

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра програмних систем і технологій



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Програмування мультиагентних систем

для студентів

галузь знань
спеціальність
освітній рівень
освітня програма
вид дисципліни

12 Інформаційні технології

**121 Інженерія програмного забезпечення
магістр**

**Інженерія програмного забезпечення
вибіркова**

Форма навчання

денна

Навчальний рік

2025/2026

Семестр

4

Кількість кредитів ECTS

3

Мова викладання

українська

Форма заключного контролю

іспит

Викладачі: **Бичков О.С., професор, д.т.н.**

Мета дисципліни – формування у студентів теоретичних знань та практичних навичок програмування мультиагентних систем (МАС) з використанням мови програмування Rust. Курс охоплює основи мови Rust, включаючи систему ownership, borrowing та lifetimes, а також концепції паралельного та асинхронного програмування для створення автономних агентів та їх координації у рої. Особлива увага приділяється безпечному програмуванню без data races, що забезпечується унікальними властивостями компілятора Rust.

Попередні вимоги до опанування або вибору навчальної дисципліни:

Дисципліна базується на знаннях, вміннях і практичних навичках, отриманих студентами при вивченні дисциплін «Основи програмування», «Об'єктно-орієнтоване програмування», «Операційні системи».

Анотація навчальної дисципліни:

«Програмування мультиагентних систем» присвячена отриманню студентом компетенцій для проектування та реалізації автономних агентів і мультиагентних систем. Курс використовує мову програмування Rust як основний інструмент, що забезпечує безпеку пам'яті та потокобезпечність на етапі компіляції. Студенти вивчають концепції автономних агентів, патерни комунікації між агентами, багатопотокове та асинхронне програмування, Actor Model та практичні аспекти координації рою агентів.

Завдання вивчення дисципліни:

- формування у здобувачів вищої освіти розуміння концепцій мультиагентних систем та автономних агентів;
- оволодіння основами мови програмування Rust: синтаксис, система типів, ownership, borrowing;
- вивчення механізмів багатопотокового програмування: threads, Arc, Mutex, channels;
- ознайомлення з асинхронним програмуванням: async/await, Tokio runtime;
- практичне застосування патернів Actor Model для побудови МАС;
- розробка наскрізного проєкту симуляції автономного рою БПЛА.

Результати навчання за дисципліною:

Код	Результат навчання (1. знати; 2. вміти; 3. комунікація; 4. автономність та відповідальність)	Форми та методи навчання	Методи оцінювання	Відсоток у підсумковій оцінці з дисципліни макс/мін
1.1	Знати синтаксис та семантику мови Rust, систему типів, ownership та borrowing	Лекція, лабораторне заняття	Захист лабораторних робіт, МКР	20/10
1.2	Знати концепції мультиагентних систем, архітектуру автономних агентів	Лекція, лабораторне заняття	Захист лабораторних робіт, МКР	20/10
2.1	Вміти розробляти безпечний код на Rust з використанням ownership та borrowing	Лабораторне заняття	Захист лабораторних робіт	40/20
3.1	Здатність працювати в команді над проектом MAC	Лабораторне заняття	Захист проекту	10/5
4.1	Демонструвати відповідальність за якість та безпеку коду	Лабораторне заняття	Захист лабораторних робіт	10/5

1. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни (код)					
	1.1	1.2	2.1	3.1	4.1
Програмні результати навчання (назва)					
Програмні результати (Стандарт вищої освіти України Спеціальність 121- Інженерія програмного забезпечення. Перший (бакалаврський) рівень)					
ПР01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.	++		+	+	
ПР03. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.	+				+
ПР06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.	+	+	+		
ПР17. Вміти застосовувати методи компонентної розробки програмного забезпечення.			+		+

Схема формування оцінки

Контроль знань студентів здійснюється за модульно-рейтинговою системою. Результати навчальної діяльності студентів оцінюються за 100-бальною шкалою.

Підсумковий контроль (іспит) проводиться у формі іспиту. Максимально можлива кількість балів, що може бути отримана протягом семестру 60. Максимальна можлива кількість балів за іспит – 40.

Мінімальний пороговий рівень оцінки, що може бути отримана протягом семестру, складає 50% від максимально можливої кількості балів, тобто - 30.

Якщо студент не здав лабораторні роботи, то він не допускається до іспиту. Для студентів, які набрали сумарно меншу кількість балів ніж пороговий мінімум, для одержання допуску до іспиту необхідно перездати модульну контрольну роботу, виконати та захистити лабораторні роботи.

Здобувачі вищої освіти, які упродовж семестру не досягли мінімального порогового рівня проходять оцінювання результатів навчання, яку не були оцінені протягом семестру.

У випадку відсутності студента з поважних причин відпрацювання та перездача МКР здійснюються у відповідності до «Положення про організацію освітнього процесу» від 11 квітня 2022 року.

Поточне оцінювання:

Поточне оцінювання проводиться для:

- лабораторних робіт 5 шт. – 6/30 балів
- модульної контрольної роботи - 20 балів
- практичної частини STEM завдання – 10 балів

Підсумкове оцінювання у формі іспиту:

Екзаменаційний білет складається із тестових завдань і практичних задач.

Максимальна можлива кількість балів за іспит – 40.

При результаті підсумкового оцінювання менше 24 балів виставляється оцінка незадовільно.

Організація оцінювання:

Упродовж семестру, після завершення лекційного матеріалу проводиться захист лабораторних робіт та оцінювання модульної контрольної роботи.

Екзаменаційна робота проводиться письмово із співбесідою (за необхідності). Співбесіда проводиться та темами лекцій, самостійної роботи та теоретичною частиною STEM завдання.

На модульний контроль виносяться теми навчальних занять (аудиторної роботи).

На іспит виносяться додатково теми самостійної роботи.

Рівень досягнення всіх запланованих результатів навчання визначається за результатами:

- оцінювання модульної контрольної роботи
- захистів виконаних лабораторних робіт,
- захисту теоретичної частини STEM завдання,
- оцінювання екзаменаційної роботи.

Оцінювання за формами контролю:

Форма контролю	Мін. балів 30	Макс. балів 60
Виконання та захист лабораторних робіт (7 робіт)	15	30
Модульна контрольна робота	10	20
Практична частина STEM завдання	5	10
ВСЬОГО	30	60

Шкала відповідності оцінок:

Відмінно / Excellent	90-100
Добре / Good	75-89
Задовільно / Satisfactory	60-74
Незадовільно / Fail	0-59

СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ТЕМАТИЧНИЙ ПЛАН ЛЕКЦІЙ І ЛАБОРАТОРНИХ ЗАНЯТЬ

№	Назва теми	Лекції	Лаб.	СР
Змістовий модуль. Основи Rust та мультиагентних систем				
1	Вступ. Концепція агента та MAC. Налаштування середовища Rust	2	2	6
2	Змінні, типи даних, функції в Rust. Моделювання стану агента	2	2	8
3	Ownership та Borrowing: унікальна система керування пам'яттю Rust	2	2	10
4	Struct та Enum. Машина станів агента. Pattern Matching	2	2	8
5	Traits та Generics. Абстракції поведінки агента	2	2	8
6	Багатопотоковість: threads, Arc, Mutex, channels	2	2	11
7	Async/Await та Tokio. Actor Model для MAC	2	2	10
	ВСЬОГО	14	14	61

Загальний обсяг 89 год., в тому числі:

Лекції – 14 год.; Лабораторні заняття – 14 год.; Самостійна робота – 61 год.

Теми лабораторних робіт:

1. Налаштування середовища Rust. Перший проєкт «Hello, Agent!»
2. Моделювання стану агента: змінні, функції, структури керування
3. Ownership та Borrowing на прикладі ресурсів агента
4. Структури та перелічення. Машина станів БПЛА
5. Traits та абстракції агента. Поліморфізм поведінки
6. Локальний рій агентів: threads, Arc, Mutex, channels
7. Асинхронний рій агентів: Tokio, Actor Model

Індивідуальне завдання (проєкт):

Розробка симуляції автономного рою БПЛА з використанням Actor Model. Проєкт включає: реалізацію агентів-БПЛА з машиною станів, комунікацію через message passing, координацію рою для виконання спільного завдання (патрулювання, пошук, доставка).

Рекомендовані джерела:

Основна:

8. Klabnik S., Nichols C. The Rust Programming Language. 2nd ed. No Starch Press, 2023. 560 p.
9. Blandy J., Orendorff J., Tindall L. Programming Rust: Fast, Safe Systems Development. 2nd ed. O'Reilly Media, 2021. 738 p.
10. Wooldridge M. An Introduction to MultiAgent Systems. 2nd ed. Wiley, 2009. 484 p.
11. Бичков О.С. Основи сучасного програмування. Київ: ВПЦ «Київський університет», 2007. 260 с.

Додаткова:

12. Gjengset J. Rust for Rustaceans: Idiomatic Programming for Experienced Developers. No Starch Press, 2021. 280 p.
13. Shoham Y., Leyton-Brown K. Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge University Press, 2009. 532 p.
14. Tokio Documentation. <https://tokio.rs/>
15. The Rust Book (офіційна документація). <https://doc.rust-lang.org/book/>