МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
Кафедра прикладної математики

Затверджено
На засіданні
кафедри прикладної математики
факультету прикладної математики та інформатики
Львівського національного університету імені Івана Франка
(протокол № 1 від 31 серпня 2023р.)

Завідувач кафедри

Юрій ЯЩУК

Силабус з навчальної дисципліни
«Паралельні та розподілені обчислення»,
що викладається в межах першого (бакалаврського) рівня вищої освіти для здобувачів
з спеціальності 113 Прикладна математика

Львів 2023 р.

Назва дисципліни	**Паралельні та розподілені обчислення**
Адреса викладання дисципліни	Головний корпус ЛНУ ім. І. Франка м. Львів, вул. Університетська 1
Факультет та кафедра, за якою закріплена дисципліна	Факультет прикладної математики та інформатики Кафедра прикладної математики
Галузь знань, шифр та назва спеціальності	11 – математика та статистика 113 – прикладна математика
Викладачі дисципліни	Макар Ігор Григорович, кандидат фізико-математичних наук, доцент кафедри прикладної математики, Переймибіда Андрій Андрійович, кандидат фізико-математичних наук, доцент кафедри прикладної математики
Контактна інформація викладачів	Ihor.makar@lnu.edu.ua; https://ami.lnu.edu.ua/employee/makar-i-h andrii.pereimybida@lnu.edu.ua;https://ami.lnu.edu.ua/employee/pereymybida-andriy-andriyovych Головний корпус ЛНУ ім. І. Франка, каб. 378. м. Львів, вул. Університетська, 1
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекцій/лабораторних занять (за попередньою домовленістю).
Сторінка курсу	https://ami.lnu.edu.ua/course/paralelni-i-rozpodileni-obchyslennia-pm
Інформація про дисципліну	Дисципліна “Паралельні та розподілені обчислення” є дисципліною на вибір зі спеціальності 113 – прикладна математика, яка викладається в 8-му семестрі в обсязі 4-ох кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Курс “Паралельні та розподілені обчислення” присвячено проектуванню та розробці відмовостійких та масштабованих систем, що піддаються високим навантаженням. Розглянуто практичні підходи до ефективного розпаралелення обчислювально складних задач. Основна увага зосереджується на розробці архітектури та практичних аспектах побудови розподілених систем за використання сучасних засобів та технологій. Викладення матеріалу здійснюється за допомогою сучасних термінів та понять з галузі інформаційних технологій.
Мета та цілі дисципліни	***Метою*** курсу є ґрунтовне ознайомлення студентів із основними підходами до проектування високопродуктивних розподілених систем, які легко масштабуються, стійкі до відмов та розгортаються у сучасні хмарні сервіси. ***Завданням*** вивчення навчальної дисципліни є здобуття студентами практичних навичок проектування та розробки високонавантажених розподілених систем за допомогою сучасних засобів і технологій.
Література для вивчення дисципліни	**Книги і статті** 1. Луцків А. М., Лупенко С. А., Пасічник В. В. Паралельні та розподілені обчислення. – Львів, Магнолія 2006, 2015. – 566 с. 2. Klepmann M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. – O'Reilly, 2017. – 611 p. 3. Fowler M. Patterns of Enterprise Application Architecture. – Addison

	Wesley, 2002. – 560 p. 4. Joshi U. Patterns of Distributed Systems. – Addison-Wesley Professional, 2023. – 416 p. 5. Brutlag J. Speed Matters for Google Web Search. – June 22, 2009 Url: https://services.google.com/fh/files/blogs/google_delayexp.pdf 6. Medeiros A. Zookeeper's atomic broadcast protocol: Theory and practice. Technical report, 2012. Url: http://www.tcs.hut.fi/Studies/T-79.5001/reports/2012-deSouzaMedeiros.pdf 7. Ananth G., Gupta A., Karypis G., Kumar V. Introduction to Parallel Computing 2nd Edition. – Addison-Wesley, 2003. – 664 p. **Основні онлайн ресурси** 8. Michael Pogrebinsky Distributed Systems & Cloud Computing with Java, Udemy Course, https://www.udemy.com/course/distributed-systems-cloud-computing-with-java/ 9. Martin Klepmann Lecture Series at University of Cambridge: https://www.youtube.com/playlist?list=PLeKd45zvjcDFUEv_ohr_HdUFe97RItdiB 10. Michael Pogrebinsky The Complete Cloud Computing Software Architecture Patterns, https://everymatrix.udemy.com/course/the-complete-cloud-computing-software-architecture-patterns/ **Додаткові онлайн ресурси** 11. MIT Course materials 6.824 Distributed Systems (Spring 2020), https://www.classcentral.com/course/youtube-mit-6-824-distributed-systems-spring-2020-53357/classroom 12. Vivek Sarkar Distributed Programming in Java, Coursers Course, https://www.coursera.org/learn/distributed-programming-in-java?specialization=pcdp 13. Lukasz Antoniak Building Modern Distributed Systems with Java, Udemy Course, https://www.udemy.com/course/building-modern-distributed-systems-with-java/ 14. Introduction to Parallel Computing Tutorial, https://hpc.llnl.gov/training/tutorials/introduction-parallel-computing-tutorial
Обсяг курсу	Загальний обсяг: 120 годин (аудиторних занять: 56 год., з них 28 год. лекцій та 28 год. лабораторних робіт; самостійної роботи: 64 год).
Очікувані результати навчання	Після завершення цього курсу студент буде: ***Знати***: - Властивості розподілених систем; - Практичні аспекти проектування розподілених систем та паралельного опрацювання великих обсягів даних і обчислювально складних задач; ***Вміти***: - Розробляти масштабовані та стійкі до відмов розподілені системи "з нуля" із використанням сучасних фреймворків та технологій.
Ключові слова	Розподілені системи, паралелізм, distributed systems, zookeeper, kafka, java, azure cloud.

<table>
<tr><td>Формат курсу</td><td>Очний. Проведення лекцій, семінарів, лабораторних робіт і консультацій.</td></tr>
<tr><td>Теми</td><td>Подано нижче у таблиці Схема курсу “Паралельні та розподілені обчислення”</td></tr>
<tr><td>Підсумковий контроль, форма</td><td>Залік</td></tr>
<tr><td>Пререквізити</td><td>У курсі розглядаються всі необхідні теми для успішного виконання індивідуальних завдань, однак для кращого засвоєння матеріалу студенти потребують знань з:
- Основ програмування;
- Основ мережних технологій;
- Основ баз даних;
- Практичні навички програмування на Java або іншій мові високого рівня.
Для розгортання сервісів у хмарні сервіси потрібно мати активну підписку Microsoft Azure, GCP чи AWS.</td></tr>
<tr><td>Навчальні методи та техніки, які будуть використовуватися під час викладання курсу</td><td>Презентації, лекції, семінари.
Домашні та індивідуальні завдання</td></tr>
<tr><td>Необхідне обладнання</td><td>Комп’ютер із програмним забезпеченням: java (python, .net), IDE (IntelliJ тощо), docker.
Права доступу для інсталювання програм на комп’ютері чи, альтернативно, можливість працювати на віртуальний машині чи у Docker.</td></tr>
<tr><td>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</td><td>Оцінювання проводиться за 100-бальною шкалою.
<table>
<tr><th colspan="2" rowspan="2">Оцінка за шкалою ECTS</th><th rowspan="2">Оцінка в балах</th><th colspan="3">Оцінка за національною шкалою</th></tr>
<tr><th colspan="2">Екзамен, диференційований залік</th><th>залік</th></tr>
<tr><td>A</td><td>Відмінно</td><td>100 - 90</td><td>Відмінно</td><td>5</td><td rowspan="5">зараховано</td></tr>
<tr><td>B</td><td>Дуже добре</td><td>81- 89</td><td rowspan="2">Добре</td><td rowspan="2">4</td></tr>
<tr><td>C</td><td>Добре</td><td>71 -80</td></tr>
<tr><td>D</td><td>Задовільно</td><td>61 - 70</td><td rowspan="2">Задовільно</td><td rowspan="2">3</td></tr>
<tr><td>E</td><td>Достатньо</td><td>51- 60</td></tr>
<tr><td>FX (F)</td><td>Незадовільно</td><td>0 - 50</td><td>Незадовільно</td><td>2</td><td>не-зараховано</td></tr>
</table></td></tr>
</table>

	Бали нараховуються за наступним співвідношенням: • *домашні завдання*: 20% семестрової оцінки; максимальна кількість балів 20 (4 завдання по 5 балів); • *індивідуальні проекти*: 80% семестрової оцінки; максимальна кількість балів 80 (2-а індивідуальні проекти по 40 балів). Загалом протягом семестру 100 балів. **Критерії оцінювання індивідуальних проектів (макс. 40 балів за кожен проект)** **1. Функціональність (0 – 20 балів)** • **Коректність (0 - 5 балів):** В проекті студент досягнув поставлених цілей та виконав функціональні вимоги. Оцінка може бути знижена, якщо: o є помилки або дефекти, які перешкоджають роботі основних функціональних можливостей? o система поводиться некоректно для певного типу вхідних даних чи умов (наприклад, навантаження, збої) • **Масштабованість та відмовостійкість (0 – 10 балів):** o Сервіс може легко масштабуватися і обробляти більше навантаження та збільшувати користувачів; o Сервіс стійкий до відмов окремих його компонент; • **Ефективність (0 – 5 балів):** o Розроблений сервіс ефективно використовує ресурси (процесор, пам'ять, мережа) **2. Проектування та реалізація (0 - 20 балів)** • **Принципи розподілених систем (0 -15 балів):** o Сервіс розроблений із дотриманням основних принципів та концепцій розподілених систем. Вдало вибрано алгоритми та структури даних для розподілених операцій. Враховано такі проблеми, як накладні витрати на мережний зв'язок, синхронізацію часу тощо. • **Якість коду (0 – 5 балів):** o Код добре структурований, документований, модульний та легкий для читання; o Студент виконав модульне тестування окремих компонент. **3. Документація та презентація (0 – 10 балів)** • **Звіт про проект:** o Звіт чітко пояснює цілі проекту, запропоновану архітектуру, деталі реалізації та результати. o Звіт добре організований, лаконічний та без граматичних помилок. • **Презентація:** o Презентація є інформативною, чітко передає мету проекту, функціональні можливості та доробок студента.

<table>
<tr><td></td><td>
Критерії оцінювання домашніх завдань (20% семестрової оцінки, 5 балів за кожне завдання):
<table>
<tr><td>4-5 балів</td><td>Студент повністю виконав умови завдання; завдання реалізовано правильно; відповідає на практично на всі запитання, пов'язані з тематикою завдання; проводить чіткий аналіз та порівняння отриманих результатів.</td></tr>
<tr><td>3-4 балів</td><td>Студент виконав завдання з незначними помилками, але самостійно їх виправляє, якщо на них вкаже викладач; на деякі запитання, пов'язані з тематикою завдання, відповідає з неточностями; проводить аналіз отриманих результатів з неточностями.</td></tr>
<tr><td>1-3 балів</td><td>Студент виконав завдання частково; завдання реалізовано з помилками, які частково може виправити, якщо на них вкаже викладач; на запитання відповідає з помилками; проводить аналіз отриманих результатів з помилками.</td></tr>
<tr><td>0-1 балів</td><td>Студент виконав завдання частково або з грубими помилками, які самостійно не може виправити; демонструє незнання матеріалу.</td></tr>
<tr><td>0 балів</td><td>Студент не виконав завдання.</td></tr>
</table>

Академічна доброчесність: Очікується, що роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі студента є підставою для її незарахуванння викладачем, незалежно від масштабів плагіату чи обману.
Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та лабораторні зайняття курсу, якщо інше не регламентується правилами узгодженими з деканатом факультету прикладної математики та інформатики. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов’язані дотримуватися термінів визначених для виконання домашніх та індивідуальних завдань, передбачених курсом.
Література. Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.
</td></tr>
<tr><td>Опитування</td><td>Оцінювання якості курсу буде доступне у системі “Dekanat’ після завершенню курсу.</td></tr>
</table>

Схема курсу ***“Паралельні та розподілені обчислення”***

Тиж.	**Тема, план, короткі тези**	**Форма діяльності (заняття)**	**Література, Ресурси в інтернеті**	**К-сть год.**
1	**Тема 1.** Вступ (*Означення, властивості розподілених систем. Переваги, недоліки та практичні аспекти реалізації. Огляд архітектури сучасних розподілених систем*)	Лекція	[2-11]	2
	Тема 1. Формулювання процесу роботи над індивідуальними і практичними завданнями, процедури здачі	Лабораторна, Самостійна робота		2 4
2	**Тема 2.** Розробка системи для розподіленого пошуку: аналіз послідовної версії алгоритму пошуку TF-IDF	Лекція	[5, 7]	4
	Тема 2. Розробка системи для розподіленого пошуку: реалізація послідовної версії алгоритму пошуку TF-IDF	Лабораторна, Самостійна робота	[5, 7]	2 4
3	**Тема 3.** Координація сервісів в кластері. Алгоритм обрання лідера і його реалізація. Архітектура координатор-працівник (a master-worker architecture)	Лекція	[7]	2
	Тема 3. Реалізація координації в кластері з використанням Apache Zookeeper. Інсталяція і налаштування Zookeeper, використання zCli і java API. Огляд альтернатив	Лабораторна, Самостійна робота	[6, 7]	2 4
4	**Тема 4.** Реєстр і виявлення сервісів в кластері (a service registry and discovery)	Лекція	[2, 7]	2
	Тема 4 Реалізація реєстру і виявлення сервісів на основі Apache Zookeeper	Лабораторна, Самостійна робота	[7]	2 6
5	**Тема 5.** Аспекти мережної комунікації в розподілених систем	Лекція	[2], [4]	2
	Тема 5. Реалізація http сервера і клієнта	Лабораторна, Самостійна робота	[7]	2 4
6	**Тема 6.** Балансування навантаження при розробці веб сервісів (Load Balancers)	Лекція	[1, 2], [7]	2
	Тема 6. Балансування навантаження на прикладі HA Proxy	Лабораторна, Самостійна робота	[4], [8], [12]	2 4
7	**Тема 7.** Огляд хмарних технологій.	Лекція	[11]	2
	Тема 7. Приклад розгортання веб сервіса в Microsoft Azure.	Лабораторна, Самостійна	[2, 7]	2 6

		робота		
8	**Тема 8.** Розробка системи для розподіленого пошуку: розпаралелення алгоритму пошуку TF-IDF	Лекція	[7]	4
	Тема 8. Розробка системи для розподіленого пошуку: реалізація версії алгоритму пошуку TF-IDF із розпаралеленням у кластері. Взаємодія вузлів у кластері	Лабораторна, Самостійна робота	[7]	2 6
9	**Тема 9.** Аналіз архітектури системи розподіленого пошуку: компоненти розподіленої системи і протоколи взаємодії	Лекція	[7]	2
	Тема 9. Розробка і функціональне тестування системи розподіленого пошуку в документах. Комунікація між фронтендом і кластером	Лабораторна, Самостійна робота	[7]	2 4
10	**Тема 10.** Методи Монте-Карло і паралельні алгоритми реалізації	Лекція	[2, 7]	2
	Тема 10. Реалізація розподіленої системи для розпаралелення методів інтегрування Монте-Карло (або розробка розподіленої системи для вирішення іншого завдання, за умови узгодження з викладачем) (*Індивідуальне завдання №1*)	Лабораторна, Самостійна робота	[2-7]	2 8
11	**Тема 11.** Презентації та оцінювання *Індивідуального завдання №1*	Лабораторна		2
12	**Тема 12.** Розподілене зберігання даних. Шардинг даних (data sharding). Узгодження хешування (consistent hashing). Реплікація даних у розподілених БД	Лекція	[1-8]	2
	Тема 12. Інсталяція та демонстрація особливостей MongoDB. Масштабування MongoDB	Лабораторна, Самостійна робота	[2, 7]	2 8
13	**Тема 13.** Брокери повідомлень (Message Brokers). Огляд Apache Kafka	Лекція	[2-7]	2
	Тема 13. Інсталяція і налаштування Apache Kafka. Розробка консюмера і продюсера із використанням Java API (Формулювання *індивідуального завдання №2*: розробка розподіленої банківської системи)	Лабораторна, Самостійна робота	[2-7]	2 6
14	**Тема 14.** Обговорення і консультування щодо *Індивідуального завдання №* 2	Лекція		2
	Тема 14. Студентські презентації та оцінювання *Індивідуального завдання №2*. Підсумки	Лабораторна		2