КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Кафедра математичної інформатики

«ЗАТВЕРДЖУЮ»
Заступник декана
з навчальної роботи
_____ Олена КАШПУР
«31» серпня 2022 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ХМАРНІ ОБЧИСЛЕННЯ

для студентів

галузь знань **12 «Інформаційні технології»**
(шифр і назва)
спеціальність **122 «Комп'ютерні науки»**
(шифр і назва спеціальності)
освітній рівень **бакалавр**
(молодший бакалавр, бакалавр, магістр)
освітня програма **«Інформатика»**
(назва освітньої програми)
вид дисципліни **вибіркова**

Форма навчання	**денна**
Навчальний рік	**2025/2026**
Семестр	**7**
Кількість кредитів ECTS	**3**
Мова викладання, навчання та оцінювання	**українська**
Форма заключного контролю	**іспит**

Викладач: **к.ф.-м.н., доц. Деревянченко О.В.**

Пролонговано: на 20__/20__ н.р. _____(_____) «__»___ 20__р.
(підпис, ПІБ, дата)

на 20__/20__ н.р. _____(_____) «__»___ 20__р.
(підпис, ПІБ, дата)

КИЇВ – 2022

Розробник: Деревянченко Олександр Валерійович, кандидат фіз.-мат. наук, доцент кафедри математичної інформатики

ЗАТВЕРДЖЕНО
Зав. кафедри «Математичної інформатики»
_____ Василь ТЕРЕЩЕНКО

Протокол № 1 від "29" серпня 2022 р.

Схвалено гарантом освітньо-професійної програми «Інформатика»

_____ Людмила ОМЕЛЬЧУК «31» серпня 2022 рік
(підпис) (прізвище та ініціали)

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

Протокол від «31» серпня 2022 року № 1
Голова науково-методичної комісії _____ Людмила ОМЕЛЬЧУК

1. Мета дисципліни – «ХМАРНІ ОБЧИСЛЕННЯ» — набуття теоретичних та практичних знань в одній з найактуальніших на сьогодні галузей інформаційних технологій, що стосується програмування складних за обсягом обчислень задач з застосуванням розподілених середовищ (хмари, кластери).

2. Попередні вимоги до опанування або вибору навчальної дисципліни (*за наявності*):

1. Успішне опанування курсів «Програмування», «Об'єктно-орієнтоване програмування», «Інструментальні середовища та технології програмування», «Архітектура обчислювальних систем та комп'ютерні мережі».

2. Знати теоретичні основи теорії алгоритмів та математичної логіки.

3. Володіння технологіями програмування, основні відомості з програмування на мовах Python та JAVA.

3. Анотація навчальної дисципліни:

Навчальна дисципліна «ХМАРНІ ОБЧИСЛЕННЯ» є вибірковою складовою освітньо-професійної програми «Інформатика» 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 - хмарних технологій для опанування знань, що готують студентів для професійної діяльності з розгортання та налаштування хмарних сервісів та оволодіння хмарними технологіями в контексті паралельної та розподіленої обробки великих об'ємів даних.

Дана дисципліна є вибірковою навчальною дисципліною за програмою "Інформатика". Викладається у 7 семестрі в обсязі – 90 годин. (3 кредитів ECTS) зокрема: лекції – 14 год., семінари – 14 год., консультації – 2 год., самостійна робота – 60 год. Завершується дисципліна заліком в 7 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 комп'ютерних наук». Зокрема, розвивати:

- здатність до математичного формулювання та досліджування неперервних та дискретних математичних моделей, обґрунтовування вибору методів і підходів для розв'язування теоретичних і прикладних задач у галузі комп'ютерних наук, аналізу та інтерпретування.;
- здатність реалізовувати високопродуктивні обчислення на основі хмарних сервісів і технологій, паралельних і розподілених обчислень при розробці й експлуатації розподілених систем паралельної обробки інформації;
- здатність застосовувати знання у практичних ситуаціях.

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 підсумкові й оцінці з дисципліни**
Код	**Результат навчання**			

РН1.1	*Знати основні поняття та області застосувань хмарних технологій. Інфраструктуру хмарних сервісів.*	*Лекція,, семінар, самостійна робота*	Контрольна робота, 60% правильних відповідей	10%
РН1.2	*Знати принципи розробки додатків у хмарному середовищі. Питання безпеки, масштабування, розгортання та резервного копіювання для хмарної інфраструктури.*	*Лекція,, семінар, самостійна робота*	Контрольна робота, 60% правильних відповідей	40%
РН1.3	*Знати засоби побудови та розгортання ПАРКС-систем у хмарах та основні принципи хмарних обчислень із застосуванням технології ПАРКС.*	*Лекція,, семінар, самостійна робота*	Захист лабораторної роботи	10%
РН2.1	*Вміти оцінювати і вибирати необхідні методи та технології для хмарних обчислень та вирішення поставленої задачі.*	*Лекція,, семінар, самостійна робота*	Захист лабораторної роботи	25%
РН3.1	*Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програм, складати письмові звіти.*	*Лекція,, семінар, самостійна робота*	Захист лабораторної роботи	10%
РН4.1	*Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість.*	*Самостійна робота*	Захист лабораторної роботи	5%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання.

Результати вивчення дисципліни / **Програмні результати навчання**	**РН 1.1**	**РН 1.2**	**РН 1.3**	**РН 2.1**	**РН 3.1**	**РН 4.1**
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 – 10 балів/6 балів;
3. Виконання завдань самостійної роботи та доповідь: РН1.2, РН2.1, РН3.1, РН4.1 – 30 балів/18 балів;
4. Виконання лабораторної роботи: РН1.3, РН2.1, РН3.1, РН4.1 – 50 балів/30 балів.
- підсумкове оцінювання (у формі заліку):
Згідно пп. 4.6.1 та 7.1.5 «Положення про організацію освітнього процесу у Київському національному університеті імені Тараса Шевченка» залік виставляється на підставі поточного контролю (див. семестрове оцінювання) як сума оцінок/балів за всіма успішно оціненими результатами навчання; оцінки нижче від мінімального порогового рівня до підсумкової оцінки не додаються.

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 7 тижня семестру;
4. Підготовка доповіді (за темами Доповідей та лекційних занять та самостійної роботи): до 9 тижня;
5. Виконання лабораторної роботи: 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.

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".

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. Сумарна оцінка за практичний модуль (лабораторна робота) та доповідь складає відповідно: 50+40=80 балів.

<u>Теми для доповідей</u>

Доповідь (презентація на 15-20 слайдів, доповідь на 25-30 хвилин) на одну з тем:

1. Google Cloud Storage.
2. Google Calendar API.

3. Google Photo API.
4. Google Cloud Translation API.
5.Google Cloud Vision API.
6. Google Cloud Speech-to-Text API.
7. Google Cloud Text-to-Speech API.
8. Google Video Intelligence API.
9. Google Colab for ML.
10. Google Drive API.
11. Google Cloud Firestore.
12. Google Firebase.
13. Google Vertex AI.
14. Google Cloud CDN.
15. Google Maps JavaScript API.
16.Gmail API.
17.Google Cloud Places API.
18.Google Big Query
19.Google Cloud SQL
20. Azure Custom Vision
21. Automated ML in Azure.
22.Azure Databricks.
23.Azure Synapse Analytics
24. AWS Transcribe.
25. AWS Step functions.
26. AWS Translate.
27. AWS Textract.
28. AWS Rekognition.
29. AWS IoT Core.
30.AWS S3
31.AWS CodeGuru
32.AWS Lex
33.Serverless API with AWS
34.AWS Comprehend
35.Amazon DynamoDB
36. Alibaba Cloud.
37. Heroku Cloud Service.
38. Dialogflow API.
39. MongoDB Atlas.
40.Docker Cloud.

Форма звіту: презентація на 15-20 слайдів, доповідь та практичне застосування. 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

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 одної з мов програмування Python або Java, 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

Зараховано / Passed	60-100
Не зараховано / Fail	0-59

8. Структура навчальної дисципліни. Тематичний план лекцій

№ лекції	Назва лекції	Кількість годин		
		Лекції	Семінари	Сам. р-та
Частина І. Теорія хмарних обчислень				
1	**ТЕМА 1.** Вступна частина. Поява та розвиток хмарних технологій. Переваги та недоліки.	1	1	5
2	**ТЕМА 2.** Архітектури хмарних систем. Основні моделі надання послуг у хмарі.	1	1	5
3	**ТЕМА 3**. Основні типи віртуалізації.	2	1	5
	Контрольна робота 1		1	5
Частина ІІ. Хмарні технології				
4	**ТЕМА 4**. Огляд сучасних хмарних платформ: Google Cloud Platform, Amazon Web Service, Microsoft Azure. Основні компоненти та засоби.	1	1	5
5	**ТЕМА 5.** Огляд хмарних застосунків.	1	1	5
6	**ТЕМА 6.** Розгортання та масштабування вебзастосунків у хмарних платформах	1	1	5
7	**ТЕМА 7.** Хмарні сховища даних.	1	1	5
8	**ТЕМА 8.** Безпека даних при використанні хмарних сервісів.	2	1	4
Контрольна робота 2			1	
Частина ІІІ. ПАРКС-технологія програмування для хмарних обчислень				
9	**ТЕМА 9.** Застосування ПАРКС-технології програмування для хмарних обчислень.	1	1	4
10	**ТЕМА 10.** Розподіл ресурсів та алгоритми диспетчеризації. Робота з системою ПАРКС-Python у Cloud.	1	1	4
11	**ТЕМА 11.** Огляд засобів для обчислень у Cloud на GPU.	1	1	4
12	**ТЕМА 12.** Порівняння різних підходів з використанням Cloud для розподілених обчислень.	1	1	4
ВСЬОГО		**14**	**14**	**60**

Загальний обсяг *90 год.,* в тому числі:
Лекцій – 14 год.
Семінари – 14 год.
Консультації – 2 год.
Самостійна робота - 60 год.

9. **Рекомендовані джерела:**

Основні:

1. Деревянченко 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 доступу: 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., Деревянченко А.В. Система ПАРУС-JAVA для параллельных вычислений на компьютерных сетях // Кибернетика и системный анализ, 2005, №1, С.25-36.

5. Деревянченко О.В. Моделювання паралельних програм за допомогою системи ПАРКС-JAVA // Наукові записки НаУКМА, Комп'ютерні науки. 2005, Т.36., С.32-38.

6. B. 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. Деревянченко 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. Деревянченко О.В. ПАРКС-JAVA система для паралельних 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

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.