

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

**ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ**

**КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ**

**«ЗАТВЕРДЖУЮ»**

Заступник декана  
з навчальної роботи

\_\_\_\_\_ Кашпур О.Ф.

«\_\_»\_\_\_\_\_ 2018 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 **вибору ВНЗ**

Форма навчання	<b>денна</b>
Навчальний рік	<b>2018/2019</b>
Семестр	<b>1</b>
Кількість кредитів ECTS	<b>3</b>
Мова викладання, навчання та оцінювання	<b>українська</b>
Форма заключного контролю	<b>екзамен</b>

**Викладач: к. ф.-м. н., доц. Верес М.М. (лекції, лабораторні заняття).**

Пролонговано: на 20\_\_/20\_\_ н.р. \_\_\_\_\_ (\_\_\_\_\_) «\_\_»\_\_ 20\_\_р.  
(підпис, ПІБ, дата)

на 20\_\_/20\_\_ н.р. \_\_\_\_\_ (\_\_\_\_\_) «\_\_»\_\_ 20\_\_р.  
(підпис, ПІБ, дата)

**КИЇВ – 2018**

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

ЗАТВЕРДЖЕНО

В. о. зав. кафедри інформаційних систем

\_\_\_\_\_ (Іванов Є.О.)  
(підпис) (прізвище та ініціали)

Протокол № 8 від «22» травня 2018 р.

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

---

Протокол від «18» червня 2018 року №\_\_

Голова науково-методичної комісії \_\_\_\_\_ (Хусаїнов Д.Я.)  
(підпис) (прізвище та ініціали)

« \_\_\_\_ » \_\_\_\_\_ 20\_\_ року

**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 або вибору навчальної дисципліни (за наявності):**

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-професійної програми підготовки фахівців за другим (*магістерським*) рівнем вищої освіти *галузі знань* 12 „Інформаційні технології” зі *спеціальності* 121 „Інженерія програмного забезпечення”, *освітньо-професійної програми* „Програмне забезпечення систем”.

Дана дисципліна є обов'язковою навчальною дисципліною за *програмою* “Програмне забезпечення систем”.

Викладається у 1 семестрі 1 курсу магістратури в **обсязі – 90 год.**

(**4 кредитів ECTS**) зокрема: *лекції – 28 год., самостійна робота – 60 год., консультації – 2 год.* У курсі передбачено **2 змістових модулі** та **2 модульні контрольні роботи**. Завершується дисципліна – **заліком у 1 семестрі**.

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 складовою циклу професійної підготовки фахівців освітньо-кваліфікаційного рівня «магістр».

**4. Завдання (навчальні цілі):**

набуття знань, умінь та навичок (компетентностей) на рівні програмування паралельних алгоритмів, відповідно до кваліфікації фахівців з інформаційних технологій. Зокрема, розвивати:

- здатність застосовувати знання у практичних ситуаціях;
- здатність оцінювати та забезпечувати якість виконуваних робіт;
- здатність проектувати та розробляти програмне забезпечення із застосуванням паралельних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління;
- здатність до алгоритмічного мислення.

### 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	Лекція, лабораторне заняття	Тест, 60% правильних відповідей, екзамен	15%
РН1.2	Знати основні етапи побудови та виконання паралельної програми	Лекція, лабораторне заняття	Тест, 60% правильних відповідей, екзамен	15%
РН1.3	Знати методи побудови паралельних алгоритмів обчислення складних задач на багатопроцесорних комп'ютерах для наукових розрахунків	Лекція, лабораторне заняття	Тест, 60% правильних відповідей, екзамен	20%
РН2.1	Вміти застосовувати на практиці інструментальні програмні засоби проектування та розробки програмного забезпечення.	Лабораторне заняття, самостійна робота	Захист лабораторної роботи, екзамен	24%
РН3.1	Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань проектування та розробки програм, складати письмові звіти	Лабораторне заняття	Поточне оцінювання, захист ЛР	10%
РН4.1	Організувати свою самостійну роботу для досягнення результату	Самостійна робота	Поточне оцінювання, Захист лабораторної роботи	8%
РН4.2	Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість	Лабораторна робота	Захист лабораторної роботи	8%

### 6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни	РН 1.1	РН 1.2	РН 1.3	РН 2.1	РН 3.1	РН 4.1	РН 4.2
Програмні результати навчання (з опису освітньої програми)							
ВІР-2. Володіння методами та технологіями нейромерж, некласичних логік, квантових обчислень, кластерних розрахунків, категорного аналізу для розв'язання завдань інженерії програмного забезпечення..	+	+	+	+			+
ПР14. Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проектування в процесі побудови і практичного застосування функціональних моделей організаційно-економічних і виробничо-технічних систем.					+	+	

## 7. Схема формування оцінки.

### 7.1 Форми оцінювання студентів:

#### - семестрове оцінювання:

1. Контрольна робота 1: РН 1.1, РН 1.2 — 10 балів/6 балів.
2. Контрольна робота 2: РН 1.2, РН1.3 — 10 балів/6 балів.
3. Лабораторна робота 1 (проект): РН 1.2, РН1.3, РН 2.1, РН3.1 – 10 балів/5 бали.
4. Лабораторна робота 2 (проект): РН1.2, РН1.3, РН 2.1, РН3.1 – 10 балів/5 бали.
5. Лабораторна робота 3 (проект): РН1.2, РН1.3, РН 2.1, РН3.1 – 10 балів/5 бали.
6. Лабораторна робота 4 (проект): РН1.2, РН1.3, РН 2.1, РН3.1 – 10 балів/5 бали.

#### - підсумкове оцінювання (у формі заліку) вказується:

- максимальна кількість балів які можуть бути отримані студентом: 40 балів;
  - результати навчання які будуть оцінюватись: РН1.1, РН1.2, РН1.3, РН2.1;
  - форма проведення і види завдань: письмова.
- Види завдань: 3 письмових завданя.

### Критерії оцінювання на екзамені

Завдання	Тема завдання	Максимальний відсоток від 40 балів	Всього відсотків
Завдання 1-3	Задача з паралельного програмування	По 33%	100%
			100%

*Студент не допускається до заліку, якщо під час семестру набрав менше ніж 24 балів. Студент допускається до заліку за умови виконання 70% передбачених планом лабораторних робіт.*

### 7.2 Організація оцінювання:

#### Терміни проведення форм оцінювання:

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

Студент має 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 студента з поважних причин відпрацювання та передачі 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 лабораторної роботи, викладач має право не зарахувати лабораторну роботу, або знизити за неї бали.

Студент має право здавати лабораторні роботи 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

<b>Відмінно / Excellent</b>	90-100
<b>Добре / Good</b>	75-89
<b>Задовільно / Satisfactory</b>	60-74
<b>Незадовільно / Fail</b>	0-59
<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	Лабораторні	Самостійна
<b>Частина 1. Мета та задачі паралельної обробки даних. Класифікація. Моделі обчислень та методи аналізу ефективності</b>				
1	<b>Тема 1</b> Мета та задачі паралельної обробки даних.	2		4
2	<b>Тема 2.</b> Зв'язок між елементами паралельних систем розрахунку. Топологія.	2		4
3	<b>Тема 3.</b> Класифікація багатопроцесорних систем	2		4
4	<b>Тема 4.</b> Векторно-конвейерні суперкомп'ютери. Кластерні системи.	2		4
5	<b>Тема 5.</b> Концепція Grid та метакомп'ютинг.	2		4
6	<b>Тема 6.</b> Моделі обчислень та методи аналізу ефективності	2		4
7	<b>Тема 7.</b> Аналіз комунікаційної складності паралельних алгоритмів.	2		4
8	<b>Тема 8.</b> Інструментальні засоби паралельного програмування	2		2
	Контрольна робота 1			
Контроль за підсумками лабораторних робіт 1 та 2				
Всього по частині 1		18		30
<b>Частина 2. Інструментальні засоби паралельного програмування. Паралельні алгоритми розв'язку задач</b>				
9	<b>Тема 9.</b> Технологія розробки для багатопроцесорних систем с розподіленою пам'яттю (стандарт передачі повідомлень MPI)	2		4
10	<b>Тема 10.</b> Технологія паралельного програмування PVM. Технологія паралельного програмування DVM	2		4
11	<b>Тема 11.</b> Технологія розробки для багатопроцесорних систем с розділеною пам'яттю (стандарт OpenMP). Технологія розробки для багатопроцесорних систем(стандарт MC#)	2		4
12	<b>Тема 12.</b> Принципи розробки паралельних методів.	2		4
13	<b>Тема 13.</b> Паралельні матричні обчислення.	2		4
14	<b>Тема 14.</b> Паралельне сортування. Паралельний розв'язок диференційних рівнянь	2		8
	Контрольна робота 2			
Контроль за підсумками лабораторних робіт 3 та 4				
Всього по частині 2		10		30
Консультація		2		

Екзамен			
<b>ВСЬОГО</b>	28		60

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

Лекцій – 28 год.

Консультації – 2 год.

Самостійна робота - 60 год.

**Теми, винесені на самостійне вивчення:**

Продуктивність паралельних комп'ютерів.

Порівняння продуктивності архітектури ПК при різних задачах.

Критерій оцінки топології мережі .

Алгоритми маршрутизації і методи передачі даних.

Типові операції взаємодії.

Представлення топології мережі.

Етапи розробки: розділення обчислень, виділення інформаційних залежностей, масштабування і розподілення обчислень між процесами

Виконання лабораторних робіт 1-4.

Деталізовані умови лабораторних робіт розміщено за посиланнями:

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

**9. Рекомендовані джерела:**

**Основні**

1. Воеводин В.В., Воеводин Вл.В. Паралельные 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. <http://www.mcsharp.net/>

**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>
7. <http://u.pereslavl.ru/~vadim/MCSharp>