

# КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ  
кафедра математичної інформатики

«ЗАТВЕРДЖУЮ»  
Заступник декана  
з навчальної роботи  
  
Олена КАШПУР  
«07» травня 2021 р.  


## РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ РОЗПОДІЛЕНІ СИСТЕМИ ОБРОБКИ ІНФОРМАЦІЇ

для студентів

галузь знань **12 «Інформаційні технології»**  
спеціальність **122 «Комп'ютерні науки»**  
освітній рівень **магістр**  
освітня програма **«Інформатика»**  
вид дисципліни **обов'язкова**

|  |            |
|--|------------|
| Форма навчання                             | денна      |
| Навчальний рік                             | 2021/2022  |
| Семестр                                    | 2          |
| Кількість кредитів ECTS                    | 4          |
| Мова викладання, навчання<br>та оцінювання | українська |
| Форма заключного контролю                  | іспит      |

Викладачі: к.ф.-м.н., доцент **Олександр ДЕРЕВ'ЯНЧЕНКО**

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

Робоча програма дисципліни «РОЗПОДІЛЕНІ СИСТЕМИ ОБРОБКИ ІНФОРМАЦІЇ» затверджена на засіданні кафедри математичної інформатики

ЗАТВЕРДЖЕНО

Завідувач кафедри математичної інформатики

 \_\_\_\_\_ Василь ТЕРЕЩЕНКО

Протокол № 10 від «27» 04 2021 р.


Схвалено гарантом освітньо-наукової програми «Інформатика»

 \_\_\_\_\_ Степан ШКІЛЬНЯК

«6» Травня 2021 року

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

Протокол від «6» Травня 2021 року № 10

Голова науково-методичної комісії  \_\_\_\_\_ Людмила ОМЕЛЬЧУК

**1. Мета дисципліни.** *«РОЗПОДІЛЕНІ СИСТЕМИ ОБРОБКИ ІНФОРМАЦІЇ»* є важливою дисципліною в сенсі набуття теоретичних та практичних знань в одній із найактуальніших на сьогодні галузей інформаційних технологій, що стосується програмування складних за обсягом обчислень задач із застосуванням багатоядерних комп'ютерів та розподілених середовищ (комп'ютерні мережі, хмари, кластери).

## **2. Попередні вимоги до опанування або вибору навчальної дисципліни:**

1. Успішне опанування курсів *«Основи програмування»*, *«Програмування»* та *«Основи об'єктно-орієнтованого програмування»*, *«Розподілене та паралельне програмування»*.
2. Знання формальних методів розробки програмних систем.
3. Володіння технологіями програмування, основні відомості з програмування на мовах C# та JAVA.

## **3. Анотація навчальної дисципліни:**

Навчальна дисципліна *«РОЗПОДІЛЕНІ СИСТЕМИ ОБРОБКИ ІНФОРМАЦІЇ»* є складовою освітньо-професійної програми підготовки фахівців за освітньо-кваліфікаційним рівнем «магістр» галузі знань 12 «Інформаційні технології» спеціальності 122 «Комп'ютерні науки», освітньо-професійної програми «Інформатика».

Метою і завданням навчальної дисципліни є ознайомлення з одним із основних науково-технологічних напрямків у галузі комп'ютерних технологій та оволодіння технологіями створення систем паралельної та розподіленої обробки даних у Cloud.

Дана дисципліна є обов'язковою навчальною дисципліною за **програмою «Інформатика»**. Викладається у 2 семестрі в **обсязі – 120 годин (4 кредитів ECTS)**; зокрема: *лекції – 26 год., лабораторні – 12 год., консультації – 2 год., самостійна робота – 80 год.* Завершується дисципліна **заліком в 2 семестрі**.

## **4. Завдання (навчальні цілі):**

набуття знань, умінь та навичок (компетентностей) на рівні новітніх досягнень у програмуванні та обробці великих об'ємом даних, відповідно до освітньої кваліфікації «Магістр з комп'ютерних наук». Зокрема, розвивати:

- СК4. Здатність застосовувати методи і засоби організації великих даних для проектування масштабованих інфраструктур консолідації ресурсів зберігання, дослідження, управління, захисту та обслуговування інформації, розв'язання завдань моделювання та прогнозування стратегічних напрямків розвитку бізнесу. здатність формулювання та досліджування математичних моделей, зокрема дискретних математичних моделей.
- СК5. Здатність вирішувати надскладні наукові та інженерні задачі, що передбачають розпаралелювання обчислень, великих витрат машинного часу, обчислювальних ресурсів і методів організації розв'язання задач на суперкомп'ютерах.
- СК6. Здатність використовувати високопродуктивні обчислення для задач з математичного моделювання та прогнозування у фундаментальних і прикладних дослідженнях різних дисциплін, взаємодіяти з іншими суперкомп'ютерними центрами України та зарубіжних країн, здійснювати спільну розробку технологій розподілених обчислень.
- СК14. Здатність проектування динамічних веб-додатків як інформаційної системи із застосуванням об'єктно-орієнтованих технологій програмування, зокрема сучасних програмних засобів підтримки взаємодії клієнта та сервера із застосуванням розподілених систем керування базами даних, супроводження та оптимізація вебсторінок.

## 5. Результати навчання за дисципліною:

| Результат навчання<br>(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 принципи побудови глобальних розподілених систем.   | <i>Лекція, лабораторне заняття</i>                       | <i>Контрольна робота, 60% правильних відповідей</i>                  | 10%  |
| РН1.2   | Знати засоби й методи створення систем штучного інтелекту та машинного навчання.  | <i>Лекція, лабораторне заняття, самостійна робота</i>    | <i>Контрольна робота, 60% правильних відповідей, доповідь</i>        | 40%  |
| РН1.3   | Знати можливості ПАРКС-технології програмування, щодо створення систем розподілених обчислень   | <i>Лекція, лабораторне заняття</i>                       | <i>Захист лабораторної роботи, доповідь</i>                          | 10%  |
| РН2.1   | Вміти оцінювати і вибирати необхідні методи та технології розподіленого та паралельного програмування для вирішення поставленої задачі        | <i>Лабораторне заняття, самостійна робота</i>            | <i>Захист лабораторної роботи</i>                                    | 25%  |
| РН3.1   | Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програм та систем, складати письмові звіти | <i>Лабораторне заняття, самостійна робота</i>            | <i>Захист лабораторної роботи, доповідь</i>                          | 10%  |
| РН4.1   | Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість   | <i>Лабораторна робота, самостійна робота</i>             | <i>Захист лабораторної роботи, доповідь</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>  |                                |        |        |        |        |        |
| <b>ПРН5.</b> Вирішувати складні проблеми, що вимагають систем з великою обчислювальною потужністю для забезпечення масштабованості паралельних алгоритмів і програм. | +                              | +      | +      | +      | +      | +      |

## 7. Схема формування оцінки.

### 7.1. Форми оцінювання студентів:

#### - семестрове оцінювання:

1. *Контрольна робота 1*: РН1.1 – 10 балів / 6 балів;
2. *Контрольна робота 2*: РН1.2 – 15 балів / 9 балів;
3. *Виконання завдань самостійної роботи та доповідь*: РН1.2, РН1.3, РН2.1, РН3.1, РН4.1 – 30 балів / 18 балів;
4. *Виконання та захист лабораторної роботи*: РН1.3, РН2.1, РН3.1, РН4.1 – 45 балів / 27 балів.

### 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 4 тижня семестру;
3. *Контрольна робота 2*: до 14 тижня семестру;
4. *Підготовка доповіді (за темами Доповідей та лекційних занять і самостійної роботи)*: до 13 тижня;
5. *Захист лабораторної роботи*: до 19 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.

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 визначені оцінки визначальною є робота в семестрі.**

Протягом семестру кожен студент повинен виконати лабораторну роботу та звіт за ними. Всі роботи мають однакову складність і відрізняються лише предметною областю, яку студент обирає на власний смак і узгоджує з викладачем. Сумарна оцінка за практичний модуль (лабораторна робота) та доповідь складає відповідно: 50+30=80 балів.

#### **Теми для доповідей**

**Доповідь (презентація на 20-25 слайдів, доповідь на 30-40 хвилин) на одну з тем:**

1. AI and ML in Google Cloud.
2. AWS Transcribe.
3. TensorFlow.
4. Automated ML in Azure.
5. Unity Engine ML Agents.
6. Google Cloud Vision API.
7. Google Vertex AI.
8. Firebase ML.

9. API management Gravitee.
10. AI Hub для задач NLP.
11. Azure Custom Vision
12. Dialogflow
13. IBM Watson Studio/AutoAI.
14. Amazon SageMaker.
15. AWS Step functions.
16. Spinnaker. Deployments at scale.
17. Cloud Speech-to-Text API.
18. Cloud Text-to-Speech API.
19. Google Video Intelligence API.
19. Cloud Natural Language API.
20. AWS Translate.
21. AWS Textract.
22. Google Colab for ML.

**Форма звіту:** презентація на 20-25 слайдів, доповідь.

Інформацію записуємо на Google Диск Вашої групи в персональну папку.

### **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 декілька 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, Python та ін., а потім за допомогою відповідної системи ПАРКС на віртуальних машинах (воркерів) у Cloud.

#### **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.

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

### **7.3 Шкала відповідності оцінок**

|                             |        |
|-----------------------------|--------|
| <b>Зараховано / Passed</b>  | 60-100 |
| <b>Не зараховано / Fail</b> | 0-59   |

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

| № лекції   | Назва лекції   | Кількість годин |                |                |
|--|--|-----------------|----------------|----------------|
|  |  | Лекції          | Лабор. заняття | Самост. робота |
| <b>Частина I. Теоретичні аспекти побудови розподілених систем</b>                    |  |                 |                |                |
| 1  | <b>ТЕМА 1.</b> Вступна частина. Сучасний розвиток паралельних обчислювальних систем.<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу.  | 2               |                | 4              |
| 2  | <b>ТЕМА 2.</b> Базові принципи побудови глобальних розподілених систем.<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу.   | 2               |                | 4              |
| Контрольна робота 1  |  | 1               |                |                |
| Всього по частині 1  |  | 5               |                | 8              |
| <b>Частина II. Створення систем штучного інтелекту та машинного навчання у хмарі</b> |  |                 |                |                |
| 3  | <b>ТЕМА 3.</b> Застосування хмарних технологій для створення розподілених систем.<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу. Підготовка доповіді. Виконання лабораторної роботи. | 2               | 2              | 6              |
| 4  | <b>ТЕМА 4.</b> Засоби розгортання та моніторингу для розподілених систем обробки інформації.<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу. Підготовка доповіді.                     | 2               |                | 4              |
| 5  | <b>ТЕМА 5.</b> Засоби створення систем штучного інтелекту та машинного навчання.<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу. Підготовка доповіді. Виконання лабораторної роботи.  | 2               | 2              | 8              |
| 6  | <b>ТЕМА 6.</b> Застосування Google Colab для машинного навчання (CPU та GPU).<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу. Підготовка доповіді. Виконання лабораторної роботи.     | 2               | 2              | 10             |
| 7  | <b>ТЕМА 7.</b> Застосування Cloud API для створення систем штучного інтелекту.<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу. Підготовка доповіді. Виконання лабораторної роботи.    | 2               |                | 6              |
| 8  | <b>ТЕМА 8.</b> Порівняння сучасних підходів для побудови систем штучного інтелекту у різних Cloud.<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи.     | 2               | 2              | 8              |
| Контрольна робота 2  |  | 1               |                |                |
| Всього по частині 2  |  | 13              | 8              | 42             |
| <b>Частина III. PARQC-C# система розподілених обчислень у Cloud</b>                  |  |                 |                |                |
| 9  | <b>ТЕМА 9.</b> Концепція керуючого простору та алгоритмічного модулю. PARQC-технологія програмування.<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи.  | 2               |                | 4              |

|                     |   |           |           |           |
|---------------------|---|-----------|-----------|-----------|
| 10                  | <b>ТЕМА 10.</b> Розподіл ресурсів та алгоритми диспетчеризації. Робота з системою ПАРКС-С# у Cloud.<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи. | 2         | 2         | 10        |
| 11                  | <b>ТЕМА 11.</b> Огляд засобів ПАРКС для обчислень на Cloud та GPU.<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи.                                  | 2         |           | 8         |
| 12                  | <b>ТЕМА 12.</b> Порівняння різних підходів паралельного програмування у Cloud.<br><i>Самостійна робота.</i> Опрацювання лекційного матеріалу. Виконання лабораторної роботи.                      | 2         | 2         | 8         |
| Всього по частині 3 |   | 8         | 4         | 30        |
| <b>ВСЬОГО</b>       |   | <b>26</b> | <b>12</b> | <b>80</b> |
| Консультація        |   | 2         |           |           |

*Загальний обсяг 120 годин, в тому числі:*

Лекцій – 26 год.

Лабораторні заняття – 12 год.

Консультації – 2 год.

Самостійна робота – 80 год.

## 9. Рекомендовані джерела:

### Основні:

1. Дерев'янченко О.В. Налаштування системи ПАРКС для хмарних обчислень // Навчальний посібник, Київ, 2017-2020 р., 60 с. [Електронний ресурс] – Режим доступу до ресурсу: <http://parcs.unicyb.kiev.ua/mr/>
2. Федоров А. Г. Windows Azure: облачная платформа Microsoft / А. Г. Федоров, Д.Н. Мартынов. [Электронный ресурс]. – Режим доступа: <http://kak.znate.ru/docs/index-61012.html>. – Дата доступа: 03.10.2016. 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. Анисимов А.В., Дерев'янченко А.В. Система ПАРУС-JAVA для параллельных вычислений на компьютерных сетях // Кибернетика и системный анализ, 2005, №1, С.25-36.
5. Дерев'янченко О.В. Моделювання паралельних програм за допомогою системи ПАРКС-JAVA // Наукові записки НаУКМА, Комп'ютерні науки. 2005, Т.36., С.32-38.
6. В. Eckel, Thinking in Java // Prentice Hall, 1999, 859 p.
7. Белицкий Р.И., Логинов В.П. Выбор алгоритма диспетчеризации для мультипроцессорной системы с общей шиной // Управляющие системы и машины, 1991, №1, С. 9–13.
8. Дерев'янченко А.В. Алгоритм оптимизации распределения заданий для решения параллельных задач в неоднородной мультипроцессорной системе // Материалы VI международного научно-практического семинара. 2006, Т.1, С.175-180.
9. Google Cloud Platform - [Електронний ресурс] – Режим доступу до ресурсу: <https://console.cloud.google.com/>
10. Дерев'янченко О.В. ПАРКС-JAVA система для параллельных 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. Czarnul P. Parallel Programming for Modern High Performance Computing Systems// CRC Press, 2018. – 304p.
4. Kurgalin S., Borzunov S. A Practical Approach to High-Performance Computing// Springer, 2019. – 206 p.
5. 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\\_Nemnyugin\\_Intel.pdf](http://www.phys.spb.ru/content/File/Library/studentlectures/Nemnugin/Metod_Nemnyugin_Intel.pdf)
6. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. М., Мир, 1979.
7. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ. М., МЦНМО, 2000., 960 с.