

**ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ І.І. МЕЧНИКОВА**  
**ФАКУЛЬТЕТ МАТЕМАТИКИ, ФІЗИКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**  
**КАФЕДРА ОПТИМАЛЬНОГО КЕРУВАННЯ ТА ЕКОНОМІЧНОЇ КІБЕРНЕТИКИ**  
**Силабус курсу «Паралельні та розподілені обчислювання»**

<b>Обсяг</b>	загальна кількість: кредитів –5; годин – 150; змістовних модулів – 2
<b>Семестр</b>	осінній
<b>Дні, Час, Місце</b>	за розкладом занять
<b>Викладач(і)</b>	Вербіцький Віктор Васильович, к. фіз.-мат. наук, доцент, доцент кафедри оптимального керування та економічної кібернетики
<b>Контактний телефон</b>	(048)7340723
<b>E-mail</b>	v.verbitskyi@onu.edu.ua
<b>Робоче місце</b>	кафедра оптимального керування та економічної кібернетики
<b>Консультації</b>	очні консультації: за розкладом on-line консультації: ZOOM за розкладом (посилання генерується на початку занять)

#### КОМУНІКАЦІЯ

Комунікація зі студентами буде здійснюватися в аудиторії, та також з використанням додатків Google Клас, Google Meet, ZOOM.

#### АНОТАЦІЯ КУРСУ

**Предметом** вивчення курсу є основні парадигми паралельного та розподіленого програмування та технології створення паралельних та розподілених паралельних додатків (MPI, OpenMP) для сучасних комп'ютерних архітектур.

#### **Пререквізити курсу**

Матеріал курсу пунктується на раніше отриманих студентами знаннях, практичних вміннях та навичках з тем та напрямів щодо програмування, алгоритмів, структур даних, операційних систем та системного програмування. Відповідні курси викладаються у межах освітньої програми першого (бакалаврського) рівня вищої освіти за спеціальністю 123 «Комп'ютерна інженерія».

#### **Постреквізити курсу**

Цей курс доповнює дисципліни «Програмування», «Архітектура комп'ютерів та низькорівневе програмування», «Введення в сучасні операційні системи і середовища», «Структури даних та алгоритми» в сфері розробки паралельних та розподілених програм на основі паралельних алгоритмів для комп'ютерів з сучасною архітектурою і є базою для засвоєння наступних дисциплін освітньо-професійної програми підготовки бакалаврів за спеціальністю 123 «Комп'ютерна інженерія»: «Проектно-технологічна практика», «Преддипломна практика», «Дипломне проектування».

**Метою курсу** є вивчення основних принципів побудови паралельних та розподілених програмних додатків для різноманітних комп'ютерних систем, а також придбання практичних навичок щодо створення, тестування та експлуатації паралельного програмного продукту з використанням сучасних пакетів та стандартів паралельного програмування.

## **Зміст курсу**

Розглядаються:

- Два основних підходи до досягнення паралелізму: паралельне програмування, розподілене програмування. Найпростіша модель паралельного програмування PRAM. Найпростіша модель розподіленого програмування. Процес проектування паралельних і розподілених програм: декомпозиція, зв'язок, синхронізація. Базові рівні програмного паралелізму. Середовища для паралельного і розподіленого програмування.
- Класифікація паралельних обчислювальних систем: SMP-системи, кластери, MPP-системи. Системи із загальною й розподіленою пам'яттю. Архітектура NUMA та ccNUMA. Організація когерентності багаторівневої ієрархічної пам'яті в SMP-системах.
- Головні парадигми паралельного програмування. Ітеративний паралелізм (множення матриць). Рекурсивний паралелізм (адаптивна квадратура). «Виробники та споживачі» (конвеєри). Проблеми паралельного і розподіленого програмування: «гонка» даних, нескінченна відстрочка, взаємоблокування, труднощі організації зв'язку.
- Паралельне програмування з використанням потоків стандарту POSIX. Створення потоку. Атрибути потоку. Очікувані й від'єднанні потоки. Скасування потоку: асинхронне скасування, синхронне скасування, потоки, які не можна скасувати. Потоківі дані. Оброблювач очищення. Очищення поточкових даних у C++. Потоківі семафори. М'ютекси. Умовні змінні POSIX.
- Технологія паралельного програмування OpenMP. Модель OpenMP-додатка. Директива паралельної обробки `parallel`. Директива розподілення роботи `for`. Директиви розподілення роботи `sections` та `section`. Директиви `single` та `master`. Директиви `tasks` та `taskwait`. Директиви синхронізації `barrier`, `ordered`, `critical`, `atomic`. Спільні та приватні змінні. Функції середовища виконання. Функції блокування та синхронізації. Змінні оточення. Алгоритми планування паралельного виконання циклів (`static`, `dynamic`, `guided`, `runtime scheduling`).
- Технологія розподіленого програмування MPI. Призначення MPI. Модель MPI-додатка. Комунікатори. Функції ініціалізації й завершення роботи. Етапи передачі повідомлень між паралельними процесами MPI. Типи даних MPI. Функції передачі повідомлень між процесами типу «один-одному». Колективні комунікації. Розподілені операції в MPI. Створення нових типів даних MPI. Створення розподілених операцій. Топології процесів. Створення декартової топології процесів в MPI-додатках. Приклад використання декартової топології процесів.
- Моделювання та аналіз паралельних обчислень. Модель обчислень у вигляді графа "операції-операнди". Граф інформаційних залежностей послідовного алгоритму. Опис схеми паралельного виконання алгоритму. Характеристики часу виконання паралельного алгоритму. Теореми про оцінку часу виконання паралельного алгоритму. Показники ефективності паралельного алгоритму. Каскадна схема сумування. Модифікована каскадна схема сумування. Оцінка максимально допустимого паралелізму (закон Амдала, ефект Амдала, закон Густавсона – Барсуса). Аналіз масштабованості паралельних обчислень.

## **ОЧІКУВАНІ РЕЗУЛЬТАТИ**

У результаті вивчення курсу студент повинен

**знати:** основні парадигми паралельного та розподіленого програмування; основні прийоми написання паралельних програм з використанням потоків стандарту POSIX; технології OpenMP, MPI написання паралельних програм.

**вміти:** створювати паралельні алгоритми для розв'язування різноманітних задач; аналізувати ефективність паралельного алгоритму; реалізувати паралельний алгоритм для комп'ютерних систем з різними паралельними архітектурами;

**Компетентності**, які отримує студент у результаті вивчення курсу:

- Здатність використовувати та впроваджувати нові технології, включаючи технології розумних, мобільних, зелених і безпечних обчислень, брати участь в модернізації та реконструкції комп'ютерних систем та мереж, різноманітних вбудованих і розподілених додатків, зокрема з метою підвищення їх ефективності.

**Результати навчання:** по завершенню курсу студент матиме навички

- здійснювати пошук інформації в різних джерелах для розв'язання задач комп'ютерної інженерії.
- виконувати експериментальні дослідження за професійною тематикою.
- оцінювати отримані результати та аргументовано захищати прийняті рішення.

#### ФОРМИ І МЕТОДИ НАВЧАННЯ

Курс буде викладений у формі лекцій (36 год.) та лабораторних занять (18 год.), організації самостійної роботи студентів (136 год.).

Основна підготовка студентів здійснюється на лекційних та лабораторних заняттях.

Під час викладання курсу використовуються такі **методи навчання**: *словесні* (лекція, пояснення); *наочні* (презентація); *практичні* (лабораторні роботи); *робота з літературними джерелами* (самостійна робота студентів).