

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

**ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ
КАФЕДРА МАТЕМАТИЧНОЇ ІНФОРМАТИКИ**

«ЗАТВЕРДЖУЮ»
Заступник декана
з навчальної роботи
Олена КАШПУР
« 7 » *листопада* 20*24* року

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
РОЗПОДІЛЕНЕ ТА ПАРАЛЕЛЬНЕ
ПРОГРАМУВАННЯ**

для студентів

галузь знань **12 «Інформаційні технології»**
спеціальність **121 «Комп'ютерні науки»**
освітній рівень **бакалавр**
освітня програма **«Інформатика»**
вид дисципліни **обов'язкова**

Форма навчання	денна
Навчальний рік	2023/2024
Семестр	8
Кількість кредитів ECTS	3
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Викладачі: к.ф.м.н., доцент Деревянченко О.В. (лекції, лабораторні заняття).


Пролонговано: на 20 /20 н.р. () « » 20 р.
на 20 /20 н.р. () « » 20 р.

КИЇВ – 2021

Розробник: Олександр ДЕРЕВ'ЯНЧЕНКО, кандидат фіз.-мат. наук, доцент кафедри математичної інформатики

ЗАТВЕРДЖЕНО

Зав. кафедри «Математичної інформатики»

 Василь ТЕРЕЩЕНКО


Протокол № 10 від «27» 07 2024 р.

Схвалено гарантом освітньо-професійної програми «Інформатика»

« 6 » травня 2024 р.  Людмила ОМЕЛЬЧУК

Схвалено науково-методичною комісією факультету комп'ютерних наук та кібернетики

Протокол від « 6 » травня 2024 року № 10

Голова науково-методичної комісії  Людмила ОМЕЛЬЧУК

1. Мета дисципліни «РОЗПОДІЛЕНЕ ТА ПАРАЛЕЛЬНЕ ПРОГРАМУВАННЯ» є важливою дисципліною в сенсі набуття теоретичних та практичних знань в одній з найактуальніших на сьогодні галузей інформаційних технологій, що стосується програмування складних за обсягом обчислень задач з застосуванням багатоядерних комп'ютерів та розподілених середовищ (комп'ютерні мережі, хмари, кластери і т.п.).

2. Попередні вимоги до опанування або вибору навчальної дисципліни:

1. *Успішне опанування курсів «Програмування» та «Об'єктно-орієнтоване програмування».*
2. *Знати теоретичні основи теорії алгоритмів та математичної логіки.*
3. *Володіння технологіями програмування, основні відомості з програмування на мовах C++ та JAVA.*

3. Анотація навчальної дисципліни:

Навчальна дисципліна «РОЗПОДІЛЕНЕ ТА ПАРАЛЕЛЬНЕ ПРОГРАМУВАННЯ» є складовою освітньо-професійної програми підготовки фахівців за освітньо-кваліфікаційним рівнем «бакалавр» галузі знань 12 «Інформаційні технології» спеціальності 122 «Комп'ютерні науки». Метою і завданням навчальної дисципліни є ознайомлення з одним із основних науково-технологічних напрямків у галузі комп'ютерних технологій «РОЗПОДІЛЕНЕ ТА ПАРАЛЕЛЬНЕ ПРОГРАМУВАННЯ» та оволодіння технологіями паралельної та розподіленої обробки великих об'ємів даних.

Викладається у 8 семестрі в **обсязі – 90 годин.**

(3 кредитів ECTS) зокрема: *лекції – 28 год., консультації – 2 год., самостійна робота – 60 год.* Завершується дисципліна **іспитом в 8 семестрі.**

4. Завдання (навчальні цілі):

набуття знань, умінь та навичок (компетентностей) на рівні новітніх досягнень у програмуванні та обробці великих об'ємом даних, відповідно освітньої кваліфікації «Бакалавр інженерії програмного забезпечення». Зокрема, розвивати:

- здатність застосовувати знання у практичних ситуаціях;
- здатність реалізовувати високопродуктивні обчислення на основі хмарних сервісів і технологій, паралельних і розподілених обчислень при розробці й експлуатації розподілених систем паралельної обробки інформації.

5. Результати навчання за дисципліною:

Результат навчання (1. знати; 2. вміти; 3. комунікація; 4. автономність та відповідальність)		Форми (та/або методи і технології) викладання і навчання	Методи оцінювання та пороговий критерій оцінювання (за необхідності)	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
РН1.1	Знати архітектури систем паралельних обчислень та парадигми паралельного програмування.	<i>Лекція, самостійна робота, іспит</i>	<i>Контрольна робота, 60% правильних відповідей</i>	20%
РН1.2	Знати концепція керуючого простору та алгоритмічного модулю та ПАРКС-технології програмування.	<i>Лекція, самостійна робота, іспит</i>	<i>Контрольна робота, 60% правильних відповідей</i>	20%
РН1.3	Знати засоби, методи та технології паралельного програмування, а само: MPI, OpenMP та ПАРКС.	<i>Лекція, самостійна робота, іспит</i>	<i>Контрольна робота, 60% правильних відповідей</i>	20%
РН2.1	Вміти оцінювати і вибирати необхідні методи та технології паралельного програмування для вирішення поставленої задачі	<i>Лабораторне заняття, самостійна робота</i>	<i>Захист лабораторної роботи</i>	30%
РН3.1	Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програм, складати письмові звіти	<i>Лекція, самостійна робота</i>	<i>Захист лабораторної роботи</i>	5%
РН4.1	Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість	<i>Лекція, самостійна робота</i>	<i>Захист лабораторної роботи</i>	5%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Програмні результати навчання	Результати навчання дисципліни					
	РН 1.1	РН 1.2	РН 1.3	РН 2.1	РН 3.1	РН 4.1
<i>(з опису освітньої програми)</i>						
ПРН17. Виконувати паралельні та розподілені обчислення, застосовувати чисельні методи та алгоритми для паралельних структур, мови паралельного програмування при розробці та експлуатації паралельного та розподіленого програмного забезпечення.	+	+	+	+	+	+

7. Схема формування оцінки.

7.1. Форми оцінювання студентів:

- семестрове оцінювання:

1. *Контрольна робота 1:* РН1.1 – 10 балів/6 бали;

2. *Контрольна робота 2:* РН1.2, РН1.3 – 10 балів/6 балів;

3. *Виконання самостійної лабораторної роботи 1:* РН1.3, РН2.1, РН3.1, РН4.1 – 20 балів/12 балів;

4. Виконання самостійної лабораторної роботи 2: РН1.3, РН2.1, РН3.1, РН4.1 – 20 балів/12 балів.

- підсумкове оцінювання (у формі іспиту):

- максимальна кількість балів які можуть бути отримані студентом: 40 балів;

- результати навчання які будуть оцінюватись: РН1.1, РН1.2, РН1.3;

- форма проведення і види завдань: письмова робота.

Види завдань: 4 теоретичних запитань максимально по 10 балів за кожне.

Критерії оцінювання відповіді студента на теоретичне питання:

- повнота розкриття питання 1-8 бали;
- логіка викладення 1 бал;
- аналітичні міркування 1 бали.

Студент не допускається до іспиту, якщо під час семестру набрав менше ніж 20 балів.

Запитання для підготовки до іспиту

1. Що таке паралельна обчислювальна система?
2. Архітектура ПОС за Флінном.
3. Які існують показники продуктивності паралельних обчислень?
4. Що таке природний паралелізм обчислень задач?
5. Чим відрізняються парадигми паралельного програмування?
6. Які ви знаєте параметри архітектури мультипроцесорних систем?
7. Які існують шляхи для уніфікації та переносимості паралельного програмного забезпечення ?
8. Які види паралелізму використовуються в архітектурі паралельних ЕОМ?
9. Яка концепція керуючого простору та алгоритмічного модулю?
10. Що таке ПАРКС-технологія програмування?
11. Розподіл ресурсів та алгоритми диспетчеризації.
12. Засоби робота з системою ПАРКС-JAVA.
13. Застосування технології OpenMP.
14. Специфікації OpenMP для мови C++.
15. Засоби покращення якості паралельних програм з застосування технології OpenMP.
16. Застосування технології MPI.
17. Які основні поняття MPI?
18. Застосування кластерів для обчислення паралельних програм за технологією MPI.
19. Порівняння різних підходів паралельного програмування.
20. Які існують технології паралельного програмування?

7.2. Організація оцінювання:

Обов'язковим для заліку є виконання усіх контрольних робіт та лабораторної роботи до вказаної викладачем дати, перед початком екзаменаційної сесії, згідно навчального плану. Переписування чи перескладання тем не практикується. Дозволяється здача окремих завдань тем у проміжках між написанням контрольних робіт (наприклад, перша тема здається до здачі наступної контрольної роботи у будь-який зручний для викладача та студента час).

Терміни проведення форм оцінювання:

1. Виконання завдань, винесених на самостійну роботу: протягом семестру;
2. Контрольна робота 1: до 3 тижня семестру;
3. Контрольна робота 2: до 13 тижня семестру;
4. Виконання самостійної лабораторної роботи 1: до 7 тижня семестру.;
5. Виконання самостійної лабораторної роботи 2: до 11 тижня семестру.

У разі неякісного виконання лабораторної роботи, викладач має право не зарахувати лабораторну роботу, або знизити за неї бали.

Студент має право здавати лабораторну роботу та доповідь після закінчення визначеного для них терміну, але з втратою одного балу за кожен тиждень, який пройшов з моменту закінчення терміну її здачі.

Для студентів, які набрали сумарно меншу кількість балів ніж *критично-розрахунковий мінімум* – 21 бал для одержання іспиту за рішенням кафедри не допустити до складання іспиту із рекомендацією здати контрольну роботу та лабораторну роботу до повторного складання іспиту.

У випадку відсутності студента з поважних причин відпрацювання та перездачі контрольних робіт здійснюються у відповідності до „Положення про організацію освітнього процесу у Київському національному університеті імені Тараса Шевченка”.

При визначенні оцінки визначальною є робота в семестрі.

Протягом семестру кожен студент повинен виконати лабораторні роботи та звіт за ними. Всі роботи мають однакову складність і відрізняються лише предметною областю, яку студент обирає на власний смак і узгоджує з викладачем. На кожного студента дві лабораторні роботи. Сумарна оцінка за практичний модуль складає $20 \cdot 2 = 40$ балів.

Практичний модуль

Вимоги до лабораторних робіт:

Завдання

Обирається з класу задач, що потребує паралельної та розподіленої обробки. Студент обирає предметну область на власний смак і узгоджує з викладачем.

Дані

1) Всі дані повинні вводитись з попередньо згенерованого файлу, необхідно мати кілька наборів даних(файлів) для тестування.

Реалізація та Тестування

Для реалізації поставленої задачі обирається Завдання, яке спочатку треба реалізувати «послідовно» за допомогою однієї з мов програмування C++, Java та ін., а потім із застосуванням технологій програмування: MPI (Лабораторна робота 1) та OpenMP (Лабораторна робота 2).

Тестування треба робити для різних Данних та різної кількості ядер (воркерів).

Звіт

Роздруковувати не потрібно інформацію записуємо на Google Диск Вашої групи в персональну папку.

В звіті мають бути наступні розділи:

Опис завдання – алгоритм, що реалізує поставлену задачу.

Посилання на реалізацію(код) – репозиторій <https://github.com/> або інший з відкритим доступом.

Вимоги до користувача – опис засобів запуску та результати тестування програми: заносимо у відповідну таблицю.

У *Висновках* потрібно порівняти різні підходи до реалізації Вашої Задачі в Лабораторній роботі 1 та 2.

Робота здається в робочому стані, всі складові мають нормально працювати; крім цього, викладач буде пропонувати провести тестування та перевіряти розуміння аспектів реалізації(щодо технології, яка використовується) та алгоритму поставленого Завдання.

7.3. Шкала відповідності

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

8. Структура навчальної дисципліни. Тематичний план лекцій і лабораторних занять

№ лекції	Назва лекції	Кількість годин		
		Лекції	Лабор. заняття	Самост. робота
Частина I. Теоретичні аспекти паралельного програмування				
1	ТЕМА 1. Вступна частина. Поява та розвиток паралельних обчислювальних систем. Архітектури систем паралельних обчислень. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 1.	2		4
2	ТЕМА 2. Парадигми паралельного програмування. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 1.	1		4
Контрольна робота 1		1		
Частина II. Програмування на кластері (MPI)				
3	ТЕМА 3. Передача та приймання повідомлень між процесами. Групи та комунікатори. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 1.	2		4
4	ТЕМА 4. Віртуальні топології та передача даних. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 1.	2		4
5	ТЕМА 5. Застосування кластерів для обчислення паралельних програм за технологією MPI. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 1.	2		4
6	ТЕМА 6. Засоби покращення якості паралельних програм з застосування технології MPI. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 1.	1		8
Лабораторна робота 1		1		
Частина III. Програмування на багатоядерних комп'ютерних системах (OpenMP)				
7	ТЕМА 7. Директиви, Runtime-процедури та змінні середовища. Специфікації OpenMP для мови C++. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 2.	2		4
8	ТЕМА 8. Розробка паралельних програм із застосуванням OpenMP. Засоби покращення якості паралельних програм з застосування технології OpenMP. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 2.	3		8
Лабораторна робота 2		1		
Частина IV. PARCS-технологія програмування				

9	ТЕМА 9. Концепція керуючого простору та алгоритмічного модулю. ПАРКС-технологія програмування. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 2.	2		4
10	ТЕМА 10. Розподіл ресурсів та алгоритми диспетчеризації. Робота з системою ПАРКС-JAVA на комп'ютерній мережі. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 2.	2		8
11	ТЕМА 11. Огляд засобів ПАРКС для обчислень на Cloud та GPU. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 2.	2		4
12	ТЕМА 12. Порівняння різних підходів паралельного програмування. <i>Самостійна робота:</i> опрацювання лекційного матеріалу. Виконання лабораторної роботи 2.	3		4
Контрольна робота 2		1		
ВСЬОГО		28		60

Загальний обсяг 90 годин, в тому числі:

Лекцій – 28 год.

Консультації – 2 год.

Самостійна робота – 60 год.

9. Рекомендовані джерела:

Основні:

1. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления // БХВ, Петербург, 2002, 608 с.
2. Системы параллельной обработки. // Под ред. Д. Ивенса. М., Мир, 1985, 416 с.
3. Технології паралельного програмування - [Електронний ресурс] – Режим доступу до ресурсу: <http://parallel.ru/tech/>
4. Глушков В.М., Анисимов А.В. Управляющие пространства в асинхронных параллельных вычислениях. // Кибернетика, 1980, №5, С.1-9.
5. Анисимов А.В., Деревянченко А.В. Система ПАРУС-JAVA для параллельных вычислений на компьютерных сетях // Кибернетика и системный анализ, 2005, №1, С.25-36.
6. Деревянченко О.В. Моделирование параллельных программ за допомогою системи ПАРКС-JAVA // Наукові записки НаУКМА, Комп'ютерні науки. 2005, Т.36., С.32-38.
7. В. Eckel, Thinking in Java // Prentice Hall, 1999, 859 p.
8. Белицкий Р.И., Логинов В.П. Выбор алгоритма диспетчеризации для мультимикропроцессорной системы с общей шиной // Управляющие системы и машины, 1991, №1, С. 9–13.
9. Деревянченко А.В. Алгоритм оптимизации распределения заданий для решения параллельных задач в неоднородной мультимикропроцессорной системе // Материалы VI международного научно-практического семинара. 2006, Т.1, С.175-180.
10. Технологія MPI - [Електронний ресурс] – Режим доступу до ресурсу: <http://www.mpi-forum.org/>
11. Технологія OpenMP - [Електронний ресурс] – Режим доступу до ресурсу: <http://www.openmp.org/>
12. Деревянченко О.В. ПАРКС-JAVA система для параллельних обчислень на комп'ютерних мережах // Навчальний посібник, Київ, 2011 р., 60 с. [Електронний ресурс] – Режим доступу до ресурсу: <http://www.csc.knu.ua/en/library/books/derevianchenko-35.pdf>

Додаткові:

1. S. Akhter, J. Roberts. Multi-Core Programming. — Intel Press, 344p.
2. Richard Gerber, Aart J.C. Bik, Kevin B. Smith, and Xinmin Tian The Software Optimization Cookbook, Second Edition — Intel Press, 404p.
3. С.А. Немнюгин Методические материалы по курсу "Средства программирования для многопроцессорных вычислительных систем"- [Електронний ресурс] – Режим доступу до ресурсу: http://www.phys.spb.ru/content/File/Library/studentlectures/Nemnugin/Metod_Nemnugin_Intel.pdf
4. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. М., Мир, 1979.
5. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ. М., МЦНМО, 2000., 960 с.