



ПАРАЛЕЛЬНЕ ПРОГРАМУВАННЯ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>121 Інженерія програмного забезпечення</i>
Освітня програма	<i>Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці</i>
Статус дисципліни	<i>Вибіркова</i>
Форма навчання	<i>Заочна</i>
Рік підготовки, семестр	<i>3 курс, осінній семестр</i>
Обсяг дисципліни	<i>4кред/120год (лекцій 6 год., комп.прак.4 год., 110 СРС)</i>
Семестровий контроль/ контрольні заходи	<i>Залік/МКР</i>
Розклад занять	<i>http://rozklad.kpi.ua/</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: к.т.н., Варава Іван Андрійович, varava_ivan@ill.kpi.ua, @Varavalvan (у робочий час) Комп'ютерні практикуми: к.т.н., Варава Іван Андрійович, varava_ivan@ill.kpi.ua, @Varavalvan (у робочий час)</i>
Розміщення курсу	<i>Кампус</i>

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Силабус навчальної дисципліни «Паралельне програмування» складено відповідно до освітньої програми підготовки бакалаврів «Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці» спеціальності 121 – Інженерія програмного забезпечення.

Метою навчальної дисципліни є оволодіння студентами теоретичних знань та практичних навичок створення програмного забезпечення з використанням паралельних обчислень, методів та засобів організації взаємодії потоків, засобів синхронізації потоків, формування та закріплення наступних компетентностей:

(ЗК 2) Здатність застосовувати знання у практичних ситуаціях.
(ФК 3) Здатність розробляти архітектури, модулі та компоненти програмних систем.
(ФК 7) Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних. (ФК8) Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення. (ФК 14) Здатність до алгоритмічного та логічного мислення. (ФК19) Здатність застосовувати транслятори мов програмування при реалізації програмних систем.

Предмет навчальної дисципліни – методи та засоби розробки програмного забезпечення для багатоядерних процесорів та багатопроцесорних систем.

Програмні результати навчання, на формування та покращення яких спрямована дисципліна: (ПРН 6) Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення; (ПРН 13) Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань. (ПРН 15) Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

Після засвоєння навчальної дисципліни студенти мають продемонструвати такі результати навчання:

ЗНАННЯ:

- ключових понять і принципів організації паралельних обчислень;
- засобів програмування процесів за допомогою сучасних мов та бібліотек паралельного програмування;
- організації пам'яті та методів взаємодії процесів;

УМІННЯ:

- проектувати паралельні алгоритми;
- розраховувати характеристики ефективності паралельного алгоритму;
- організовувати ефективну взаємодію процесів;
- виконувати відлагодження паралельних програм.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Пререквізити дисципліни. Знання, отримані при вивченні дисциплін: «Основи програмування», «Алгоритми та структури даних», «Компоненти програмної інженерії. Частина 3. Архітектура програмного забезпечення».

Постреквізити дисципліни. Компетенції, знання та уміння, одержані в процесі вивчення кредитного модуля «Паралельне програмування» можуть бути використані при вивченні дисциплін «Розробка програмного забезпечення мобільних пристроїв», «Побудова масштабованих систем обробки даних у реальному часі», «Переддипломна практика» та «Дипломне проектування».

3. Зміст навчальної дисципліни

Тема 1. Основи паралельного програмування

Тема 2. Бібліотеки паралельних обчислень

Тема 3. Паралельне програмування мовою C#

4. Навчальні матеріали та ресурси

Основна література

1. Минайленко Р.М. Паралельні та розподілені обчислення : навч. посіб. — Кропивницький: Видавець Лисенко В. Ф., 2021. — 153 с.
2. Малашонок Г. І., Сідько А. А. Паралельні обчислення на розподіленій пам'яті: OpenMPI, Java, Math Partner : підручник. — Київ : НаУКМА, 2020. — 266 с.
3. Rainer Grimm Concurrency with Modern C++. What every professional C++ programmer should know about concurrency Packt Publishing, 2019. — 727p.
4. Shakti Tanwar Hands-On Parallel Programming with C# and .NET Core. Packt Publishing, 2019, 346 p.

5. Richard Ansorge *Programming in Parallel with CUDA: A Practical Guide*. Cambridge University Press, 2022. – 395 p.

Додаткова література

6. Pas, Ruud van der *Using OpenMP-The Next Step* // Ruud van der Pas, Eric Stotzer, Christian Terboven / Cambridge, MA: The MIT Press, 2017. – 381p.
7. Michael Voss. *Pro Tbb: C++ Parallel Programming with Threading Building Blocks* // Michael Voss, Rafael Asenjo, James Reinders // Apress Berkeley, CA, 2019. – 754p.
8. NVIDIA CUDA – NVIDIA Docs [Електронний ресурс]. – Режим доступу: <https://docs.nvidia.com/cuda/doc/index.html>. – Назва з екрану
9. OpenCL Overview - The Khronos Group Inc [Електронний ресурс]. – Режим доступу: <https://www.khronos.org/opencl/>. – Назва з екрану.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Лекційні заняття

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, посилання на інформаційні джерела)
1	Тема 1. Основи паралельного програмування Лекція 1. Основні поняття та визначення. Огляд паралельних обчислювальних систем. Класифікація паралельних комп'ютерних систем. Типи паралелізму. Характеристики ефективності паралельного алгоритму. Класичні задачі паралельного програмування. Засоби синхронізації потоків. Алгоритми синхронізації.
2	Тема 2. Бібліотеки паралельних обчислень Лекція 2. POSIX Threads, OpenMP, MPI. Intel TBB, Потіки Windows API.
3	Тема 3 Паралельне програмування мовою C# Лекція 3. Багатопоточність в C#. Створення потоків <i>Delegate ThreadStart</i> . Потіки з параметрами та <i>ParameterizedThreadStart</i> . Синхронізація потоків. Оператор <i>lock</i> . Клас <i>System.Threading.Monitor</i> . Клас <i>AutoResetEvent</i> . Клас <i>Thread</i> .

Практичні заняття

№ з/п	Назва теми заняття та перелік основних питань
1	Практичне заняття №1. Створення додатку з використанням MPI <u>Основні питання:</u> Ініціалізація підсистеми MPI; види комунікацій в MPI; блочне перемноження матриць (алгоритм Фокса).
2	Практичне заняття №2. Робота з паралельними колекціями в C# <u>Основні питання:</u> клас <i>Task</i> , простір імен <i>System.Collection.Concurrent</i> , класи <i>ConcurrentQueue<T></i> та <i>ConcurrentStack<T></i>

6. Самостійна робота студента

На самостійну роботу студента відведено 110 годин.

№ з/п	Вид самостійної роботи	Кількість годин СРС
1	Опрацювання тем, які винесені на самостійну роботу	70
2	Підготовка та виконання практичних занять	30
3	Підготовка до МКР	10

4	Підготовка до заліку	30
---	----------------------	----

Теми, які винесені на самостійну роботу

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, посилання на інформаційні джерела)
1	Тема 1. Основи паралельного програмування Монітори. Реалізація моніторів. CAS-операції та атоміки. Lock-free структури даних
2	Тема 2. Бібліотеки паралельних обчислень Модель планування потоків. Функції очікування у Windows API.
3	Тема 3. Паралельне програмування мовою C# Бібліотека Task Parallel Library. Клас Task. Делегат Action. Методи очікування виконання задач. Wait. Клас Parallel. Конструкції Parallel.For та Parallel.ForEach

Питання практичної складової, винесені на самостійне опрацювання

№ з/п	Назва теми заняття та перелік основних питань
1	Паралельний розрахунок інтеграла з використанням pthreads <u>Основні питання:</u> Функція pthread_create(); передача аргументів потоковій функції; функція pthread_exit(); функція pthread_join(); інформація про потік.
2	Програмна реалізація класичної задачі паралельного програмування <u>Основні питання:</u> Задача «Виробники — споживачі», задача «Письменники — читачі» та її варіанти, задача «Обідаючі філософи», задача «Сплячий перукар», механізми синхронізації
3	Створення додатку з використанням OpenMP <u>Основні питання:</u> Директиви паралелізму: parallel, for, sections; директиви синхронізації: critical, barrier, lock/unlock; типи змінних: shared, private; змінні оточення.
4	Розробка багатопоточного додатка з використанням Windows API <u>Основні питання:</u> Створення потоків CreateThread(), ідентифікатор потоку GetCurrentThreadId(), завершення потоків ExitThread() або TerminateThread(), відновлення потоку ResumeThread(), функції очікування WaitForSingleObject() або WaitForMultipleObjects().
5	Розробка багатопоточного додатка мовою C# <u>Основні питання:</u> пріоритет потоків; методи класу Thread: Sleep(), Join(), Interrupt(), Abort(), ResetAbort(), Suspend(), Resume()

7. Контрольна робота

Метою контрольної роботи є закріплення та перевірка теоретичних знань із кредитного модуля, набуття студентами практичних навичок самостійного вирішення задач. Модульна контрольна робота (МКР) розбивається на дві контрольні роботи тривалістю в одну годину кожна. Контрольна робота проводиться за допомогою Google Forms після вивчення всього матеріалу.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Система вимог, які викладач ставить перед студентом:

- правила відвідування занять: відповідно до Наказу 1-273 від 14.09.2020 р. заборонено оцінювати присутність або відсутність здобувача на аудиторному занятті, в тому числі

нараховувати заохочувальні або штрафні бали. Відповідно до PCO даної дисципліни бали нараховують за відповідні види навчальної активності на лекційних та практичних заняттях.

- **правила поведінки на заняттях:** студент має можливість отримувати бали за відповідні види навчальної активності на лекційних та практичних заняттях, передбачені PCO дисципліни. Використання засобів зв'язку для пошуку інформації на гугл-диску викладача, в інтернеті, в дистанційному курсі на платформі Сікорський здійснюється за умови вказівки викладача;

- **політика дедлайнів та перескладань:** якщо студент не проходив або не з'явиться на МКР (без поважної причини), його результат оцінюється у 0 балів. Перескладання результатів МКР не передбачено;

- **політика щодо академічної доброчесності:** Кодекс честі Національного технічного університету України «Київський політехнічний інститут» <https://kpi.ua/files/honorcode.pdf> встановлює загальні моральні принципи, правила етичної поведінки осіб та передбачає політику академічної доброчесності для осіб, що працюють і навчаються в університеті, якими вони мають керуватись у своїй діяльності, в тому числі при вивченні та складанні контрольних заходів з дисципліни «Паралельне програмування»;

- **при використанні цифрових засобів зв'язку з викладачем (мобільний зв'язок, електронна пошта, переписка на форумах та у соцмережах тощо) необхідно дотримуватись загальноприйнятих етичних норм, зокрема бути ввічливим та обмежувати спілкування робочим часом викладача.**

8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Календарний контроль: провадиться під час сесії згідно виконання вимог силабусу.

Семестровий контроль: залік.

Умови допуску до семестрового контролю: семестровий рейтинг більше 40 балів.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
95-100	Відмінно
85-94	Дуже добре
75-84	Добре
65-74	Задовільно
60-64	Достатньо
Менше 60	Незадовільно
Менше 30	Не допущено

Загальна рейтингова оцінка студента після завершення семестру складається з балів, отриманих за:

- тестування по кожному лекційному заняттю;
- виконання завдань до практичних занять;
- виконання модульної контрольної роботи (МКР);
- виконання додаткових завдань.

Тестування по лекціям	Практичні заняття	МКР	Додаткові бали
24	50	20	6

Тестування по матеріалам лекційних занять

Ваговий бал 8. Максимальна кількість балів за тестування – 8 бал * 3 лекцій = 24 балів.

Тестування проводиться за допомогою Google Forms наприкінці поточної лекції. У деяких випадках термін проходження тестування може бути продовжений лектором. Тривалість проходження одного тестування – 5 хвилин. Кількість спроб – одна. У деяких випадках, що пов'язані з технічними проблемами студентів, може надатися повторна спроба на окремі тестування.

Кожне тестування містить 8 запитань різного формату (вибір правильного варіанту з переліку; вибір правильних варіантів з переліку, вірно/невірно; визначити відповідність; чисельна відповідь).

Критерії оцінювання

- запитання типу «вибір правильного варіанту з переліку», «вірно/невірно», «чисельна відповідь» оцінюються однозначно: вірна відповідь – 1 бал, невірна відповідь – 0 балів;
- запитання, на які немає однієї конкретної відповіді, типу «вибір правильних варіантів з переліку» або «визначити відповідність» оцінюються у відповідності до кількості елементів у тесті (наприклад, якщо потрібно обрати 4 вірні варіанти відповіді із 7, то студент отримає по 0,25 балів за один правильний варіант відповіді, а за всі 4 правильні відповіді отримає відповідно 1 бал) – невірна відповідь – 0 балів, частково вірна відповідь – 0,25-0,75 балів, вірна відповідь 1 бал.

Практичні заняття

Ваговий бал – 25. Максимальна кількість балів за всі практичні заняття – 25 балів * 2 заняття = 50 балів.

Два практичні завдання студенти виконують на парах під час сесії, а сім практичних завдань – протягом семестру після установчої сесії.

Критерії оцінювання

- практичне завдання вирішено вірно та надані повні відповіді – 25 балів;
- практичне завдання вирішено вірно, але деякі відповіді неповні або неточні – 18-25 балів;
- практичне завдання вирішено із незначними помилками та надані неповні чи неточні відповіді – 12-17 балів;
- практичне завдання вирішено із значними помилками – повертається на доопрацювання.

Модульна контрольна робота

Ваговий бал за МКР – 20 балів.

Критерії оцінювання

Модульна контрольна робота складається із 20 завдань. Окреме завдання оцінюється в 1 бал.

Календарний контроль базується на поточній рейтинговій оцінці. Умовою позитивної атестації є значення поточного рейтингу студента не менше 50% від максимально можливого на час атестації.

Додаткові (бонусні) бали

Рейтинговою системою оцінювання передбачені додаткові бали за виконання додаткових завдань. Один студент не може отримати більше ніж 6 бонусних балів у семестрі. При отриманні більш ніж 6 балів, вони обмежуються на рівні 6. Бонусні бали можуть бути отримані за виконання «Завдання до лекцій».

Завдання до лекцій. Студенти, за бажанням, можуть виконувати додаткові завдання за матеріалами лекцій. За одне додаткове завдання нараховується 2 бали. Максимальна кількість балів, що можна отримати за завдання до лекцій складає 6 балів.

Форма семестрового контролю – залік

Максимальна сума балів складає 100. Необхідною умовою допуску до заліку є зараховані обидві частини МКР та здані усі завдання до практичних занять. Для отримання заліку з кредитного модулю «автоматом» потрібно мати рейтинг не менше 60 балів, а також виконані умови допуску до заліку.

Студенти, які наприкінці семестру мають рейтинг менше 60 балів, а також ті, хто хоче підвищити свою оцінку в системі ECTS, виконують залікову контрольну роботу. При цьому набрані бали студентом анулюються, а оцінка за залікову контрольну роботу є остаточною.

***Залікова робота.** Залікова робота проводиться на останньому лекційному занятті. Студент проходить тестування за допомогою Google Forms. На тестування пропонується 50 тестових завдань, кожне з яких оцінюється в 2 бали. Для отримання позитивної оцінки необхідно набрати 60 балів і вище. Час тестування зазвичай складає 90 хвилин, але може бути скоригований лектором та (або) викладачам, що приймає залік.*

Робочу програму навчальної дисципліни (силабус) «Паралельне програмування»:

Складено доцентом кафедри інженерії програмного забезпечення в енергетиці, к.т.н. Варавою І.А.

Ухвалено кафедрою інженерії програмного забезпечення в енергетиці (протокол № 43 від 24.06.2026 р.)

Погоджено Методичною комісією ННІАТЕ КПІ ім. Ігоря Сікорського (протокол № 9 від 26.06.2026 р.)