

**ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ АВТОМОБІЛЬНО-ДОРОЖНИЙ
УНІВЕРСИТЕТ
СИСТЕМА УПРАВЛІННЯ ЯКІСТЮ**

**Система забезпечення якості освітньої діяльності та якості вищої освіти
сертифікована на відповідальність ДСТУ ISO 9001:2015/ ISO 9001:2015**

Кафедра комп'ютерних технологій і мехатроніки

ЗАТВЕРДЖЕНО

Гарант освітньо-професійної програми
«Інженерія програмного забезпечення»
першого(бакалаврського) рівня освіти:

зав. каф. КТМ, д.т.н., проф.


Ніконов О.Я.

**СИЛАБУС
ПРОГРАМУВАННЯ РОЗПОДІЛЕНИХ СИСТЕМ ТА
ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ//
DISTRIBUTED SYSTEMS AND PARALLEL COMPUTING
SYLLABUS**

освітній ступінь	бакалавр / bachelor
галузь знань	12 Інформаційні технології / Information Technology
спеціальність	121 Інженерія програмного забезпечення / Software Engineering

Харків 2020

Розповсюдження та тиражування без офіційного дозволу ХНАДУ заборонено.

Автор: Ніконов Олег Якович, професор кафедри комп'ютерних технологій і мехатроніки.

Силабус розглянуто та затверджено на засіданні кафедри комп'ютерних технологій і мехатроніки, протокол № 1 від «03» вересня 2020 р.

**СИЛАБУС
ТЕХНОЛОГІЇ РОЗПОДІЛЕНИХ СИСТЕМ ТА
ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ//
DISTRIBUTED SYSTEMS AND PARALLEL COMPUTING
TECHNOLOGIES
SYLLABUS**

освітній ступінь	бакалавр / bachelor
галузь знань	12 Інформаційні технології / Information Technology
спеціальність	121 Інженерія програмного забезпечення / Software Engineering

Анотація курсу

1. Викладачі

Лектор: Ніконов Олег Якович

- професор кафедри комп'ютерних технологій і мехатроніки;
- педагогічний стаж – 23 роки
- контактний телефон +38-057-707-37-43
- e-mail:
- наукові інтереси: комп'ютерні мережі, інформаційні технології.

2. Дисципліна «Програмування розподілених систем та паралельних обчислень»

- рік навчання: 3;
- семестр навчання: 6;
- кількість кредитів та годин за семестр: 4 кредити/120 годин, в т. ч.
 - лекційних: 16;
 - практичних занять: 32;
 - на самостійне опрацювання: 72;
- кількість аудиторних годин на тиждень:
 - лекційних: 2 (раз на два тижні);
 - практичних занять: 2 (раз на тиждень у кожній групі потоку).

3. Час та місце проведення

- аудиторні заняття – відповідно до розкладу ХНАДУ, ауд. 314;
- позааудиторна робота – самостійна робота студента із використанням технологій віртуалізації Oracle, хмарних технологій Google та Microsoft.

4. Пререквізити та постреквізити навчальної дисципліни:

- **пререквізити:** “Алгоритмізація та програмування”, “Об’єктно-орієнтоване програмування”, “Мережеві технології та системне адміністрування” ;
- **постреквізити** (дисципліни та компетентності, які необхідні в професійній діяльності фахівця): дисципліна «Технології розподілених систем та паралельних обчислень» є базовою для вивчення таких спеціальних дисциплін як «Емпіричні методи програмної інженерії», «Web-програмування», «Людино-машинна взаємодія», дипломне проектування.

5. Характеристика дисципліни:

Програма вивчення навчальної дисципліни «**Технології розподілених систем та паралельних обчислень**» складена відповідно до освітньо-кваліфікаційної характеристики та навчального плану підготовки **бакалавра** спеціальності **121 Інженерія програмного забезпечення**.

5.1. Призначення навчальної дисципліни полягає в наступному. Потреба вирішення складних прикладних задач з великим об'ємом обчислень і принципова обмеженість максимального швидкодії "класичних" - за схемою

фон Неймана - ЕОМ привели до появи багатопроцесорних обчислювальних систем (БОС). Використання таких засобів обчислювальної техніки дозволяє істотно збільшувати продуктивність ЕОМ при будь-якому існуючому рівні розвитку комп'ютерного обладнання. При цьому, однак, необхідно "паралельне" узагальнення традиційної - послідовної - технології вирішення завдань на ЕОМ. Так, чисельні методи в разі БОС повинні проектуватися як системи паралельних і взаємодіючих між собою процесів, що допускають виконання на незалежних процесорах. Застосовувані алгоритмічні мови та системне програмне забезпечення повинні забезпечувати створення паралельних програм.

Предметом розгляду даного курсу і є вивчення перерахованого кола питань.

5.2. Метою вивчення дисципліни «Технології розподілених систем та паралельних обчислень» є формування у студентів сукупності знань та навичок щодо типових підходів, методів та технологій паралельних обчислень в комп'ютерних і розподілених системах на рівні професійних вимог зі спеціальності, розвитку системного підходу що до проектування розподілених інформаційних.

5.3. Задачі вивчення дисципліни: теоретична та практична підготовка майбутніх фахівців передбачає практичне освоєння наступних тем теорії розподілених систем та паралельного програмування:

- архітектурні принципи реалізації паралельної обробки в обчислювальних машинах;
- методи і мовні механізми конструювання паралельних програм;
- паралельні обчислювальні методи.

Вивчення перелічених тем досягається наявністю в цьому курсі наступного набору розділів:

- цілі і завдання паралельної обробки даних;
- принципи побудови паралельних обчислювальних систем;
- моделювання та аналіз паралельних обчислень;
- принципи розробки паралельних алгоритмів і програм;
- системи розробки паралельних програм;
- паралельні чисельні алгоритми для вирішення типових задач обчислювальної математики.

Очікувані результати навчання з дисципліни

Професійні компетентності, які отримують студенти після вивчення навчальної дисципліни:

Інтегральна компетентність:

здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій.

Загальні компетентності:

здатність застосовувати знання у практичних ситуаціях;
здатність спілкуватися державною мовою як усно, так і письмово;
здатність вчитися і оволодівати сучасними знаннями.

Фахові компетентності

здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних;

здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення;

здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.

Програмні результати навчання:

мотивовано обирати мови програмування для розв'язання завдань створення і супроводження програмного забезпечення;

застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення;

знати та вміти застосовувати методи та засоби управління проектами.;

вміти розробляти людино-машинний інтерфейс.

вміти працювати в команді при розробці, погодженні, оформленні та випуску програмної документації.

5.4. Зміст навчальної дисципліни відповідає навчальній та робочій програмам дисципліни і узгоджений з вимогами ринку праці.

5.5. План вивчення дисципліни.

Результати навчання	Навчальна діяльність	Робочий час студента (год.)	Оцінювання (бали)
<p>Знати: здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій</p>	<p>Тема 1. Принципи побудови паралельних обчислювальних систем Лекція 1. План лекції. 1. Мета та задачі вивчення дисципліни, поняття паралельних обчислень. 2. Принципи побудови та архітектура паралельних обчислювальних систем 3. Класифікація паралельних обчислювальних систем 4. Характеристика типових схем комунікації в багато-процесорних обчислювальних системах Список рекомендованих джерел: [1,3]</p>	2	
<p>Вміти: здатність створювати програмне забезпечення для зберігання, відобування та опрацювання даних; здатність застосувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення; здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення</p>	<p>Самостійна робота студентів: - Розглянути способи забезпечення когерентності кешей в системах із загальною пам'яттю</p>	5	4
	<p>Практична робота 1. Інсталяція та налаштування програмного середовища для розробки та налагодження програм з паралельних та розподілених обчислень План заняття: - вивчення теоретичного матеріалу; - виконання завдань; - відповідь на контрольні питання; - захист результатів роботи.</p>	4	8
<p>Знати: здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов,</p>	<p>Тема 2. Моделювання та аналіз паралельних обчислень. Лекція 2. План лекції. 1. Основні показники якості паралельних методів. 2. Оцінка максимально досяжного паралелізму. 3. Загальна характеристика механізмів передачі даних 4. Аналіз трудомісткості основних операцій передачі даних. Список рекомендованих джерел: [2,4]</p>	2	1

Результати навчання	Навчальна діяльність	Робочий час студента (год.)	Оцінювання (бали)
застосуванням теоретичних та методів інформаційних технологій			
Вміти: здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення	Самостійна робота студентів: - Розробити модель і виконати аналіз ефективності паралельних обчислень для завдання множення матриці на вектор. Практична робота 2. Моделювання топології взаємодії процесів. Оцінка комунікаційної трудомісткості паралельних алгоритмів План заняття: - вивчення теоретичного матеріалу; - виконання завдань; - відповідь на контрольні питання; - захист результатів роботи.	5	3
Знати: здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризують комплексністю та невизначеністю умов, застосуванням теоретичних та методів інформаційних технологій	Тема 3. Основи OpenMP. Лекція 3. План лекції. 1. Поняття процесу, потоку і багатопоточності. 2. Технологія OpenMP, особливості та її компоненти. 3. Модель пам'яті. Класи змінних в OpenMP. 4. Режими виконання багатопоточних програм. Список рекомендованих джерел: [4-6]	2	1
Вміти: здатність створювати програмне забезпечення для зберігання, відобування та опрацювання даних; здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення	Самостійна робота студентів: - Самостійно виконати всі приклади, які були наведені в лекції, а також програми запропоновані для самостійного виконання. Практична робота 3. Генерація випадкових чисел з використанням декількох потоків. План заняття: - вивчення теоретичного матеріалу; - виконання завдань; - відповідь на контрольні питання; - захист результатів роботи.	6	3
		2	8

Результати навчання	Навчальна діяльність	Робочий час студента (год.)	Оцінювання (бали)
<p>Знати: здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій</p>	<p>Тема 4. Директиви розподілу і синхронізації роботи. Лекції 4. План лекцій. 1. Розпаралелювання виконання циклів. 2. Розподіл декількох структурних блоків між потоками 3. Розподіл роботи на основі незалежних завдань. 4. Синхронізація виконання різних потоків. Список рекомендованих джерел: [4-6].</p>	4	1
<p>Вміти: здатність створювати програмне забезпечення для зберігання, відобування та опрацювання даних; здатність застосувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення;</p>	<p>Самостійна робота студентів: - Самостійно виконати всі приклади, які були наведені в лекції, а також програми запропоновані для самостійного виконання.</p>	4	4
	<p>Практична робота 4. Рішення диференціальних рівнянь в приватних похідних. План занять: - вивчення теоретичного матеріалу; - виконання завдань; - відповідь на контрольні питання; - захист результатів роботи.</p>	4	8
<p>Знати: здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій</p>	<p>Тема 5. Векторні обчислення за допомогою OpenMP. Лекція 5. План лекції. 1. Поняття векторизації. 2. Векторизація виконуваного коду для сучасних процесорів. 3. Нові можливості в стандарті OpenMP 4. 4. Приклади використання векторизації. Список рекомендованих джерел: [4,5]</p>	2	1
<p>Вміти: здатність створювати програмне забезпечення для зберігання, відобування та опрацювання даних;</p>	<p>Самостійна робота студентів: - Самостійно виконати всі приклади, які були наведені в лекції, а також програми запропоновані для самостійного виконання.</p>	5	4
	<p>Практична робота 5. Векторизація коду для сучасних процесорів. План заняття:</p>	4	8

Результати навчання	Навчальна діяльність	Робочий час студента (год.)	Оцінювання (бали)
здатність застосувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення;	<ul style="list-style-type: none"> - вивчення теоретичного матеріалу; - виконання завдань; - відповідь на контрольні питання; - захист результатів роботи. 		
Знати: здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій	Тема 6. Системи з розподіленою пам'яттю. Основи MPI. Лекція 6. План лекції. 1. Основні поняття і визначення, склад MPI. Синтаксис функцій. 2. Розробка паралельної програми з використанням MPI 3. Прискорення роботи паралельної програми. 4. Побудова паралельних алгоритмів. Список рекомендованих джерел: [4,6]	2	1
Вміти: здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення	Самостійна робота студентів: - Самостійно виконати всі приклади, які були наведені в лекції, а також програми запропоновані для самостійного виконання. Лабораторна робота 6. Розробка паралельних програм з використанням функцій MPI. План заняття: <ul style="list-style-type: none"> - вивчення теоретичного матеріалу; - виконання завдань; - відповідь на контрольні питання; - захист результатів роботи. 	5	4
Знати: здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій	Тема 7. Прийом і передача повідомлень між окремими процесами. Лекція 7. План лекції. 1. Огляд двоточкових обмінів повідомленнями. 2. Блокуючі обміни. 3. Неблокуючі обміни. 4. Паралельні алгоритми підсумовування. Список рекомендованих джерел: [4-6]	2	1
Вміти:	Самостійна робота студентів:	5	2

Результати навчання	Навчальна діяльність	Робочий час студента (год.)	Оцінювання (бали)
здатність здійснювати процес інтеграції системи, застосувати стандарти процедури управління змінами для підтримки цілісності загальної функціональності і надійності програмного забезпечення	<p>- Самостійно виконати всі приклади, які були наведені в лекції, а також програми запропоновані для самостійного виконання.</p> <p>Практична робота 7. Розробка паралельних програм з використанням функцій парної взаємодії процесів.</p> <p>План заняття:</p> <ul style="list-style-type: none"> - вивчення теоретичного матеріалу; - виконання завдань; - відповідь на контрольні питання; - захист результатів роботи. 	2	8
Знати: здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризують комплексністю та невизначеністю умов, застосуванням теоретичних та методів інформаційних технологій	<p>Тема 8. Колективні операції. Лекція 8. План лекції.</p> <ol style="list-style-type: none"> 1. Огляд колективних операцій. Широкомовлення. 2. Функції збору даних зі всіх процесів. 3. Функції розподілу даних по всіх процесам. 4. Функції редукції. <p>Список рекомендованих джерел: [4-6]</p>	2	
Вміти: здатність здійснювати процес інтеграції системи, застосувати стандарти процедури управління змінами для підтримки цілісності загальної функціональності і надійності програмного забезпечення	<p>Самостійна робота студентів: - Самостійно виконати всі приклади, які були наведені в лекції, а також програми запропоновані для самостійного виконання.</p> <p>Практична робота 8. Розробка паралельних програм з використанням функцій колективної взаємодії процесів.</p> <p>План заняття:</p> <ul style="list-style-type: none"> - вивчення теоретичного матеріалу; - виконання завдань; - відповідь на контрольні питання; <p>захист результатів роботи.</p> <p>Самостійна робота студентів: підготовка до складання іспиту</p>	5	2
Разом		120 годин /4 кред.	100 балів
Підсумковий контроль			екзамен

6. Список рекомендованих джерел

Основний

1. С. Немнюгин, О. Стесик, Параллельное программирование для многопроцессорных вычислительных систем. "БХВ", Санкт-Петербург, 2002 г., 396 с.
2. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. - СПб.: БХВ-Петербург, 2011. – 285 с.
3. Орлов С.А., Цилькер Б.Я. Организация ЭВМ и систем: Учебник для вузов. - Питер, 2011. - 680 с.
4. Антонов А.С. "Параллельное программирование с использованием технологии OpenMP: Учебное пособие".-М.: Изд-во МГУ, 2009. - 77 с.
5. Гергель В.П. Теория и практика параллельных вычислений. Учебное пособие. - М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2007 г.
6. Старченко А.В. Методы параллельных вычислений: учебник для вузов / А.В. Старченко, В.Н. Берцун. – Томск: Издательство Томского университета, 2013. – 224 с.
7. Дацюк В.Н. Методическое пособие по курсу "Многопроцессорные системы и параллельное программирование.- Ростов-на-Дону: РГУ, 2014.– 58 с.

Додатковий

8. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ. - М.: МЦНТО, 2001. – 424 с.
9. Корнеев В.В.. Параллельные вычислительные системы. - М.: Нолидж, 2007. – 128 с.
10. Корнеев В.В. Параллельное программирование в MPI. Ижевск: Институт компьютерных исследований, 2009. – 164 с.
11. Barker, M. (Ed.) (2010). Cluster Computing Whitepaper <http://www.dcs.port.ac.uk/~mab/tfcc/WhitePaper/>.
12. Braeunl T. Parallel Programming. An Introduction.- Prentice Hall, 2006.
13. Miller R., Boxer L. A Unified Approach to Sequential and Parallel Algorithms. Prentice Hall, Upper Saddle River, NJ. 2009.
14. Pacheco, S. P. Parallel programming with MPI. Morgan Kaufmann Publishers, San Francisco. 2001.

Internet-ресурси

15. <https://www.openmp.org/>
16. https://uk.wikipedia.org/.../Розподілена_система_керування
17. https://uk.wikipedia.org/wiki/Паралельні_обчислення
18. Introduction to Parallel Computing (Teaching Course) (<http://www.ece.nwu.edu/~choudhar/C58/>)

7. Контроль та оцінювання результатів навчання: під час вивчення дисципліни «Технології розподілених систем та паралельних обчислень» викладачем здійснюється поточний та підсумковий контроль.

Поточний контроль та оцінювання передбачає:

- перевірку рівня засвоєння теоретичного матеріалу (тестування за матеріалами лекції, який здійснюється на початку кожної наступної лекції);
- захист практичних робіт (проходить під час наступної практичної роботи).

8. Політика навчальної дисципліни:

8.1 Відвідування лекційних та практичних занять: відвідування лекційних та практичних занять є обов'язковим. Допускаються пропуски занять з таких поважних причин, як хвороба (викладачу надається копія довідки від медичного закладу), участь в олімпіаді, творчому конкурсі тощо за попереднього домовленістю та згодою викладача за умови дозволу деканату (надаються документи чи інші матеріали, які підтверджують заявлену участь у діяльності студента).

8.2. Відпрацювання пропущених занять: відпрацювання пропущених занять є обов'язковим незалежно від причини пропущеного заняття. Лекційне заняття має бути відпрацьоване до наступної лекції на консультації викладача. Відпрацювання лекційного матеріалу передбачає вивчення пропущеного теоретичного матеріалу та складання тесту за цим матеріалом. Практичне заняття відпрацьовується під час консультації викладача (розклад консультацій на сайті університету).

8.3. Правила поведінки під час занять: обов'язковим є дотримання техніки безпеки в комп'ютерних лабораторіях. Студенти повинні приймати активну участь в обговоренні навчального матеріалу ознайомившись з ним напередодні (навчальний матеріал надається викладачем).

8.4. За порушення академічної доброчесності студенти будуть притягнені до академічної відповідальності у відповідності до положення про дотримання академічної доброчесності педагогічними, науково-педагогічними, науковими працівниками та здобувачами вищої освіти ХНАДУ:

- повторне проходження оцінювання (контрольна робота, іспит, залік тощо);
- повторне проходження навчального курсу.