

Київський національний університет
будівництва і архітектури
Кафедра інформаційних технологій

Шифр спеціальності	Назва спеціальності, освітньої програми	Шифр освітньої компоненти за ОП
122	Комп'ютерні науки, "Інформаційні управляючі системи і технології"	Бакалавр

«Затверджую»

Завідувачка кафедри

Тетяна ГОНЧАРЕНКО



Розробник силабуса

Ольга СОЛОВЕЙ



СИЛАБУС

БК "Технологія розподілених систем та паралельних обчислень"

(назва освітньої компоненти)

1) Статус освітньої компоненти: вибіркова	
2) Контактні дані викладача: к.т.н., доцент Соловей Ольга Леонідівна, solovey.ol@knuba.edu.ua , +380 (67) 169 60 14 https://www.knuba.edu.ua/elementor-161775/	
3) Пререквізити: програмування та алгоритмічні мови; об'єктно-орієнтоване програмування.	
4) Коротка анотація освітньої компоненти придбання студентами, які навчаються за напрямом підготовки 122 «Комп'ютерні науки» навички розробки програмного забезпечення з використанням паралельних та багатопотокових обчислень. Ознайомити студентів із сучасними методами та інструментами для створення ефективних багатопоточних додатків, алгоритмів обробки великих обсягів даних, а також інтеграції таких додатків із системами обробки даних реального часу. Завдання освітньої компоненти — оволодіти теоретичними основами паралельних і розподілених обчислень; Опанувати сучасні інструменти для багатопотокового програмування в Java. Навчитися розробляти ефективні алгоритми для паралельної обробки даних.	
5) Структура курсу: лекції, лабораторні роботи, самостійні роботи, розрахунково-графічна робота, залік.	
Загальна кількість кредитів ECTS	3,0
Сума годин:	90
Вид індивідуального завдання	РГР
Форма контролю	Залік
6) Зміст курсу: Лекції Модуль 1. Основні поняття теорії паралельних обчислень Лекція 1. Поняття паралельних та розподілених обчислень Тема 1. Загальні характеристики послідовних та паралельних обчислень. Тема 2. Класифікація паралельності за рівнями. Тема 3. Класифікація паралельних обчислювальних систем: класифікація Фліна та	

класифікація Фенга.

Тема 4. Актуальність і перспективи використання паралельних обчислень.

Тема 5. Поняття потоків у Java: потоки рівня користувача та потоки рівня ядра. Інтерфейси Callable і Runnable — характеристики та відмінності. Методи класу java.lang.Thread. Стани задачі у потоці.

Лекція 2. Блокуючі черги та деки

Тема 1. Основні задачі та характеристики інтерфейсів BlockingQueue, TransferQueue, BlockingDeque.

Тема 2. Методи обробки запитів на отримання, додавання та перегляд задач у блокуючій черзі або деці.

Тема 3. Методи класів ArrayBlockingQueue, DelayQueue, LinkedBlockingDeque, LinkedBlockingQueue, LinkedTransferQueue, PriorityBlockingQueue, SynchronousQueue.

Лекція 3. Пули потоків та керування їх виконанням

Тема 1. Інтерфейс ExecutorService — основні методи для виконання та очікування завершення задач у пулі потоків.

Тема 2. Методи визначення стану потоку та методи зупинки потоків.

Тема 3. Типи пулів потоків: FixedThreadPool, CachedThreadPool, ScheduledThreadPool, SingleThreadExecutor.

Лекція 4. Реалізація рекурсивного підходу в багатопоточному програмуванні мовою Java

Тема 1. Організація виконання завдань у пулі за алгоритмом «розділити – виконати – об'єднати».

Тема 2. Клас ForkJoinTask. Алгоритм «крадіжки задач» (work stealing).

Тема 3. Класи RecursiveTask та RecursiveAction — спільні характеристики та відмінності.

Лекція 5. Атомарні змінні та асинхронні результати в багатопотокових Java-додатках

Тема 1. Класи, включені до пакета java.util.concurrent.atomic, та їх застосування для реалізації неблокуючих (lock-free) операцій синхронізації.

Тема 2. Клас DoubleAccumulator як засіб високопродуктивного накопичення числових значень у багатопотоковому середовищі.

Тема 3. Інтерфейси Future та ScheduledFuture для представлення, керування та планування асинхронних результатів обчислень.

Тема 4. Методи класу CompletableFuture для побудови, композиції та обробки асинхронних обчислень у мові Java.

Модуль 2. Розширені механізми синхронізації та удосконалені алгоритми та механізми блокування

Лекція 6. Розширені механізми синхронізації в мові Java

Тема 1. Блокування за допомогою лічильника. Клас Semaphore.

Тема 2. Методи класу ReentrantLock для керування взаємним виключенням, блокуванням та справедливістю доступу потоків.

Тема 3. Методи класу ReentrantReadWriteLock та їх застосування в багатопотокових сервісах і кешах з інтенсивними операціями читання.

Лекція 7. Удосконалені алгоритми та механізми блокування

Тема 1. Блокування за допомогою лічильника з можливістю зупинки блокування. Клас CountdownLatch.

Тема 2. Бар'єрна синхронізація з можливістю перевикористання створеного бар'єру. Клас CyclicBarrier.

Тема 3. Бар'єрна синхронізація з можливістю перевикористання бар'єру та динамічного додавання сторін. Клас Phaser.

Теми лабораторних робіт:

Лабораторна робота №1. На мові програмування Java розробити власну реалізацію черг ArrayBlockingQueue, DelayQueue, PriorityQueue, SynchronousQueue з бібліотеки java.util.concurrent.

Лабораторна робота №2. На мові програмування Java розробити власну реалізацію політик обробки відхилених задач (RejectedExecutionHandler) з бібліотеки java.util.concurrent.

Лабораторна робота №3. Розробка алгоритму «Сортування злиттям» методами ForkJoinPool.

Лабораторна робота №4. Розробити програму для алгоритму трьохканального злиття з використанням методу get() інтерфейсу Future.

Лабораторна робота №5. Розробити власну реалізацію алгоритму LockFreeQueue.

Лабораторна робота №6. Реалізувати «задачу філософів, що обдають» методами класу Phaser.

Самостійна робота:

Тема 1. Класифікація систем за архітектурою залежно від того, як організована взаємодія між процесорами, пам'яттю та іншими компонентами: SMP (Symmetric Multiprocessing), MIMD (Multiple Instruction, Multiple Data), NUMA (Non-Uniform Memory Access).

Тема 2. Моделі обміну повідомленнями (Message Passing) — інтерфейс обміну повідомленнями, типи повідомлень, транзакції та підтвердження, основні функції MPI.

Тема 3. Порівняння паралельних і розподілених обчислень за параметрами — архітектура, доступ до пам'яті, типи задач, продуктивність, захист від відмов.

Тема 4. Принципи роботи системи обміну повідомленнями Apache Kafka.

Тема 5. Програмування за допомогою бібліотеки CUDA (Compute Unified Device Architecture) — основні компоненти CUDA: ядра, потоки, блоки, сітки. Основи програмування на CUDA.

Тема 6. Розподілені системи для обробки даних: Hadoop та Spark.

Розрахунково-графічна робота на тему «Розгортання Apache Kafka сервера з реалізацією джерела даних та споживача в багатопоточності».

7) Посилання на сторінку електронного навчально-методичного комплексу

дисципліни: <https://org2.knuba.edu.ua/course/view.php?id=307>