


Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Кафедра «Інформаційних технологій проектування» (№ 105)

ЗАТВЕРДЖУЮ

Голова НМК 2

 Д.М. Крицький
« 31 » 08 2021 року

**РОБОЧА ПРОГРАМА *ОБОВ'ЯЗКОВОЇ*
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

**ТЕХНОЛОГІЇ РОЗПОДІЛЕНИХ СИСТЕМ
ТА ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ**
(назва навчальної дисципліни)

Галузь знань: 12 «Інформаційні технології»
(шифр і назва галузі знань)

Спеціальність: 122 «Комп'ютерні науки»
(шифр і назва напрямку підготовки)

Освітня програма: «Інформаційні технології проектування»
(найменування освітньої програми)

Форма навчання: денна

Рівень вищої освіти: перший (бакалаврський)

Харків 2021

Робоча програма «Технології розподілених систем та паралельних обчислень»

(назва дисципліни)

для студентів за спеціальністю 122 «Комп'ютерні науки»

освітньою програмою «Інформаційні технології проектування»

31.08.2021 р., – 9 с.

Розробник: Каратанов О. В., доцент кафедри 105 «Інформаційних технологій проектування», к.т.н.

(прізвище та ініціали, посада, наукова ступінь та вчене звання)


(підпис)

Робочу програму розглянуто на засіданні кафедри інформаційних технологій проектування

(назва кафедри)

Протокол № 1 від 31.08.2021 року

Завідувач кафедри 105, к.т.н., доцент

(наукова ступінь та вчене звання)


(підпис)

Д.М. Крицький
(ініціали та прізвище)

1 Опис навчальної дисципліни

Найменування показників	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни
Кількість кредитів – 6	Галузь знань 12 «Інформаційні технології» Спеціальність 122 «Комп'ютерні науки» Освітня програма «Інформаційні технології проектування» Рівень вищої освіти: перший (бакалаврський)	денна форма навчання
Модулів – 2		<i>Обов'язкова</i>
Змістових модулів – 2		Навчальний рік
Загальна кількість годин – 84 / 180		2021/2022
Тижневих годин для денної форми навчання: аудиторних – 5,25 самостійної роботи студента – 6		Семестр
		8-й / 6-й (скор. форм. навч.)
		Лекції*
	48 год.	
	Практичні, семінарські	
	0 год.	
	Лабораторні*	
	36 год.	
	Самостійна робота	
	96 год.	
	Вид контролю	
	іспит	

Співвідношення кількості годин ауд. занять до сам. роботи становить для денної форми $(48 + 36) / 96 = 0,875$.

*Аудиторне навантаження може бути зменшеним або збільшеним на одну годину в залежності від розкладу занять.

2 Мета та завдання навчальної дисципліни

Метою дисципліни є здобуття теоретичних знань та практичних навичок програмування складних, розподілених та навантажених систем за допомогою сучасних технологій та паралельних обчислень. Отримані у процесі навчання знання є важливою складовою для формування спеціалістів за фахом.

Основними **завданнями** дисципліни є отримання навичок розробки алгоритмів паралельних обчислень, програмування паралельних методів розв'язань рівнянь та перетворення арифметичних виразів за допомогою сучасних технологій, виконання віддалених викликів процедур та застосування методів.

У межах дисципліни студенти отримають теоретичні знання щодо типів паралелізму, базових понять розподілених систем, засобів підтримки паралельних обчислень і створення та управління кластерами та Grid-системами.

Згідно з вимогами освітньо-професійної програми студенти повинні досягти таких компетентностей:

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК6. Здатність вчитися й оволодівати сучасними знаннями.

ЗК11. Здатність приймати обґрунтовані рішення.

СК16. Здатність реалізовувати високопродуктивні обчислення на основі хмарних сервісів і технологій, паралельних і розподілених обчислень при розробці й експлуатації розподілених систем паралельної обробки інформації

Програмні результати навчання:

ПР5. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.

ПР17. 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.

У результаті вивчення навчальної дисципліни студент повинен

знати:

– архітектури та стандарти компонентних моделей, комунікаційних засобів та розподілених обчислень.

– засоби вирішення проблеми масштабованості та віддаленої підтримки;

– основні типи паралелізму;

– базові складові кластеру, Grid та розподілених систем;

– засоби підтримки паралельних обчислень;

– паралельні методи розв'язання математичних задач;

– топології та архітектури розподілених систем;

– моделі віддаленого виклику процедур та застосування методів;

– способи передавання даних у розподілених системах;

– базові поняття розподілених ресурсів та баз даних;

вміти:

– вирішувати проблеми масштабованості та підтримки розподілених систем;

– проектувати паралельні алгоритми математичних завдань;

– створювати багатопоточні програми;

– оцінювати ефективність паралельних обчислень;

– застосовувати технології віддалених викликів процедур та методів;

– використовувати різні типи паралелізму при вирішенні паралельних задач;

– проектувати базові складові розподілених систем;

– використовувати засоби підтримки паралельних обчислень.

Студент має надбати первинні навички розроблення паралельних програм та проектування розподілених систем, не залежно від предметної області використання та прикладного програмного забезпечення.

Міждисциплінарні зв'язки: вивчення цього курсу потребує базис з дисциплін, які націлені на вивчення програмування. Отримані навички з паралельних обчислень безумовно можуть бути використані при написанні програмного забезпечення до дипломного проекту.

3 Програма навчальної дисципліни

Модуль 1

Змістовний модуль 1. Основні поняття паралельних обчислень, Grid-системи та їх забезпечення, багатопоточне програмування

Тема 1. Призначення, задачі, функції паралельних обчислень

Історія паралельних обчислень (ПО). Поняття ПО, їх призначення. Завдання і функції паралельних обчислень. Галузі застосування ПО. Що таке процесор, які процесори використовуються при побудові розподілених систем та кластерів, види процесорів. Найсучасніші суперкомп'ютери, їх застосування та характеристики. Загальна схема побудови кластеру.

Тема 2. Поняття розподіленої системи, поняття та функції Grid-систем

Поняття розподіленої системи (РС). Функції та призначення РС. Архітектура РС, можливі труднощі проектування РС. Поняття Grid-системи, їх функції та призначення. Базові складові Grid-системи. Засоби підтримки та управління РС та Grid-системами. Безпека файлових систем, сертифікат відкритих ключів.

Тема 3. Введення у багатопоточне програмування

Поняття потоку, процесу, ядра процесору. Чим відрізняється процес та потік на рівні програми та операційної системи. Реалізації багато поточного програмування на мовах програмування C, C++ та Java. Базові алгоритми та підходи до створення одного та декількох потоків у програмі. Методи та алгоритми запуску, зупинки та паузи потоку. Опис видів та базових проблем, що виникають при розробленні багато поточних програм, методи та напрямки їх вирішення.

Тема 4. Синхронізація потоків, способи передавання даних

Поняття синхронізації потоків, коли і чому фона необхідна. Види складнощів, що виникають під час синхронізації декількох потоків. Поняття synchronized методів та атомарних дій. Механізм синхронізації потоків wait/notify. Алгоритми синхронізації потоків за допомогою Conditions та Locks. Поняття взаємного блокування (Deadlock). Поняття черги завдань, існуючі реалізації черг, блокуючи черга. Опис патерну програмування Producer-Consumer, реалізація цього патерну за допомогою черг. Поняття бар'єру, як вони використовуються для синхронізації потоків, циклічний бар'єр. Поняття пулу потоків, існуючі реалізації пулів, галузь їх застосування, фабрики пулів потоків.

Тема 5. Огляд апаратного забезпечення Grid-систем

Введення до базових понять апаратного забезпечення PC та Grid-систем. Поняття, вимоги, переваги та недоліки систем з розподіленою пам'яттю та зі спільною пам'яттю. Поняття мультипроцесорної системи. Функції та переваги використання GPU процесорів. Базова структура апаратного забезпечення кластеру. Методи оцінки швидкості виконання операцій на суперкомп'ютерах. Метод HPL – швидкість вирішення системи лінійних рівнянь. Гібридні системи, та їх структура.

Тема 6. Типи паралелізму, основи створення паралельних алгоритмів

Аналіз структури алгоритму та виявлення паралелізму. Модель паралельного алгоритму. Представлення паралельного алгоритму у вигляді графу, паралельна форма графу. Редукція висоти древа паралельного алгоритму. Усунення зв'язків алгоритму для забезпечення можливості паралельного виконання програми.

Тема 7. Введення у види паралелізму та показники ПО

Види паралелізму. Базові поняття алгоритмічного, геометричного (ітеративного та рекурсивного), конвеєрного та колективного паралелізму. Введення поняття стіни Фокса. Показники якості паралельного алгоритму. Прискорення та ефективність ПО. Закон Амдала. Основні етапи розробки паралельного алгоритму, вибір структури алгоритму. Декомпозиція завдань, організація взаємодії між завданнями. Розподіл завдань між процесорами.

Модульний контроль 1.

Модуль 2.

Змістовний модуль 2. Інструментальні засоби підтримки паралельних обчислень, віддалений виклик процедур

Тема 8. Засіб підтримки паралельних обчислень OpenMP

Введення в існуючі засоби підтримки ПО. Їх призначення, функції, можливості та типи. Знайомство з технологією підтримки ПО OpenMP. Модель програмування OpenMP. Поняття Fork-Join паралелізму. Базові складові створення паралельної програми за допомогою технології OpenMP. Використання директив компілятора для позначення паралельних секцій програми. Директиви компілятора parallel for, reduction, private. Паралелізація циклів, та виконання колективних операцій над даними. Поняття завдань у стандарті OpenMP 2.0. Зразки програм з використанням завдань.

Тема 9. Розроблення паралельних алгоритмів для вирішення математичних задач за допомогою OpenMP

Розгляд практичних аспектів використання технології підтримки ПО OpenMP та методик створення паралельних алгоритмів на базі математичних задач. Директиви компілятора ordered та schedule. Побудова паралельної версії програми розрахунку ряду Фібоначі. Розгляд підходів до вирішення звичайних диференціальних рівнянь (ЗДР) за допомогою технології OpenMP. Складнощі створення паралельних версій чисельних методів вирішення ЗДР методами Ейлера та Рунге-Кута.

Тема 10. Технологія паралельних обчислень MPI

Введення в технологію підтримки ПО Message Passing Interface (MPI). Огляд стандартів MPI 1 та MPI 2. Опис нововведень у MPI 2. Коротка історія розробки та створення MPI. Існуючі реалізації стандарту для різних мов програмування. Модель розроблення програм за допомогою

MPI. Поняття комунікатору. Функції MPI, загальна структура програми, компіляція та запуск. Розгляд способу комунікації вузлів розподіленої системи за допомогою MPI.

Тема 11. Технологія MPI: Синхронізація та способи передавання даних

Функції передачі даних за допомогою повідомлень MPI. Структура повідомлення. Стандартні блокуючі функції передачі даних MPI_Send та MPI_Recv. Базові типи даних MPI. Інші види блокуючих викликів при передачі та прийомі даних: MPI_Bsend, MPI_Ssend, MPI_Rsend. Модель блокуючої передачі та прийому даних. Які складнощі виникають при блокуючому прийомі та передачі даних. Ситуація взаємного блокування. Неблокуючі функції передачі та прийому даних: MPI_Isend, MPI_Ibsend, MPI_Issend, MPI_Irsend. Модель неблокуючого прийому та передачі даних. Сумісний прийом та передача даних MPI_Sendrecv. Неблокуюча перевірка повідомлень MPI_Iprobe.

Тема 12. Технологія MPI: колективні, глобальні обчислювальні операції над розподіленими даними

Поняття колективних та глобальних обчислювальних операцій над розподіленими даними. Їх реалізація у стандарті MPI. Основні типи колективних операцій та їх функції: обмін даними, колективні обчислювання, синхронізація. Деталізований розгляд функцій обміну даними: MPI_Bcast, MPI_Scatter, MPI_Gather, MPI_Allgather, MPI_alltoall. Деталізований розгляд функцій колективних обчислень: MPI_Reduce, MPI_Allreduce, MPI_Reduce_scatter, MPI_Scan. Колективні операції над розподіленими даними. Створення колективних операцій над даними. Зразки програм з використанням колективних обчислень у MPI. Синхронізація за допомогою Barrier.

Тема 13. Моделі віддаленого виклику процедур (RPC) та застосування методів

Введення в сучасні технології віддалених викликів процедур та застосування методів. Реалізація взаємодії розподілених компонентів системи від компанії Microsoft DCOM. Використання інтерфейсу, глобального унікального ідентифікатору та засобів виклику методів. Протокол доступу до віддалених об'єктів SOAP. Структура протоколу, SOAP повідомлення. Опис веб-сервісу за допомогою мови WSDL, віддалений доступ до методів веб-сервісу за допомогою SOAP. Введення у технологію REST API. Поняття CRUD операцій. Розгляд типових операцій над даними у REST API. Операції отримання даних, вставлення, редагування та видалення даних.

Тема 14. Технологія RPC для Java

Знайомство із реалізацією технології RPC для мови програмування Java. Бібліотека grjson4j. JSON як формат передачі віддалених викликів. Введення у JSON, опис формату, його переваги, правила створення JSON строки. Функції та призначення бібліотеки grjson4j. Модель створення серверу для прийому віддалених викликів та клієнту для відправлення викликів. Використання інтерфейсу у сервері та клієнті. Можливості використання різноманітних протоколів для передачі JSON повідомлень. Зв'язок із веб-технологіями.

Тема 15. Розподілені ресурси та бази даних

Введення у поняття розподілених ресурсів. Поняття бази даних та технологій розподілених баз даних. Недоліки реляційних баз даних. Альтернативні рішення NoSQL. Розгляд розподіленої бази даних Apache HBase. Концептуальна модель даних, фізична модель даних. Архітектура HBase кластеру. Розгляд розподіленої бази даних Apache Cassandra. Типи даних та індекси. Розподіл даних між вузлами. Стратегії розподілу даних. Огляд Cassandra Query Language. Знайомство із поняттям графової бази даних. Графова база даних Neo4j. Типи даних, операції, мова виконання запитів Cypher.

Тема 16. Підсумки дисципліни, організація та управління кластерними системами

Огляд та підсумок розглянутого у дисципліні матеріалу. Огляд основних підходів до організації та управління кластером. Зв'язок дисципліни із веб-технологіями та сітьовими технологіями. Сучасні напрямки розвитку ПО та РС. Повторення основних методик створення алгоритмів та програм ПО. Оцінка алгоритмів, їх рефакторинг. Засоби підтримки ПО OpenMP, MPI. Напрямки для самостійної роботи, технологія MapReduce. Висновки за дисципліною.

Модульний контроль 2.

4 Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
		л.	п. р.	лаб.	к. р.	с. р.
Модуль 1						
Змістовий модуль 1						
Тема 1	10	2	—	4	—	4
Тема 2	12	2	—	4	—	6
Тема 3	8	2	—	—	—	6
Тема 4	12	2	—	4	—	6
Тема 5	8	2	—	—	—	6
Тема 6	8	2	—	—	—	6
Тема 7	20	8	—	6	—	6
Модульний контроль 1	2	—	—	2	—	—
Разом за змістовим модулем 2	80	20	—	20	—	40
Модуль 2						
Змістовий модуль 2						
Тема 8	4	2	—	—	—	2
Тема 9	8	2	—	—	—	6
Тема 10	8	2	—	—	—	6
Тема 11	8	2	—	—	—	6
Тема 12	22	10	—	6	—	6
Тема 13	12	2	—	4	—	6
Тема 14	8	2	—	—	—	6
Тема 15	12	2	—	4	—	6
Тема 16	8	2	—	—	—	6
Тема 17	8	2	—	—	—	6
Модульний контроль 2	2	—	—	2	—	—
Разом за змістовим модулем 2	80	28	—	16	—	56
Усього годин за дисципліною	180	48	—	36	—	96

5 Теми семінарських занять

Семінарські заняття навчальним планом не передбачені.

6 Теми практичних занять

Практичні заняття навчальним планом не передбачені.

7 Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Розроблення багатопоточної програми	4
2	Синхронізація потоків (методи synchronized)	4
3	Реалізація патеру програмування Producer-Consumer	4
4	Використання пулів потоків при виконанні ПО	4
	Модульний контроль 1	4
5	Знайомство із засобом підтримки ПО OpenMP	4
6	Вирішення простих математичних задач за допомогою OpenMP	4
7	Вирішення ЗДР за допомогою OpenMP	4
	Модульний контроль 2	4
	Разом	36

У разі дистанційного навчання лабораторні заняття замінюються на практичні.

8 Самостійна робота

Самостійна робота у якості розрахунково-графічної роботи за курсом передбачає оформлення пояснювальної записки, де студент описує процес проектування алгоритму вирішення паралельних задач та розроблення паралельної програми.

Також самостійна підготовка включає підготовку до лабораторних робіт, модульного контролю та заліку.

№ з/п	Назва теми	Кількість годин
1	Розрахунково-графічна робота (домашня робота) за обраною темою	50
2	Підготовка до лабораторних робіт, модульного контролю та заліку	44
	Разом	99

9 Індивідуальне завдання

Зміст: Проектування алгоритму вирішення паралельних задач та розроблення паралельної програми. Варіанти завдань – паралельні математичні задачі, та задачі одночасного обслуговування тощо. Форма звітності - подача пояснювальної записки та демонстрація функціонування ПЗ.

Обсяг звітних документів: 10–20 сторінок. Трудомісткість: 30 годин самостійної роботи.

План-графік виконання ІДЗ:

№	Найменування розділу	Обсяг, %	Тиждень, задачі	Кількість сторінок ПЗ	Трудомісткість	
					аудитори.	самостійн.
1	Постановка завдання	10	3	1-2	-	2
2	Розроблення алгоритму вирішення паралельної задачі	10	5	1-2	-	10
3	Розроблення паралельної програми	50	7	7-12	-	28
4	Тестування паралельної програми, оцінка прискорення та ефективності розробленої програми	30	16	2-6		10
Разом		100		11-20	-	50

10 Методи навчання

Лекції проводяться з використанням основних розділів конспекту лекцій в електронній формі, елементів мультимедійної підтримки курсу (презентації), демонстрацій окремих прийомів роботи з інструментальним середовищем.

Лабораторні роботи виконуються з використанням навчальних (демонстраційних) та ліцензованих робочих версій ПЗ (Eclipse, Microsoft Visual Studio, MPJ Express).

Самостійна робота включає підготовку до лабораторних робіт, модульного контролю та заліку, виконання розрахунково-графічної роботи і вивчення вказаних вище тем за конспектом, літературними джерелами та програмною документацією.

11 Методи контролю

Поточний контроль здійснюється відповідно до повноти, своєчасності та якості виконання лабораторних робіт та домашнього завдання (розрахунково-графічної роботи).

Проміжний (модульний) контроль – письмова контрольна робота на 7 та 14-му тижні.

Підсумковий контроль – письмовий залік та усні відповіді на запитання.

12 Критерії оцінювання та розподіл балів, які отримують студенти

12.1 Розподіл балів, які отримують студенти

Складові навчальної роботи	Бали за одне завдання (завдання)	Кількість занять (занядань)	Сумарна кількість балів
Модульні контрольні роботи	0–50	2	0–50
Лабораторні роботи	1–7	7	7–30
Розрахунково-графічна робота	1–15	1	1–10
Робота на лекціях	0–1	10	0–10
Усього за семестр			8–100

Семестровий контроль (іспит/залік) проводиться у разі відмови студента від балів поточного тестування та за наявності допуску до іспиту/заліку. При складанні семестрового іспиту/заліку студент має можливість отримати максимум 100 балів.

Білет для іспиту/заліку складається з 5 запитань: 3 теоретичних та 2 практичних, по 20 балів за кожне.

12.2 Якісні критерії оцінювання

Проміжна атестація включає в себе теоретичні завдання, що дозволяють оцінити рівень засвоєння учнями знань, і практичні завдання, що виявляють ступінь сформованості умінь і володіння.

Засвоєні знання і освоєння вміння перевіряються за допомогою модулів, вміння і володіння перевіряються в ході вирішення завдань.

Обсяг і якість освоєння учнями дисципліни, рівень сформованості дисциплінарних компетенцій, оцінюються за результатами поточних і проміжної атестацій кількісною оцінкою, вираженою в балах, максимальна сума балів з дисципліни дорівнює 100 балам.

12.3 Критерії оцінювання роботи студента протягом семестру

Сума балів, набраних студентом з дисципліни, переводиться в оцінку відповідно до таблиці.

Сума балів з дисципліни	Оцінка за проміжної атестації	Характеристика рівня освоєння дисципліни
від 90 до 100	«зараховано» / «відмінно»	Студент демонструє сформованість дисциплінарних компетенцій на підсумковому рівні, виявляє всебічне, систематичне і глибоке знання навчального матеріалу, засвоїв основну літературу і знайомий з додатковою літературою, рекомендованою програмою, вміє вільно виконувати практичні завдання, передбачені програмою, вільно оперує набутими знаннями, вміннями, застосовує їх у ситуаціях підвищеної складності.
від 75 до 89	«зараховано» / «добре»	Студент демонструє сформованість дисциплінарних компетенцій на середньому рівні: основні знання, вміння освоєні, але допускаються незначні помилки, неточності, труднощі при аналітичних операціях, перенесення знань і умінь на нові, нестандартні ситуації.
від 60 до 74	«зараховано» / «задовільно»	Студент демонструє сформованість дисциплінарних компетенцій на базовому рівні: в ході контрольних заходів допускаються значні помилки, виявляється відсутність окремих знань, умінь, навичок за деякими дисциплінарними компетенціями, студент відчуває значні труднощі при оперуванні знаннями та вміннями при їх перенесенні на нові ситуації.
від 41 до 59	«не зараховано» / «незадовільно»	Студент демонструє сформованість дисциплінарних компетенцій на рівні нижче базового, проявляється недостатність знань, умінь, навичок.
від 0 до 40	«не зараховано» / «незадовільно»	Дисциплінарні компетенції не формувати. Виявляється повне або практично повна відсутність знань, умінь, навичок.

Шкала оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

13 Методичне забезпечення

Навчально-методичний комплекс дисципліни у електронному вигляді знаходиться на кафедральних серверах та на гугл-диску – Каратанов О. В. Навчально-методичний комплекс дисципліни [Електронний ресурс] / О. В. Каратанов. – Х. : ХАІ, 2021. – Режим доступу : <https://drive.google.com/file/d/1ISMiCpzhb40tsxVpkgQSIKAJ8hYLCbyR/>

- робоча програма дисципліни;
- конспект лекцій, підручники (навчальні посібники), в тому числі в електронному вигляді, які за змістом повністю відповідають робочій програмі дисципліни;
- методичні вказівки та рекомендації для виконання розрахунково-графічних робіт, лабораторних робіт, а також рекомендації для самостійної підготовки;
- тематики індивідуальних завдань;
- приклади розв'язування типових задач чи виконання типових завдань;
- питання, тести для контрольних заходів;
- каталоги інформаційних ресурсів.

14 Рекомендована література

14.1 Базова

1. Аксак Н.Г. Паралельні та розподілені обчислення: підруч. / Н. Г. Аксак, О. Г. Руденко, А.М.Гуржій. – Х. : Компанія СМІТ, 2009. – 480 с.
2. Эндрюс Г.Р. Основы многопоточного, параллельного и распределенного программирования / Эндрюс Г.Р. – М. : «Вильямс», 2003. – 512 с.
3. Богачев К.Ю. Основы параллельного программирования. /Богачев К.Ю. – М. : БИНОМ. Лаборатория знаний, 2003.
4. Фельдман Л. П., Назарова І.А. Паралельні однокрокові методи чисельного розв'язання задачі Коші. Донецьк: «ДВНЗ» ДонНТУ, 2011. – 185 с.: іл.
5. М.В. Якововский. Распределенные системы и сети. Учебное пособие. – М. : МГТУ «Станкин», 2000. – 118 с., ил.
6. Дорошенко А.Ю. Лекції з паралельних обчислювальних систем. Київ : Видавничий дім “КМ Академія”, 2003. – 42 с.

14.2 Допоміжна

1. Ammar Ahmad Awan, MB Amin, S Hussain, A Shafi, Sungyoung Lee. An MPI-IO compliant Java Based Parallel I/O Library, 13th Cluster, Cloud and Grid Computing (CCGrid) 2013.
2. Aamir Shafi , Jawad Manzoor, Kamran Hameed, Bryan Carpenter, and Mark Baker, Multi-core-enabling the MPJ Express Messaging Library, 8th International Conference on the Principles and Practice of Programming in Java (PPPJ), September 15-17 2010, Vienna, Austria.
3. Г.Ю. Сісюк, О.А. Дурницька. Методичні вказівки "Паралельне програмування" до лабораторних робіт для студентів зі спеціальностей 7.091501«Комп'ютерні системи та мережі» денної форми навчання. Кременчук: КНУ, 2003. – 19с.

4. Берченко М.М., Березовська І.Б. Багатоядерні процесори: мікроархітектура та особливості застосування. Тернопіль: ТНТУ, 2011.
5. Эккель Б. Философия Java. — СПб. : Питер, 2001.
6. Блэк Ю. Сети ЭВМ: протоколы, стандарты, интерфейсы. — М. : Мир, 1990.
7. Вязовик Н. А. Программирование на Java. — М. : Интернет-ун-т информационных технологий — ИНТУИТ.РУ, 2003.
8. Дейтел Х. М. и др. Технологии программирования на Java 2. — М. : Бином-пресс, 2003. — Кн. 3. Корпоративные системы, сервлеты, JSP, Web-сервисы.
9. Джонс Э., Олант Д. Программирование в сетях Microsoft Windows. — СПб. : Питер, 2002.
10. Как программировать на Java. — М. : Бином-пресс, 2003. — Кн. 1. Основы программирования.
11. Перроун П. и др. Создание корпоративных систем на основе Java 2 Enterprise Edition. Руководство разработчика. — М. : Изд. Дом “Вильямс”, 2001.
12. Хорстманн К. С., Корнелл Г. Библиотека профессионала: Java 2. — М. : Изд. Дом “Вильямс”, 2004. — Т. I. Основы. — 848 с.; Т. II. Тонкости программирования. — 1120 с.

15 Інформаційні ресурси

1. Паралельні та розподілені обчислення [Електронний ресурс]. – Режим доступу: <http://shad.yandex.ru/lectures/computing.xml>.
2. Кластер [Електронний ресурс]. – Режим доступу: [http://uk.wikipedia.org/wiki/Кластер_\(Інформатика\)](http://uk.wikipedia.org/wiki/Кластер_(Інформатика)).
3. Web Service Modelling Ontology [Електронний ресурс]. – Режим доступу: <http://www.wsmo.org/>.
4. MPJ Express [Електронний ресурс]. – Режим доступу: <http://mpj-express.org>.