту.....	12
8. Критерії підсумкової оцінки знань студентів.....	13
9. Опорний конспект лекцій.....	14
10. Рекомендована література.....	22



Навчальне видання

**Мірошник Марина Анатоліївна**

**Горбачов Віктор Едуардович**

**Григор'єва Тетяна Ігорівна**

## **ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ ТЕХНОЛОГІЇ**

---

Методичні рекомендації для самостійної роботи здобувачів

Українською мовою