

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ
КАФЕДРА ІНТЕЛЕКТУАЛЬНИХ ПРОГРАМНИХ СИСТЕМ**

«ЗАТВЕРДЖУЮ»

Заступник декана
з навчальної роботи

_____ Кашпур О.Ф.

«__» _____ 2019 року

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
КЛАСТЕРНІ РОЗРАХУНКИ
для студентів**

галузь знань	12 Інформаційні технології
спеціальність	121 Інженерія програмного забезпечення
освітній рівень	магістр
освітня програма	Програмне забезпечення систем
вид дисципліни	обов'язкова

Форма навчання	денна
Навчальний рік	2019/2020
Семестр	1
Кількість кредитів ECTS	3
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	залік

Викладачі: **к. ф.-м. н., доцент Верес М.М.** (лекції).

Пролонговано: на 20__/20__ н. р. _____ (_____) «__» 20__ р.

на 20__/20__ н. р. _____ (_____) «__» 20__ р.

Розробник: Верес Максим Миколайович, к. ф.-м. н., доцент, доцент кафедри інтелектуальних програмних систем.

ЗАТВЕРДЖЕНО

Завідувач кафедри інтелектуальних програмних систем

_____ О.І. Провотар

Протокол № __ від «__» _____ 2019 р.

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

Протокол від «__» _____ 2019 року №__

Голова науково-методичної комісії _____ Л.Л. Омельчук

«__» _____ 2019 року

Затверджено вченою радою факультету комп'ютерних наук та кібернетики

Протокол від «__» _____ 2019 року №__

Голова вченої ради факультету _____ А.В. Анісімов

1. Мета дисципліни – оволодіння навиком самостійного опрацювання науково-технічної літератури в галузі паралельного програмування, методами створення ефективного коду на багатопроцесорних комп'ютерах для наукових розрахунків, застосуванні їх на практиці та самостійної підготовки і проведення наукової презентації на вибрану тему.

2. Попередні вимоги до опанування навчальної дисципліни: Для успішного вивчення дисципліни «Кластерні розрахунки» студенти повинні відповідати наступним вимогам:

1. **Знати:** поняття програмування та принципи розробки програм; класичні алгоритми та різновиди структур даних
2. **Вміти:** проектувати, розробляти та тестувати програми на базовому рівні.
3. **Володіти елементарними навичками:** програмування мовами C, C++.

3. Анотація навчальної дисципліни. Навчальна дисципліна «Кластерні розрахунки» є складовою освітньо-наукової програми підготовки фахівців за другим (магістерським) рівнем вищої освіти у галузі знань 12 Інформаційні технології за спеціальністю 121 Інженерія програмного забезпечення в рамках освітньо-наукової програми «Програмне забезпечення систем».

Дана дисципліна є нормативною навчальною дисципліною в рамках освітньої програми «Програмне забезпечення систем». Викладається у **1 семестрі 1 курсу магістратури в обсязі – 90 год. (3 кредити ECTS)** зокрема: лекції – 28 год., самостійна робота – 60 год., консультації – 2 год. У курсі передбачено 2 змістовних частини та 2 контрольні роботи. Завершується дисципліна – **заліком**.

В результаті вивчення навчальної дисципліни студент повинен:

знати відомості про методи побудови паралельних алгоритмів обчислення складних задач на багатопроцесорних комп'ютерах для наукових розрахунків.

вміти проектувати та розробляти паралельні програми на багатопроцесорних комп'ютерах для наукових розрахунків.

Дисципліна «Кластерні розрахунки» є базовою для вивчення дисципліни «Актуальні проблеми об'єктно-орієнтованого програмування».

4. Завдання (навчальні цілі). Основними завданнями дисципліни «Кластерні розрахунки» є набуття знань, умінь та навичок (компетентностей) на рівні новітніх досягнень в області кластерних обчислень відповідно до освітньої кваліфікації магістр з інженерії програмного забезпечення. Зокрема, розвивати:

- Здатність до абстрактного мислення, аналізу та синтезу (ЗК-1).
- Здатність проведення теоретичних та прикладних досліджень на відповідному рівні (ЗК-3).
- Здатність проектувати програмне забезпечення, включаючи проведення моделювання його архітектури, поведінки та процесів функціонування окремих підсистем і модулів (СК-3).
- Вміння планувати і проводити наукові дослідження, готувати результати наукових робіт з інженерії програмного забезпечення до оприлюднення (СК-9).

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 підсумковій оцінці з дисципліни
Код	Результат навчання			
РН1.1	Знати поняття програмування на MPI.	Лекції, самостійна робота.	Контрольна робота, 60% правильних відповідей, поточне оцінювання, проектів.	15%
РН1.2	Знати основні етапи побудови та виконання паралельної програми.	Лекції, самостійна робота.	Контрольна робота, 60% правильних відповідей, поточне оцінювання, проектів.	15%
РН1.3	Знати методи побудови паралельних алгоритмів обчислення складних задач на багатопроцесорних комп'ютерах для наукових розрахунків.	Лекції, самостійна робота.	Контрольна робота, 60% правильних відповідей, поточне оцінювання, проектів.	20%
РН2.1	Вміти застосовувати на практиці інструментальні програмні засоби проектування та розробки програмного забезпечення.	Самостійна робота.	Поточне оцінювання, проектів.	24%
РН3.1	Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програм, складати письмові звіти.	Самостійна робота.	Поточне оцінювання, проектів.	10%
РН4.1	Ініціювати проекти, лідерство та повна автономність при їх реалізації.	Самостійна робота.	Поточне оцінювання, проектів.	8%
РН4.2	Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість.	Самостійна робота.	Захист проектів.	8%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання.

Програмні результати навчання	Результати навчання дисципліни						
	РН1.1	РН1.2	РН1.3	РН2.1	РН3.1	РН4.1	РН4.2
ПРН-1. Знати і системно застосовувати методи аналізу та моделювання прикладної області, виявлення інформаційних потреб і збору вихідних даних для проектування програмного забезпечення.	+	+	+	+			+
ПРН-3. Знати і застосовувати базові концепції і методології моделювання інформаційних процесів.		+	+				
ПРН-6. Аналізувати, оцінювати і обирати методи, сучасні програмно-апаратні інструментальні та обчислювальні засоби, технології, алгоритмічні та програмні рішення для ефективного виконання конкретних виробничих задач з програмної інженерії.	+	+	+	+			+
ПРН-7. Обґрунтовано обирати парадигми і мови програмування для вирішення прикладних завдань; застосовувати на практиці системні та спеціалізовані засоби, компонентні технології (платформи) та інтегровані середовища розробки програмного забезпечення.		+			+	+	+
ПРН-8. Проводити аналітичне дослідження параметрів функціонування програмних систем для їх валідації та верифікації, а також проводити аналіз обраних методів, засобів автоматизованого проектування та реалізації програмного забезпечення.			+	+	+	+	+

7. Схема формування оцінки.

7.1 Форми оцінювання студентів.

Семестрове оцінювання:

1. Контрольна робота 1: РН1.1, РН1.2 – **10 балів/6 балів.**
2. Контрольна робота 2: РН1.2, РН1.3 – **10 балів/6 балів.**
3. Самостійна робота (проект 1): РН1.1, РН2.1, РН3.1, РН4.1, РН4.2 – **20 балів/12 балів.**
4. Самостійна робота (проект 2): РН1.2, РН2.1, РН3.1, РН4.1, РН4.2 – **20 балів/12 балів.**
5. Самостійна робота (проект 3): РН1.3, РН2.1, РН3.1, РН4.1, РН4.2 – **20 балів/12 балів.**
6. Самостійна робота (проект 4): РН1.1, РН1.3, РН2.1, РН3.1, РН4.1, РН4.2 – **20 балів/12 балів.**

Підсумкове оцінювання (у формі заліку):

- Залікові бали визначаються як сума оцінок/балів за всіма успішно оціненими результатами навчання передбачених даною програмою.
- Оцінки нижче від мінімального порогового рівня не додаються.
- Мінімальний пороговий рівень для сумарної оцінки за всіма компонентами становить 60% від максимально можливої кількості балів.

7.2 Організація оцінювання:

Терміни проведення форм оцінювання:

1. Контрольна робота: до 7 тижня семестру.
2. Контрольна робота: до 14 тижня семестру.
3. Самостійна робота (проект 1): до 3 тижня семестру.
4. Самостійна робота (проект 2): до 6 тижня семестру.
5. Самостійна робота (проект 3): до 9 тижня семестру.
6. Самостійна робота (проект 4): до 12 тижня семестру.

Студенти мають право на одне перескладання кожної контрольної роботи із можливістю отримання максимально 80% початково визначених за цю контрольну роботу балів. Термін перескладання визначається викладачем.

У випадку відсутності студентів з поважних причин відпрацювання та перездачі контрольних робіт здійснюються у відповідності до «Положення про порядок оцінювання знань студентів при кредитно-модульній системі організації навчального процесу» від 1 жовтня 2010 року.

У разі неякісного виконання проекту, викладач має право не зарахувати проект, або знизити бали.

Студенти мають право здавати проекти після закінчення визначеного для них терміну, але з втратою одного балу за кожен тиждень, який пройшов з моменту закінчення терміну її здачі.

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	Самостійна робота
Частина 1. Мета та задачі паралельної обробки даних. Класифікація. Моделі обчислень та методи аналізу ефективності. Паралельні алгоритми розв'язку задач.			
1	Тема 1 Мета та задачі паралельної обробки даних.	2	4
2	Тема 2. Зв'язок між елементами паралельних систем розрахунку. Топологія.	2	3
3	Тема 3. Класифікація багатопроцесорних систем.	2	4
4	Тема 4. Векторно-конвеєрні суперкомп'ютери. Кластерні системи.	2	4
5	Тема 5. Моделі обчислень та методи аналізу ефективності.	2	3
6	Тема 6. Аналіз комунікаційної складності паралельних алгоритмів.	2	4

7	Тема 7. Принципи розробки паралельних методів. Паралельні матричні обчислення.	2	4
8	Тема 8. Паралельне сортування. Паралельні алгоритми розв'язку систем лінійних рівнянь.	2	4
Контрольна робота 1			2
Всього по частині 1		16	32
Частина 2. Інструментальні засоби паралельного програмування. Паралельні алгоритми розв'язку задач			
9	Тема 9. Технологія розробки для багатопроцесорних систем с розподіленою пам'яттю (стандарт передачі повідомлень MPI).	2	6
10	Тема 10. Технологія паралельного програмування PVM. Технологія паралельного програмування DVM.	2	4
11	Тема 11. Технологія розробки для багатопроцесорних систем с розділеною пам'яттю (стандарт OpenMP).	2	4
12	Тема 12. Паралельні алгоритми на графах.	2	4
13	Тема 13. Паралельний розв'язок диференційних рівнянь.	2	4
14	Тема 14. Концепція Grid та метакомп'ютинг.	2	4
Контрольна робота 2			2
Всього по частині 2		12	28
Консультація		2	
ВСЬОГО		28	60

Загальний обсяг – **90** год., в тому числі:

Лекції – **28** год.

Консультації – **2** год.

Самостійна робота – **60** год.

Теми, винесені на самостійне опрацювання.

1. Продуктивність паралельних комп'ютерів.
2. Порівняння продуктивності архітектури ПК при різних задачах.
3. Критерій оцінки топології мережі.
4. Алгоритми маршрутизації і методи передачі даних.
5. Типові операції взаємодії.
6. Представлення топології мережі.
7. Етапи розробки: розділення обчислень, виділення інформаційних залежностей, масштабування і розподілення обчислень між процесами

Виконання проектів 1-4(аналіз прискорення та ефективності):

1. Паралельні алгоритми сортування.

2. Паралельні алгоритми розв'язку систем лінійних рівнянь
3. Паралельні алгоритми на графах
4. Паралельні алгоритми розв'язку диференційних рівнянь

Деталізовані умови проектів розміщено за посиланням:

<https://drive.google.com/drive/folders/0B1dkotQCgQPvZTJ0d2lvcEpPbk0>

9. Рекомендовані джерела.

Основні:

1. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. – Санкт-Петербург, 2004.
2. Грегори Р. Эндрюс. Основы Многопоточного, параллельного и распределенного программирования. – Москва-Санкт-Петербург-Киев, 2003.
3. Сергей Немнюгин, Ольга Стесик. Параллельное программирование для многопроцессорных вычислительных систем – Санкт-Петербург, 2002.
4. Миллер Р. Последовательные и параллельные алгоритмы. – Москва, 2006.
5. Курс CS338 «Многопроцессорные вычислительные системы и параллельное программирование». <http://hpc-education.unn.ru/ru/>

Додаткові:

1. Макс К. Гофф. Сетевые распределенные вычисления: достижения и проблемы. – Москва, 2005.
2. <http://www.parallel.ru>
3. <http://www.dtf.ru>
4. <http://www.intel.com/software/ru/insight/>
5. <http://openmp.org>
6. <http://www.mpi-forum.org>