

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І
ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ**

Кафедра комп'ютерних систем, мереж та кібербезпеки

ЗАТВЕРДЖЕНО

Факультет інформаційних технологій
«____» _____ 2025 р.

**РОБОЧА ПРОГРАМА
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«Паралельні та розподілені обчислення»**

Галузь знань	123 «Інформаційні технології»
Спеціальність	123 «Комп'ютерна інженерія»
Освітня програма	«Комп'ютерна інженерія»
Факультет	інформаційних технологій
Розробник:	Смолій В.В., доцент, к.т.н., доцент

Київ – 2025 р.

Опис навчальної дисципліни «Паралельні та розподілені обчислення»

Вивчаються тенденції розвитку науки та техніки в галузі створення паралельних систем обробки даних; призначення, класифікацію та структуру цих засобів; способи їх організації, методи проектування алгоритмів з паралельної обробки даних, оцінювати їх складність, вибирати та застосовувати інструментальні засоби та мови з паралельної обробки даних, працювати з технічною літературою, довідниками, стандартами, технічною документацією.

Галузь знань, напрям підготовки, спеціальність, освітньо-кваліфікаційний рівень		
Галузь знань	Інформаційні технології	
Спеціальність	123 – «Комп'ютерна інженерія»	
перший рівень	Бакалавр	
Характеристика навчальної дисципліни		
Вид	Обов'язкова	
Загальна кількість годин	90	
Кількість кредитів ECTS	3	
Кількість змістових модулів	3	
Курсовий проект (робота) (якщо є в робочому навчальному плані)	-	
Форма контролю	Іспит	
Показники навчальної дисципліни для денної та заочної форм навчання		
	денна форма навчання	заочна форма навчання
Рік підготовки	4	
Семестр	7	
Лекційні заняття	30 год.	
Практичні, семінарські заняття		
Лабораторні заняття	30 год.	
Самостійна робота	30 год.	
Індивідуальні завдання		
Кількість тижневих годин для денної форми навчання: аудиторних	4 год.	

1. Мета, компетентності та програмні результати навчальної дисципліни

Мета: забезпечення сприяння формуванню знань щодо проектування та використання технічних засобів систем високопродуктивної обробки даних на основі принципів паралелізму, підготовки студентів в галузі застосування апаратних засобів та розробки програмного забезпечення систем паралельної обробки даних.

Набуття компетентностей:

Інтегральна компетентність (ІК): Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми під час професійної діяльності в комп'ютерній галузі або навчання, що передбачає застосування теорій та методів комп'ютерної інженерії і характеризується комплексністю та невизначеністю умов.

Загальні компетентності:

ЗК1. Здатність до абстрактного мислення, аналізу і синтезу.

ЗК6. Навички міжособистісної взаємодії.

ЗК7. Вміння виявляти, ставити та вирішувати проблеми.

Спеціальні (фахові) компетентності:

СК6. Здатність проектувати, впроваджувати та обслуговувати комп'ютерні системи та мережі різного виду та призначення.

СК7. Здатність використовувати та впроваджувати нові технології, включаючи технології розумних, мобільних, зелених і безпечних обчислень, брати участь в модернізації та реконструкції комп'ютерних систем та мереж, різноманітних вбудованих і розподілених додатків, зокрема з метою підвищення їх ефективності.

СК12. Здатність ідентифікувати, класифікувати та описувати роботу програмно-технічних засобів, комп'ютерних та кіберфізичних систем, мереж та їхніх компонентів шляхом використання аналітичних методів і методів моделювання.

СК13. Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій.

СК14. Здатність проектувати системи та їхні компоненти з урахуванням усіх аспектів їх життєвого циклу та поставленої задачі, включаючи створення, налаштування, експлуатацію, технічне обслуговування та утилізацію.

Програмні результати навчання (ПРН):

ПРН1. Знати і розуміти наукові положення, що лежать в основі функціонування комп'ютерних засобів, систем та мереж.

ПРН2. Мати навички проведення експериментів, збирання даних та моделювання в комп'ютерних системах.

ПРН3. Знати новітні технології в галузі комп'ютерної інженерії.

ПРН4. Знати та розуміти вплив технічних рішень в суспільному, економічному, соціальному і екологічному контексті.

ПРН7. Вміти розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності.

ПРН8. Вміти системно мислити та застосовувати творчі здібності до формування нових ідей.

ПРН13. Вміти ідентифікувати, класифікувати та описувати роботу комп'ютерних систем та їх компонентів.

ПРН14. Вміти поєднувати теорію і практику, а також приймати рішення та виробляти стратегію діяльності для вирішення завдань спеціальності з урахуванням загальнолюдських цінностей, суспільних, державних та виробничих інтересів.

ПРН15. Вміти виконувати експериментальні дослідження за професійною тематикою.

2. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин											
	денна форма						Заочна форма					
	усього	у тому числі					усьог	у тому числі				
		л	п	лаб	інд	с.р.	о	л	п	лаб	інд	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Модуль 1. Паралелізм в системах із загальною пам'яттю.												
Тема 1. Визначення паралелізму та розподіленості обчислень	6	3				3						
Тема 2. Програмування із OpenMP.	10	3		4		3						
Тема 3. Розпаралелення по задачам із застосуванням OpenMP	10	3		4		3						
Тема 4. Керування розподілом ітерацій із застосуванням OpenMP	10	3		4		3						
Тема 5. Узгодження паралельних обчислень	9	3		3		3						
Разом за змістовим модулем 1	45	15		15		15						
Модуль 2. Паралелізм в системах із розподіленою пам'яттю.												
Тема 6. Програмування із MPI	10	3		4		3						
Тема 7. Колективні взаємодії в MPI	9	3		3		3						
Тема 8. Похідні типи даних в MPI	10	3		4		3						
Тема 9. Моделі комп'ютерних систем з паралельними та розподіленими обчисленнями	6	3				3						
Тема 10. Переналаштування в MPI	10	3		4		3						
Разом за змістовим модулем 2	45	15		15		15						
Усього годин за курс	90	30		30		30						

3. Теми лекцій

№ з/п	Назва теми	Кількість годин
1.	Розпаралелення поза задачам із застосуванням OpenMP.	4
2.	Узгодження паралельних обчислень.	2
3.	Моделі комп'ютерних систем з паралельними та розподіленими обчисленнями.	4
4.	Головні положення з гетерогенних обчислень.	4
5.	Програмування із OpenCL.	4
6.	Програмування із CUDA.	2
7.	Метрики паралельних обчислень.	2
8.	Методика надання паралелізму комп'ютерним обчисленням.	2
9.	Похідні типи даних в MPI.	2
10.	Керування розподілом ітерацій із застосуванням OpenMP.	4
	Усього	30

4. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1.	Вказання на паралелізм в програмі з використанням OpenMP	2
2.	Розподілення обчислень між паралельними секціями в програмі з OpenMP	2

3.	Розподілення обчислень циклу в програмі з OpenMP	4
4.	Обмін повідомленнями між окремими процесами з використанням MPI	4
5.	Застосування колективних взаємодій у MPI	2
6.	Похідні типи даних в MPI	4
7.	Віртуальні топології в MPI	4
8.	Вказання на паралелізм в програмі з використанням CUDA	4
9.	Вказання на паралелізм в програмі з використанням OpenCL	4
	Разом	30

5. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Класифікації комп'ютерних систем. Технічні характеристики комп'ютерних систем. Приклади застосування комп'ютерних MIMD-систем. Методика обробки даних векторними командами.	2
2	Налагодження програми з OpenMP. Правила застосування директиви Parallel OpenMP. Керування друком повідомлень. Визначення номеру потоку. Розрахунок часу виконання паралельної програми.	2
3	Налагодження програми з директивою sections OpenMP. Дослідження паралельного обчислення контрольного виразу із застосуванням секцій.	2
4	Налагодження програми з директивою parallel for OpenMP. Дослідження паралельного обчислення контрольного виразу із застосуванням ітераційного паралелізму.	2
5	Проаналізувати синхронізацію на прикладах розв'язання «класичних задач»: -«Виробники-Споживачі» (Producer-Consumer problem), -«Читачі-Письменники» (Readers-Writers problem), -«Філософи, які обідають» (Dining Philosopher problem), -«Цирульник, який спить» (Sleeping Barber problem).	2
6	Налагодити режим роботи з використанням MPI. Апробувати роботу з MPI на прикладах - програма для визначення номеру процесу, виведення номеру процесу і кількості активних процесів, характеристик системного таймеру (роздільну здатність і час, необхідний на виконання окремих дій)	2
7	Протестувати програму обчислення суми ряду, для чого забезпечити контрольний друк початкових значень чисел масиву, друк чисел в масивах процесів та обчислення суми іншим довільним способом. Розробити програму обчислення суми ряду при довільному розмірі матриці та кількості процесів, формат чисел – з рухомою комою. Порівняти швидкодію роботи програм.	2
8	Завершити створення програми розсилки різнотипних даних (допускається зміна характеристик початкових даних). Проаналізувати процедуру опрацювання різнотипних даних. Надати графічне пояснення щодо результатів роботи програми розсилки різнотипних даних.	2
9	Приклади використання топології. Кодування вузлів у гіперкубі. Схемотехнічна реалізація арбітражу в топології Bus та Crossbus.	2
	Багатошинна динамічна топологія. Схемотехнічна реалізація комутаторів. Приклади багатоступеневої динамічної топології мережі, що блокує. Приклади багатоступеневої динамічної топології, що не блокує. Приклади багатоступеневої динамічної топології, що ре конфігурує.	
10	Проаналізувати процедуру створення програм з різними групами і комунікаторами. Застосувати різні топології паралельних системи: (декартова, графова) при розв'язанні прикладу обчислення математичного виразу. Надати графічне пояснення щодо результатів роботи програми.	2

11	Приклади гетерогенних комп'ютерних систем з графічними процесорами на базі SIMD-архітектури.	2
12	Аналіз методики ефективної генерації програмного коду з CUDA.	2
13	Аналіз методики ефективної генерації програмного коду з OpenCL.	2
14	Приклади застосування законів Амдала, Густавсона для оцінки параметрів паралельних обчислень.	2
15	Приклади застосування методики розпаралелення для реалізації матричних перетворень та сортування даних.	2
	Разом	30

Самостійна робота студентів передбачає:

- систематичне вивчення лекційного матеріалу і навчальної літератури, що рекомендуються;
- сумлінну підготовку до лабораторних занять;
- роботу над індивідуальними завданнями по лабораторним роботам;
- розробка курсового проєкту;
- вчасне і якісне оформлення звітів про лабораторні роботи.

Систематичний контроль за самостійною роботою студентів і якістю засвоєння ними поточного навчального матеріалу передбачає:

- перевірку на лабораторних роботах підготовки до виконання роботи;
- вивчення літератури, що рекомендувалася, та конспекту лекцій;
- оформлення звітів з лабораторних робіт;
- перевірку виконання етапів курсового проєкту.

6. Методи та засоби діагностики результатів навчання:

- іспит;
- захист звітів з лабораторних робіт;

7. Методи навчання:

- словесний метод (лекція, співбесіда тощо);
- практичний метод (лабораторні заняття): виконання лабораторних робіт з використанням наочних технічних засобів навчання у вигляді систем моделювання за допомогою інженерних пакетів проектування цифрових пристроїв;
- метод командної роботи;
- метод проєктного навчання;
- наочний метод (метод ілюстрацій, метод демонстрацій);
- робота з навчально-методичною літературою;
- відеометод (дистанційні, мультимедійні);
- самостійна робота (вивчення теоретичного матеріалу, виконання індивідуальних завдань).

Поточний контроль знань студентів проводиться:

- на лабораторних роботах оцінюється підготовка до роботи, обсяг її виконання, результати захисту звіту;
- на лекційних заняттях виконується вибіркове опитування студентів.

8. Оцінювання результатів навчання

Оцінювання знань здобувача вищої освіти відбувається за 100-бальною шкалою і переводиться в національну оцінку згідно чинного «Положення про екзамен та заліки у НУБіП України»

8.1. Розподіл балів за видами навчальної діяльності

Вид навчальної діяльності	Результати навчання	Оцінювання	
Модуль 1. Паралелізм в системах із загальною пам'яттю			
Лабораторна робота 1. Вказання на паралелізм в програмі з використанням OpenMP.	ПРН 1,2,3,4,7,8,13,14,15. Основними завданнями вивчення дисципліни є теоретична і практична підготовка студентів щодо формування здатності створювати та аналізувати програмні компоненти комп'ютерних систем з використанням засобів підтримки паралельних комп'ютерних обчислювальних операцій. Внаслідок вивчення дисципліни студенти повинні знати методи паралельних обчислень для інженерно-технічних задач; основні напрями забезпечення паралелізму в комп'ютерних системах; методику розробки паралельних програм. Вміти створювати модель виконання паралельних програм; оцінювати ефективність паралельних обчислень; аналізувати складність обчислень і можливість розпаралелення алгоритмів; застосовувати принципи розробки паралельних програм; оцінювати основні параметри паралельних програм.	20	
Самостійна робота з підготовки до лабораторної роботи 1.		20	
Лабораторна робота 2. Розподілення обчислень між паралельними секціями в програмі з OpenMP.			20
Самостійна робота з підготовки до лабораторної роботи 2.			
Лабораторна робота 3. Розподілення обчислень циклу в програмі з OpenMP.			
Самостійна робота з підготовки до лабораторної роботи 3.		20	
Лабораторна робота 4. Обмін повідомленнями між окремими процесами з використанням MPI.			40
Самостійна робота з підготовки до лабораторної роботи 4.			
Лабораторна робота 5. Застосування колективних взаємодій у MPI.			
Самостійна робота з підготовки до лабораторної роботи 5.	100		
Всього за модулем 1			
Модуль 2. Паралелізм в системах із розподіленою пам'яттю			
Лабораторна робота 6. Похідні типи даних в MPI.	ПРН 1,2,3,4,7,8,13,14,15. Основними завданнями вивчення дисципліни є теоретична і практична підготовка студентів щодо формування здатності створювати та аналізувати програмні компоненти комп'ютерних систем з використанням засобів підтримки паралельних комп'ютерних обчислювальних операцій. Внаслідок вивчення дисципліни студенти повинні знати методи паралельних обчислень для інженерно-технічних задач; основні напрями забезпечення паралелізму в комп'ютерних системах; методику розробки паралельних програм. Вміти створювати модель виконання паралельних програм; оцінювати ефективність паралельних обчислень; аналізувати складність обчислень і можливість розпаралелення алгоритмів; застосовувати принципи розробки паралельних програм; оцінювати основні параметри паралельних програм.	20	
Самостійна робота з підготовки до лабораторної роботи 6		20	
Лабораторна робота 7. Віртуальні топології в MPI.			
Самостійна робота з підготовки до лабораторної роботи 7		20	
Лабораторна робота 8. Вказання на паралелізм в програмі з використанням CUDA.			40
Самостійна робота з підготовки до лабораторної роботи 8			
Лабораторна робота 9. Вказання на паралелізм в програмі з використанням OpenCL.		100	
Самостійна робота з підготовки до лабораторної роботи 9			
Всього за модулем 2			
Навчальна робота	(M1 + M2)/2*0,7 ≤ 70		

Вид навчальної діяльності	Результати навчання	Оцінювання
Залік		30
Всього за курс	(Навчальна робота + екзамен) ≤ 100	

Самостійна робота з підготовки до лабораторної роботи передбачає вивчення лекційного матеріалу, підготовку до виконання лабораторної роботи, підготовку звітів з лабораторної роботи.

8.2. Шкала оцінювання знань здобувача вищої освіти

Рейтинг здобувача вищої освіти, бали	Оцінка за національною системою (екзамени/заліки)
90-100	відмінно
74-89	добре
60-73	задовільно
0-59	незадовільно

8.3. Політика оцінювання

Політика щодо дедлайнів та перескладання:	Дедлайни визначені в ЕНК. Роботи, які здаються з порушенням термінів без поважних причин, оцінюються на нижчу оцінку.
Політика щодо академічної доброчесності:	Списування під час самостійних робіт, тестування та екзаменів заборонені (в т.ч. з використанням мобільних пристроїв).
Політика щодо відвідування:	Відвідування занять є обов'язковим. За об'єктивних причин (наприклад, хвороба, міжнародне стажування) навчання може відбуватись індивідуально (в дистанційній on-line формі за погодженням з деканом факультету відповідно до графіку ліквідації заборгованостей після закінчення дії об'єктивних причин).

9. Методичне забезпечення

Електронний навчальний курс на платформі Moodle вміщує методичне забезпечення включаючи: лекції, презентації до лекцій, методичні вказівки до виконання лабораторних робіт, глосарій термінів тощо.

<https://elearn.nubip.edu.ua/enrol/index.php?id=1589>

10. Рекомендована література

Базова

1. Roman, Trobec. (2018) Introduction to Parallel Computing. From Algorithms to Programming on State-of-the-Art Platforms / Roman, Trobec, Boštjan Slivnik, Patricio Bulić, Borut Robič. (Springer Nature Switzerland AG 2018).

2. Tanenbaum, Andrew S. (2013) Structured computer organization / Andrew S. Tanenbaum, Todd Austin. -- 6th ed. (Pearson Education, Inc., publishing as Prentice Hall).
3. Patterson, David A. (2014) Computer organization and design: the hardware/software interface / David A. Patterson, John L. Hennessy. - 5th ed. (The Morgan Kaufmann series in computer architecture and design).
4. Stallings, William. (2016) Computer organization and architecture : designing for performance / William Stallings. - 10th ed. (Pearson Education, Inc., Hoboken).

Допоміжна

1. Gregory R. Andrews. (2000) Foundations of Multithreaded, Parallel and Distributed Programming. Addison Wesley Longman, Inc
2. Intel® 64 and IA-32 Architectures Software Developer's Manual

11. Інформаційні ресурси

1. <https://www.openmp.org//>
2. <https://computing.llnl.gov/tutorials/openMP/>
3. <https://sourceforge.net/projects/mingw-w64/>
4. <http://mingw-w64.org/doku.php>
5. <https://www.gnu.org/software/software.html>
6. <https://docs.microsoft.com/en-us/message-passing-interface/microsoft-mpi>
7. <https://docs.microsoft.com/en-us/archive/blogs/windowshpc/>
8. [how-to-compile-and-run-a-simple-ms-mpi-program](https://docs.microsoft.com/en-us/archive/blogs/windowshpc/how-to-compile-and-run-a-simple-ms-mpi-program)