

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

ЗАТВЕРДЖУЮ
В. о. декана факультету комп'ютерних
інформаційних технологій
Ігор ЯКИМЕНКО
" 31 " 2023 р.

ЗАТВЕРДЖУЮ
В. о. проректора
з науково-педагогічної роботи
Віктор ОСТРОВЕРХОВ
" 31 " 2023 р.

РОБОЧА ПРОГРАМА

з дисципліни

«Паралельні та розподілені обчислення»

Ступінь вищої освіти: магістр
Галузь знань – 12 «Інформаційні технології»
Спеціальність – 121 «Інженерія програмного забезпечення»
Освітньо-наукова програма – «Математичне та програмне забезпечення
комп'ютерних систем»

Кафедра комп'ютерних наук

Форма навчання	Курс	Семестр	Лекції (год.)	Лаб. (год.)	ІРС (год.)	Тренінг (год.)	СРС (год.)	Разом (год.)	Екз. (сем.)
Денна	II	3	24	24	5	4	93	150	3

5/10/2023

Тернопіль – ЗУНУ
2023

Робоча програма складена на основі освітньо-наукової програми «Математичне та програмне забезпечення комп'ютерних систем» підготовки магістра галузі знань 12 «Інформаційні технології» спеціальності 121 «Інженерія програмного забезпечення», затвердженої Вченою радою ЗУНУ, протокол № 9 від 15.06.23 р.

Робоча програма розроблена доцентом кафедри комп'ютерних наук, к.т.н. Манжулою Володимиром Івановичем

Робоча програма затверджена на засіданні кафедри комп'ютерних наук, протокол №1 від 28 серпня 2023р.

Завідувач кафедри д.т.н, професор  Андрій ПУКАС

Розглянуто та схвалено групою забезпечення спеціальності 121 Інженерія програмного забезпечення, протокол № 1 від 30.08 2023р.

Голова групи
забезпечення спеціальності,
д.т.н., професор

 Микола ДИВАК

Гарант ОП,
к.т.н., доцент

 Ірина СПИВАК

СТРУКТУРА РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«Паралельні та розподілені обчислення»

1. Опис дисципліни «Паралельні та розподілені обчислення»

Дисципліна – «Паралельні та розподілені обчислення»	Галузь знань, спеціальність, ступінь вищої освіти	Характеристика навчальної дисципліни
Кількість кредитів ECTS- 5	Галузь знань: 12 «Інформаційні технології»	Статус дисципліни: обов'язкова Мова навчання: українська
Кількість залікових модулів – 4	Спеціальність: 121 «Інженерія програмного забезпечення»	Рік підготовки: <i>Денна – 2</i> Семестр: <i>Денна – 3</i>
Кількість змістових модулів – 3	Ступінь вищої освіти: магістр	Лекції: <i>Денна – 24 год.</i> Лабораторні заняття: <i>Денна – 24 год.</i>
Загальна кількість годин – 150		Самостійна робота: <i>Денна – 93 год.</i> <i>Тренінг – 4</i> Індивідуальна робота: <i>Денна – 5 год.</i>
Тижневих годин: Денна форма навчання 3 семестр – 12 год., з них аудиторних – 4 год.		Вид підсумкового контролю: <i>Денна – екзамен, 3;</i>

2. Мета й завдання вивчення дисципліни «Паралельні та розподілені обчислення»

2.1. Мета вивчення дисципліни

Метою викладання навчальної дисципліни «Паралельні та розподілені обчислення» є формування у студентів теоретичних знань і практичних навичок з організації паралельних та розподілених обчислень, застосування апаратних та програмних засобів високопродуктивних обчислень.

Мета проведення лекцій полягає у тому, щоб ознайомити студентів із головними принципами та засобами паралельних та розподілених обчислень, сучасними засобами високопродуктивних обчислень.

Мета проведення лабораторних занять полягає в ознайомленні студентів з практичними основами розпаралелення класичних алгоритмів, застосування основних принципів побудови паралельних програм з використанням сучасних апаратних засобів.

2.2. Завдання вивчення дисципліни

Завдання проведення лекцій:

- викладення студентам у відповідності з програмою та робочим планом основних питань побудови паралельних алгоритмів та програм, звертаючи головну увагу на необхідність і можливість їх використання в практичній фаховій діяльності;
- сформуванню у студентів цілісну систему теоретичних знань з курсу «Паралельні та розподілені обчислення».

Завдання проведення лабораторних занять:

- засвоїти методи, принципи та засоби для побудови паралельних алгоритмів та програм;
- навчитися використовувати сучасні технології для паралельних та розподілених обчислень;
- глибше засвоїти та закріпити теоретичні знання, одержані на лекціях.

2.3 Найменування та опис компетентностей, формування котрих забезпечує вивчення дисципліни «Паралельні та розподілені обчислення»:

- здатність ефективно керувати фінансовими, людськими, технічними та іншими проєктними ресурсами у сфері інженерії програмного забезпечення;
- здатність застосовувати і розвивати фундаментальні і міждисциплінарні знання для успішного розв'язання наукових проблем інженерії програмного забезпечення;

2.4. Передумови для вивчення дисципліни. новий пункт

Дисципліни, які повинні бути вивчені попередньо:

- «Архітектура комп'ютера»;
- «Алгоритми та структури даних»;
- «Об'єктно-орієнтоване програмування»;
- «Архітектура та проектування програмного забезпечення».

2.5. Результати навчання

У результаті вивчення дисципліни «Паралельні та розподілені обчислення» студенти повинні вміти:

- аналізувати, оцінювати і застосовувати на системному рівні сучасні програмні та апаратні платформи для розв'язання складних задач інженерії програмного забезпечення;
- модифікувати існуючі та розробляти нові алгоритмічні рішення детального проектування програмного забезпечення.
- розробляти математичне і програмне забезпечення для наукових досліджень в галузі інженерії програмного забезпечення.

3. Зміст дисципліни «Паралельні та розподілені обчислення»

Змістовий модуль 1. Теорія паралельних обчислень та систем

Тема 1. Цілі та задачі паралельних обчислень.

Обмеження максимальної продуктивності однопроцесорних ПК. Постійна необхідність розв'язку задач, які перевищують можливості сучасних ПК (проблеми «великого виклику»). Необхідність колективного режиму розв'язку задач. Автоматизація управління розподілених технічних систем. Технічні вимоги по зниженню вартості та збільшенню надійності. Відмінність багатозадачних, паралельних і розподілених обчислень.

Література: 1- 3.

Тема 2. Проблеми використання паралелізму.

Існування послідовних алгоритмів (закон Амдаля). Збільшення продуктивності послідовних комп'ютерів (закон Мура). Втрати на взаємодію та передачу даних (гіпотеза Мінського). Висока вартість паралельних систем (закон Гроша). «Послідовність» існуючих алгоритмів і програмного забезпечення. Складність розробки паралельних алгоритмів. Трудомісткість перевірки правильності паралельних програм.

Література: 2 - 5.

Тема 3. Класифікація багатопроцесорних обчислювальних систем. Апаратні засоби високопродуктивних обчислень.

Класифікація Фліна. Потоки даних і команд. Класифікація Шора. Структурна нотація Хокні та Джесхоупа. Апаратні засоби високопродуктивних обчислень: багатопроцесорні системи (з спільною та розподіленою пам'яттю, симетричні мультипроцесори SMP, масивно-паралельні комп'ютерні системи MPP), багатоядерні системи, кластерні системи, grid-системи, багатоядерні графічні процесори. Приклади сучасних високопродуктивних обчислювальних систем. Суперкомп'ютерні обчислювальні системи в Україні та світі.

Література: 1- 5.

Змістовий модуль 2. Методи паралельних та розподілених обчислень

Тема 4. Оцінка ефективності паралельних обчислень.

Показники ефекту розпаралелення (прискорення, продуктивність, ефективність). Способи оцінки показників. Максимальна (пікова) продуктивність. Залежність ефективності паралельних обчислень від особливостей апаратури (архітектура, кількість процесорів, топологія каналів передачі даних).

Література: 4 - 7.

Тема 5. Загальні принципи побудови паралельних алгоритмів і програм.

Розпаралелення обчислень на рівні команд, виразів, програмних модулів, задач. Вибір паралельного алгоритму. Реалізація алгоритму в виді паралельної програми. Декомпозиція алгоритму на блоки, що виконуються паралельно.

Література: 1- 5.

Тема 6. Технологічні аспекти розпаралелення.

Розподіл задач по процесорах і балансування. Організація взаємодії. Поняття процесу. Синхронізація паралельних процесів: семафори, м'ютекси, події, бар'єри. Концепція ресурсу. Взаємовиключення паралельних процесів: алгоритм Деккера, семафори Дейкстра, монітори Вірта. Взаємодія паралельних процесів. Передача повідомлень: черги, поштові ящики, порти. Проблеми взаємодії процесів. Поняття тупика та умови його виникнення. Запобігання тупиків. Виявлення тупиків і відновлення стану процесів. Багатозадачний режим роботи ПК як окремий випадок паралельної обробки.

Література: 1- 5.

Змістовий модуль 3. Технології реалізації паралельних та розподілених обчислень

Тема 7. Програмні засоби розробки паралельних програм.

Мови паралельного програмування. Використання спеціалізованих мов паралельного програмування (Occam). Застосування паралельних розширень існуючих алгоритмічних мов (HPF). Побудова паралельного програмного забезпечення на основі існуючих послідовних програм з використанням засобів предкомпіляції (технологія OpenMP). Використання технологічних (інструментальних) бібліотек паралельного програмування (бібліотеки MPI і PVM). Мови паралельного програмування на основі розділеного глобально-адресного простору (UPC, CAF, Chapel, X10). Технологія DVM і мова програмування T++. Технології розробки паралельних програм для графічних процесорів (CUDA, OpenCL). Організація паралельних обчислень в математичних пакетах (на прикладі MatLab). Загальна характеристика проблеми тестування і відлагодження паралельних програм.

Література: 1- 5.

Тема 8. Технологія OpenMP для розробки паралельних програм для систем із спільною пам'яттю.

Загальна характеристика стандарту OpenMP. Створення паралельних областей. Розподіл обчислювального навантаження між потоками. Робота з даними. Синхронізація. Функції та змінні оточення. Загальна характеристика середовища виконання.

Література: 6.

Тема 9. Розробка паралельних програм для систем з розподіленою пам'яттю з використанням бібліотеки MPI.

Система MPI. Загальна характеристика. Підтримка моделі взаємодії паралельних обчислень за допомогою передачі повідомлень. Управління даними. Управління процесами. Загальна характеристика середовища виконання. Основні програмні примітиви системи MPI. Приклад використання.

Література: 7, 8.

Тема 10. Технологія розробки паралельних програм для процесорів нових архітектур.

Можливі переваги обчислень на графічному процесорі. Засоби розробки для графічного процесора. Використання технології CUDA для обчислень на

графічних процесорах. Модель програмування. Модель виконання та ієрархія потоків. Ієрархія пам'яті. Інтерфейс програмування CUDA. Специфікатори типів змінних і функцій. Вбудовані змінні. Конфігурування виконання ядер. Синхронізація. Управління пам'яттю. Загальні принципи обчислень на базі технології CUDA. Використання OpenCL для обчислень на графічних процесорах.

Література: 9-12.

4. Структура залікового кредиту з дисципліни «Паралельні та розподілені обчислення»

	Кількість годин				
	Лекції	Лабораторні заняття	Самостійна робота	Індивідуальна робота	Контрольні заходи
Змістовий модуль 1. Теорія паралельних обчислень та систем					
Тема 1. Цілі та задачі паралельних обчислень.	2		4		Усне опитування та тестування
Тема 2. Проблеми використання паралелізму.	2		4		Усне опитування та тестування
Тема 3. Класифікація багатопроцесорних обчислювальних систем. Апаратні засоби високопродуктивних обчислень.	2	2	4		Усне опитування та тестування
Змістовий модуль 2. Методи паралельних та розподілених обчислень					
Тема 4. Оцінка ефективності паралельних обчислень.	2	2	5	2	Усне опитування та тестування
Тема 5. Загальні принципи побудови паралельних алгоритмів і програм.	2	4	5		Усне опитування та тестування
Тема 6. Технологічні аспекти розпаралелення.	2	2	5		Усне опитування та тестування
Змістовий модуль 3. Технології реалізації паралельних та розподілених обчислень					
Тема 7. Програмні засоби розробки паралельних програм.	4	4	5		Усне опитування та тестування
Тема 8. Технологія OpenMP для розробки паралельних програм для систем із спільною пам'яттю.	2	4	5		Усне опитування та тестування
Тема 9. Розробка паралельних програм для систем з розподіленою пам'яттю з використанням бібліотеки MPI.	2	4	5		Усне опитування та тестування
Тема 10. Технологія розробки паралельних програм для процесорів	4	4	5	2	Усне опитування та

нових архітектур.					тестування
Тренінг					Захист
Разом	24	24	97	5	

5. Тематика лабораторних занять

Лабораторне заняття 1.

Тема. Апаратні засоби високопродуктивних обчислень.

Мета. Множення матриць з використанням спільної пам'яті.

Література: 3,4.

Лабораторне заняття 2.

Тема. Оцінка ефективності паралельних обчислень.

Мета. Множення матриць з використанням розподіленої пам'яті.

Література: 9.

Лабораторне заняття 3.

Тема. Загальні принципи побудови паралельних алгоритмів і програм.

Мета. Розпаралелення методу редукції.

Література: 5.

Лабораторне заняття 4.

Тема. Розробка паралельних програм для систем з розподіленою пам'яттю з використанням бібліотеки MPI

Мета. Розпаралелення алгоритмів сортування.

Література: 6.

Лабораторне заняття 5.

Тема. Технологія OpenMP для розробки паралельних програм для систем із спільною пам'яттю.

Мета. Розпаралелення методу Гауса.

Література: 7.

Лабораторне заняття 6.

Тема. Технологія розробки паралельних програм для процесорів нових архітектур.

Мета. Налаштування технології CUDA на графічному процесорі.

Література: 9.

6. Комплексне практичне індивідуальне завдання

Комплексне практичне індивідуальні завдання з дисципліни «Паралельні та розподілені обчислення» виконується самостійно кожним студентом. КППЗ охоплює усі основні теми дисципліни «Паралельні та розподілені обчислення». Метою виконання КППЗ є оволодіння навичками застосування теоретичних

знань з дисципліни. КППЗ оформлюється у відповідності з встановленими вимогами. Виконання КППЗ є одним із обов'язкових складових модулів залікового кредиту з дисципліни «Паралельні та розподілені обчислення».

Варіанти з дисципліни «Паралельні та розподілені обчислення»:

1. Nvidia Kepler
2. Blade server
3. Nvidia Tesla
4. Nvidia Quadro
5. Неявний паралелізм
6. Технологія MPI
7. Хмарні технології
8. Storage area network
9. Temporal multithreading
10. Automatic parallelization
11. Instruction-level parallelism
12. Класифікація архітектур паралельних обчислювальних систем
13. Кластерні системи
14. Nvidia Pascal
15. GRID
16. GPGPU
17. Nvidia Maxwell
18. Multiprocessing
19. Nvidia Fermi
20. Distributed shared memory
21. TOP500
22. Sharding
23. CUDA
24. Класифікація паралельних обчислювальних систем
25. Cache coherence
26. OpenCL
27. Crossbar switch
28. Lock (computer science)
29. Data parallelism
30. Транзакційна пам'ять
31. GSI (Quard)

7. Самостійна робота

1. Методи оцінки продуктивності паралельних алгоритмів і систем
2. Основні класи паралельних комп'ютерів
3. Мови паралельного програмування
4. Області застосування і задачі паралельної обробки
5. Конвекризація і паралелізм
6. Засоби для проведення паралельних обчислень

- 7 Рівні розпаралелення
 - 8 Паралельні операції
 - 9 Основні принципи паралелізму (розпаралелення)
 - 10 Класифікація структур паралельної обробки
 - 11 Загальні зауваження стосовно оцінки продуктивності паралельних алгоритмів та систем
 - 12 Фактори, які необхідно враховувати при оцінці продуктивності
 - 13 Методи оцінки продуктивності паралельних систем
 - 14 Характеристики продуктивності паралельних алгоритмів
 - 15 Порівняння MIMD і SIMD структур за продуктивністю
 - 16 Паралелізм даних
 - 17 Паралелізм задач
 - 18 Етапи розробки паралельного алгоритму
 - 19 Структури зв'язку між процесорами
 - 20 Шинні мережі
 - 21 Мережі з комутаторами
 - 22 Структури, що забезпечують зв'язок типу «пункт-пункт»
 - 23 Основні класи паралельних комп'ютерів
 - 24 Методи комутацій
 - 25 Масивно-паралельні системи
 - 26 Симетричні мультипроцесорні системи
 - 27 Паралельні векторні процесори
 - 28 Кластерні системи
 - 29 Технології паралельного програмування Message Passing Interface
 - 30 Класифікація мов і систем паралельного програмування
- Разом

8. Організація і проведення тренінгу

Тематика: Розробка паралельної програми для процесорів нового покоління

Порядок проведення:

- Практично освоїти принципи розпаралелення алгоритмів.
 - На основі вибраної архітектури побудувати блок-схему розпаралеленого алгоритму.
 - Побудувати послідовну та паралельну програми для обраного алгоритму.
 - Представити порівняльні результати продуктивності цих програм.
- Література: 11, 12.

9. Методи оцінювання

У процесі вивчення дисципліни «Паралельні та розподілені обчислення» використовуються наступні методи оцінювання навчальної роботи студентів:

- поточне опитування;
- стандартизовані тести;
- залікове модульне тестування та опитування;

- презентації результатів виконання завдань;
- оцінювання результатів КППЗ;
- ректорська контрольна робота;
- тренінг;
- екзамен.

10. Критерії, форми поточного та підсумкового контролю

Підсумковий бал (за 100-бальною шкалою) з дисципліни «Паралельні та розподілені обчислення» визначається як середньозважена величина, залежно від питомої ваги кожної складової залікового кредиту:

Заліковий модуль 1	Заліковий модуль 2 (ректорська контрольна робота)	Заліковий модуль 3 (підсумкова оцінка за КППЗ,	Заліковий модуль 4 (письмовий екзамен)	Разом
20%	20 %	20 %	40%	100 %
1. Усне або письмове опитування під час заняття – 40 балів. 2. Лабораторні роботи – 60 балів	1. Тестові завдання (30 тестів по 2 бали за тест) – 60 балів 2. Завдання 1 – 20 балів 3. Завдання 2 – 20 балів	1. Виконання та захист КППЗ – 80 балів 2. Виконання завдань під час тренінгу – 20 балів	1. Тестові завдання (30 тестів по 2 бали за тест) – 60 балів 2. Завдання 1 – 20 балів 3. Завдання 2 – 20 балів	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.	Електронний варіант лекцій	1-10
2.	Методичні рекомендації до виконання лабораторних робіт.	1-10

Рекомендовані джерела інформації

1. Buck I. Brook for GPUs: Stream Computing on Graphics Hardware / Ian Buck, Tim Foley, Daniel Horn, Jeremy Sugerman, Kayvon Fatahalian, Mike Houston, Pat Hanrahan // ACM Trans Graph. – ACM, 2014. – №3 (23). – P. 777-786.
2. Farber R. CUDA Application Design and Development /Rob Farber. – Morgan Kaufmann Publishers, 2019. – 400 p.
3. Fujimoto N. Faster matrix-vector multiplication on geforce 8800gtx / N. Fujimoto // Proceeding of IEEE Parallel and Distributed Processing. – 2018. – P. 1-8.
4. Garland M. Parallel computing experiences /Michael Garland, Scott Le Grand, John Nickolls, Joshua Anderson, Jim Hardwick, Scott Morton, Everett Phillips, Yao Zhang, Vasily Volkov // IEEE Micro. - Santa Clara, 2018. – №4 (28). – P. 13-27.
5. Nickolls J. The GPU Computing Era / John Nickolls, William J. Dally // IEEE Micro. – Santa Clara, 2016. – №2 (30). – P. 56-69.
6. Kirk D. Programming Massively Parallel Processors: A Hands-on Approach / David Kirk, Wen-mei Hwu. - Morgan Kaufmann, 2015. – 280 p.
7. Lindholm E. NVIDIA Tesla: A Unified Graphics and Computing Architecture / Erik Lindholm, John Nickolls, Stuart Oberman, John Montrym // IEEE Micro. – 2018. – Vo 28(2). – P. 39-55.
8. Pacheco P. An Introduction to Parallel Programming /Peter Pacheco. – Elsevier Science, 2016. – 392 p.
9. NVIDIA CUDA Compute Unified Device Architecture, Programming Guide, Version 2.0. – 2018. –107 p.
10. Peters H. Efficiently Using a CUDA-enabled GPU as Shared Resource /H. Peters, Koper, N. Luttenberger, //Proceedings of the 2010 10th IEEE International Conference on Computer and Information Technology. – 2017. – P. 1122-1127.