Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Кафедра обчислювальної техніки

Організація обчислень потоків даних

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	***Другий (магістерський)***
Галузь знань	*12 Інформаційні технології*
Спеціальність	*121 Інженерія програмного забезпечення*
Освітня програма	*Інженерія програмного забезпечення комп'ютерних систем*
Статус дисципліни	*Нормативна*
Форма навчання	*очна*
Рік підготовки, семестр	*2 рік, 3 семестр*
Обсяг дисципліни	*5 кредитів (ECTS), 150 годин, включаючи 54 аудиторних години та 96 годин самостійної роботи.*
Семестровий контроль/ контрольні заходи	*Екзамен*
Розклад занять	*Згідно розкладу https://schedule.kpi.ua/*
Мова викладання	*Українська*
Інформація про керівника курсу / викладачів	Лектор та викладач з лабораторних занять: д.т.н. *Анатолій Михайлович Сергієнко, emailanat.srg@gmail.com*
Розміщення курсу	Курс розміщено в GoogleClassroom у домені lll.kpi.ua. Код доступу: 2c3drarv https://classroom.google.com/c/ODQzMjg2MjQ3MzMx?cjc=2c3drarv

Програма навчальної дисципліни

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

Викладання теорії потокових обчислень в університеті має такі аспекти:

Академічний аспект. Теорія потокових обчислень- це наука, яка дає фундаментальні знання. Мета - ознайомити студентів з основними поняттями алгоритмічних моделей, паралельних алгоритмів та з організацією обчислювальних процесів обробки потоків даних.

Професійний аспект. Знання методів обробки потоків даних є обов'язковими для програмістів, які розробляють програмне забезпечення для сучасних паралельних комп'ютерних систем. Мета цього курсу – сформувати навички аналізу паралельних алгоритмів та створення програмного забезпечення для обробки потоків даних. Зокрема, переважна більшість задач обробки сигналів та зображень, штучного інтелекту, розпізнавання образів та глибинного навчання вирішується за допомогою обробки потоків даних.

Інтелектуальні та освітні аспекти. Вивчення теорії потокових обчисленьта алгоритмів обробки потоків даних сприяє розвитку когнітивних навичок, нового погляду на технологію програмування, що є суттєвим інтелектуальним фактором у процесі створення програмного забезпечення.

В результаті вивчення цієї дисципліни відповідно освітній програмі формуються та підсилюються наступні

Компетентності:

ФК12 Здатність створювати та використовувати програмне забезпечення високопродуктивних комп’ютерних систем

ФК13 Здатність планування обчислювальних процесів та проєктування архітектури програмного забезпечення у паралельних обчислювальних системах.

Програмні результати навчання (ПРН):

ПРН22 Знати методи побудови високопродуктивних комп’ютерних систем

ПРН27 Проводити системний аналіз обробки потоків даних, обґрунтовувати вибір комп’ютерної системи для такої обробки.

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

Пререквізити: освітній компонент «Програмне забезпечення комп'ютерних систем»

Постреквізити: освітній компонент «Виконання магістерської дисертації».

3. Зміст навчальної дисципліни

Тема 1: Моделі паралельних алгоритмів.

Предмет і мета курсу «теорія потокових обчислень». Алгоритми, моделі та обчислювальні процеси. Архітектура з точки зору програміста. Алгоритмічні моделі. Паралелізм алгоритму і його види. Паралелізм алгоритмів в мультипроцесорах. Функціональний алгоритм і його графова модель. Потік даних. Синхронні та асинхронні моделі. Графові моделі алгоритмів. Перевірка коректності алгоритму. Класифікація моделей за типом розкладу. Регулярний, динамічний актори. Мережа обробки потоків даних. Графи синхронних потоків даних (ГСПД). Графи залежностей по даних і керуванню. Методи перевірки алгоритму на відсутність блокувань.

Тема 2: Синтез обчислювальних систем обробки потоків даних

Високорівневий синтез ОС обробки потоків даних. Методи планування обчислень. Етапи визначення множини ресурсів, складання розкладу і призначення на ресурси. Методи планування з обмеженням на ресурси. Задача призначення ресурсів. Метод синтезу систолічних процесорів. Засоби високорівневого проектування ОС обробки потоків даних. Мови опису алгоритмів обробки потоків даних. Компілятори-синтезатори. Високорівневі компілятори. Високорівневі компілятори з мови С/С++. Допоміжні системи високорівневого синтезу.

Тема 3: Синтез обчислювальних систем обробки синхронних потоків даних

Програмування ГСПД.Методи оптимізації ГСПД. Параметри ГСПД. Методи ресинхронизації, конвеєризації, розгортання, згортання ГСПД. Метод просторового ГСПД. Відображення конфігурації алгоритму у конфігурації передування та структури. Цілочисельна оптимізація ГСПД. Двохетапний синтез ОС. Способи мінімізації числа ліній зв'язку в ОС. Способи мінімізації об’єму пам'яті. Способи синтезу буферів даних.Синтез ОС, що виконує умовні оператори. Відображення ієрархічного умовного ГСПД у багатопроцесорну ОС.

Тема 4: Програмування обробки потоків даних

Відображення ГСПД в структуру SIMD-процесора. SIMD-архітектури MMX, SSE. Метод відображення ГСПД в структуру SIMD-процесора. Відображення регулярного алгоритму в

паралельну ОС. Основи програмної конвеєризації RISC-процесорів. Метод відображення регулярного алгоритму в паралельну ОС з програмною конвеєеризацією. Програмування систем штучного інтелекту. Архітектура графічного акселератора. Архітектура мови TensorFlow. Відображення ГСПД в структуру графічного акселератора. Відображення ГСПД в структуру акселератора на ПЛІС.

4. Навчальні матеріали

Базова література

1. Кормен Т. Г., Лейзерсон Ч. Е., Рівест Р. Л., Стайн К. Вступ до алгоритмів. Вид. К.І.С. 2019.1288 с.

2. Lee E. A., Seshia S. A. Introduction to Embedded Systems - A Cyber-Physical Systems Approach,Second Edition, MIT Press, 2017. 585 p.
https://ptolemy.berkeley.edu/books/leeseshia/releases/LeeSeshia_DigitalV2_2.pdf

3. Федотов І. Е. Моделі паралельного програмування. 2012. — 384 с.

Додаткова література

1. Сергієнко А.М. VHDL для проєктування обчислювальних пристроїв. К.: ДіаСофт. 2003. 210с.

2. A. Sergiyenko, O. Maslennikow, Y. Vinogradow, "Tensor approach to the application specificprocessor design," 2009 10th International Conference - The Experience of Designing and Application of CAD Systems in Microelectronics, Lviv, UKraine, 2009, pp. 146-149.

3. A. Sergiyenko, A. Serhienko and A. Simonenko, "A method for synchronous dataflowretiming," 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), Kyiv,UKraine, 2017, pp. 1015-1018, doi: 10.1109/UKRCON.2017.8100404

4. Waidyasooriya H. M., Hariyama M. Uchiyama K. Design of FPGA-Based Computing Systems with OpenCL. Springer, 2018. 131 p.

5. Carkci M. Dataflow and Reactive Programming Systems. A Practical Guide to Developing Dataflow and Reactive Programming Systems. Lean Publishing. 2014. 153 p.

6. Hapke H., Nelson C. Building Machine Learning Pipelines. Automating Model Life Cycles with TensorFlow. O'Reilly, 2020. 367 p.

7. System Design, Modeling, and Simulation using Ptolemy II. C. Ptolemaeus, Ed, Ptolemy.org, 2014. 690 p. http://ptolemy.org/systems.

8. Sriram S., Bhattaharyya S. S. Embedded Multiprocessors: Scheduling and Synchronization, 2dEd. 2009. 361 p. DOI: 10.1201/9781420048025

9. Zhou Ye, Communication Systems Modeling in Ptolemy II. 2003.

10. RaoS. K., KailathT. Regular iterative algorithms and their implementation on processor arrays, in Proceedings of the IEEE, vol. 76, no. 3, pp. 259-269, 1988, doi: 10.1109/5.4402.

11. Campbell S. L., Chancelier J.-P., Nikoukhah R. Modeling and Simulation inScilab/Scicos. Springer, 2006. 308 p. https://www.academia.edu/25264504/Modeling_and_Simulation_in_Scilab

Інформаційні ресурси

Слайди лекцій та керівництва до лабораторних робіт https://kanyevsky.kpi.ua/en/for-students/

Обладнання, що необхідне для проведення занять

Лекційні заняття проводяться в аудиторії, яку обладнано проектором, лабораторні заняття – в комп'ютерному класі.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Навчальний зміст дисципліни складається з лекцій та лабораторних робіт.

Лекції.

Тема 1: Моделі паралельних алгоритмів.

Лекція 1. Вступ.

Предмет і мета курсу. Сучасний рівень обчислювальної техніки та виклики, які стоять перед IT-галуззю. Навчальні матеріали. Програмне забезпечення для використання. Рейтинговасистема оцінок. Концепція алгоритму. Алгоритми і моделі. Обчислювальні процеси. Історія комп'ютерних архітектур. Ієрархія архітектур. Архітектура з точки зору програміста. Мови програмування.

Лекція 2. Алгоритмічні моделі. Паралелізм алгоритму.

Поняття алгоритму. Модель Маркова. Модель Колмогорова. Модель Кнута. Модель SMM. Модель RAM. Моделі Ганді. Моделі ASM. Види обчислювальних систем.Паралелізм алгоритму і його види. Явний і прихований паралелізм. Акт синхронізації. Зернистість обчислень. Дрібно-, середньо- та грубозернистий паралелізм. Мультипроцесорна модель. Паралелізм в мультипроцесорах. Паралелізм доступу пам'яті. Послідовнісна узгодженість ОС. Приклади порушення послідовнісної узгодженості.

Лекція 3. Функціональний алгоритм і його графова модель

Функціональний алгоритм. Графи потоків даних (ГПД). Мережа Петрі. Граф Карпа і Міллера. ГПД Деніса. Мережа процесів Кана. Потік даних. Класифікація потоків даних. Синхронні та асинхронні моделі. Види синхронізації потоків.

Лекція 4. Графові моделі алгоритмів

Перевірка коректності алгоритму. Класифікація моделей за типом розкладу. Повністю статичний, частково статичний, динамічний розклади. Розклад зі статичним призначенням. Актор. Регулярний, динамічний актори. Функції запуску актора. Класифікація моделей за типом акторів. Мережа обробки потоків даних.

Лекція5. Графові моделі алгоритмів (продовження)

Граф синхронних потоків даних (ГСПД). Однорідний та неоднорідний ГСПД. Граф періодичного алгоритму. Сигнальний граф. Масштабований ГСПД. Багатовимірний ГСПД. Цикло-статичний ГСПД. Динамічний граф потоків даних (ГПД). Булевський ГПД. Цілочисельний ГПД. Квазістатичний ГПД. ГПД з ефективною поведінкою. Параметричний ГСПД. Блоковий ГСПД. Ієрархічний ГПД. Згорнутий ГСПД.

Лекція 6. Графові моделі алгоритмів (закінчення)

Графи залежностей по даних (ГЗД) і керуванню. Ярусно-паралельна форма ГЗД. Базова ОС. Урівноважена базова ОС. Розгортка обчислень. Граф систолічного процесора. Методи перевіркиалгоритму на відсутність блокувань. Метод граток. Метод лінійного програмування. Метод рівняння балансу. Класифікація моделей за складністю перевірки блокувань.

Тема 2: Синтез ОС обробки потоків даних

Лекція 7. Високорівневий синтез ОС обробки потоків даних

Етапи високорівневого синтезу. Методи планування обчислень. Задача і мета планування. Етапи визначення множини ресурсів, складання розкладу і призначення на ресурси. Задачі планування з обмеженням на ресурси і з обмеженням по часу. Методи планування з обмеженням на ресурси. Метод спискового планування. Метод планування направленим зусиллям. Планування обчислень в конвейерних ОС. Функціональна конвейєризація. Задача призначення ресурсів. Евристика Тсенга. Метод лівої границі. Метод інтервального графу з циклічними дугами.

Лекція 8. Метод синтезу систолічних процесорів

Представлення гнізда циклів гратчастим ГЗД або матрицею векторів міжітераційних залежностей з поліедром.Відображення гратчастого ГЗД в структуру систолічного процесора. Метод синтезу систолічних процесорів в САПР.

Лекція 9. Засоби високорівневого проектування ОС обробки потоків даних

Мови опису алгоритмів обробки потоків даних. Мови синхронного програмування Esterel, Lustre, Signal, Silage. Мова Adata похідні від неї. Мови CSP, Occam та похідні від них. Мови на основі динамічної моделі. Компілятори-синтезатори. Високорівневі компілятори. Високорівневі компілятори з мови С/С++. Допоміжні системи високорівневого синтезу.

Тема 3: Синтез ОС обробки синхронних потоків даних

Лекція 10. Програмування ГСПД.

Вимоги до початкового представлення ГСПД. Представлення однорідного ГСПД мовою VHDL. Представлення неоднорідного ГСПД. Реалізація буферної пам’яті потоку. Представлення булевського і цілочисельного ГПД. Переваги ГСПД перед іншими формами представлення потокових алгоритмів.

Лекція 11. Методи оптимізації ГСПД.

Параметри ГСПД: довжина маршрута, тривалість циклу, границя циклу, границя ітерації, критичний цикл. Урівноваження ациклічного ГСПД. Мета оптимізації ГСПД. Методи ресинхронизації, конвеєризації, розгортання, згортання ГСПД.

Лекція 12. Метод просторового ГСПД.

Просторовий ГСПД. Конфігурація алгоритму. Конфігурацією передування. Конфігурація структури. Умови коректності розкладу просторового ГСПД. Відображення конфігурації алгоритму у конфігурації передування та структури. Досконалий кістяк ГСПД. Цілочисельна оптимізація ГСПД.

Лекція 13. Метод просторового ГСПД (продовження).

Двохетапний синтез ОС. Способи мінімізації числа ліній зв'язку в ОС. Способи мінімізації об’єму пам'яті. Способи синтезу буферів даних.

Лекція 14. Метод просторового ГСПД (закінчення).

Синтез ОС, що виконує умовні оператори. Відображення ієрархічного умовного ГСПД у багатопроцесорну ОС. Способи ресинхронізації, розгортання, згортання ГСПД. Спосіб випереджальних обчислень.

Тема 4: Програмування обробкипотоків даних

Лекція 15. Відображення ГСПД в структуру SIMD-процесора

SIMD-архітектури MMX, SSE. Метод відображення ГСПД в структуру SIMD-процесора.

Лекція 16. Відображення регулярного алгоритму в паралельну ОС

Основи програмної конвеєризації RISC-процесорів. Метод відображеннярегулярного алгоритму в паралельну ОС з програмною конвеєеризацією.

Лекція 17. Програмування систем штучного інтелекту

Архітектура графічного акселератора. Архітектура мови TensorFlow. Відображення ГСПД в структуру графічного акселератора. Відображення ГСПД в структуру акселератора на ПЛІС.

Лекція 18.

Модульна контрольна робота.

Лабораторні роботи

Лабораторна робота 1.

Основи програмування в системі Xcos.

Лабораторна робота 2.

Моделювання однорідного ГСПД.

Лабораторна робота 3.

Моделювання неоднорідного ГСПД.

Лабораторна робота 4.

Оптимізація ГСПД.

6. Самостійна робота студента

Види самостійної роботи для студентів денної форми навчання (96 годин):

• підготовка до лекцій шляхом вивчення попереднього лекційного матеріалу, а такожлітературних джерел, на яких ґрунтується матеріал попередніх лекцій (1 година х 18 лекцій = 18 годин);

• підготовка до лабораторної роботи через ознайомлення із завданням та методичними вказівками до практичних робіт, включаючи вивчення теоретичного матеріалу, необхідного для відповіді на контрольні запитання до практичних робіт, складання і програмування алгоритму, компіляцію та відлагодження програми, розгортання програмного середовища та завантаження вихідних кодів програмного забезпечення для виконання практичних робіт, робота з системою контролю версій Git, а також оформлення звіту. Кожна практична робота вимагає для підготовки та виконання двох тижнів і витрат до 18 годин часу (4 х 18 годин = 72 години);

• підготовка до МКР (6 годин).

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Система вимог до студентів:

• студент зобов’язаний відвідувати лекції та лабораторні заняття, активно працювати над засвоєнням викладеного на них матеріалу;

• на лекції лектор використовує власний презентаційний матеріал;

• лабораторні роботи захищаються у два етапи: перший етап - студенти виконують завдання, складають електронний звіт та надсилають викладачу; другий етап - захист

лабораторних робіт у лабораторії. Контроль знань на лабораторних роботах здійснюється шляхом перевірки звіту про лабораторні роботи, а також питань з матеріалів робіт.

• модульна контрольна робота пишеться як тест на лекції з використанням усіх наявних матеріалів,але без використання комп'ютерних засобів.

Правила відвідування занять. Відвідування занять є вільним, бали за присутність на лекціях та практичних заняттях не додаються, відповідно штрафні бали не передбачаються. Втім, вагома частина рейтингу формується через активну участь у заходах на практичних заняттях.

Правила поведінки на заняттях. На заняттях слід дотримуватись норм етичної поведінки визначених у Кодексі честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» (https://kpi.ua/code). На території університету здобувачі мають поводити себе відповідно до Правил внутрішнього розпорядку (https://kpi.ua/admin-rule).

Дистанційний режим навчання. У разі запровадження обмежень на відвідування університету, пов'язаних з введенням карантину або режиму воєнного стану в державі, освітній процес здійснюється у дистанційному режимі відповідно до Положення про дистанційне навчанняв КПІ ім. Ігоря Сікорського (https://osvita.kpi.ua/index.php/node/188), Регламенту організації освітнього процесу в дистанційному режимі (https://profkom.kpi.ua/reglament-organizatsiyi-osvitnogo-protsesu-v-distantsiynomu-rezhimi) та Регламенту проведення семестрового контролю в дистанційному режимі (https://osvita.kpi.ua/node/148). У режимі дистанційного навчання заняття відбуваються у вигляді онлайн-конференції на платформах BigBlueButton, Google Meet, Zoom.

Посилання на конференцію видається на початку семестру і розміщується в АС «Електронний кампус». З метою забезпечення якісної підготовки здобувачів, дистанційний курс дисципліни розміщено на Платформа дистанційного навчання «Сікорський» (https://www.sikorsky-distance.org). Результати оцінювання висвітлюють у АС «Електронний кампус» на особистій сторінці здобувача (https://ecampus.kpi.ua).

Визнання результатів навчання, набутих у неформальній/інформальній освіті. Порядок визнання таких результатів регламентується Положенням про визнання результатів навчання, набутих у неформальній / інформальній освіті (https://osvita.kpi.ua/index.php/node/179). Можуть бути зараховані окремі змістовні модулі або теми дисципліни. В такому разі здобувач звільняється від виконання відповідних завдань, отримуючи за них максимальний бал відповідно до рейтингової системи оцінювання.

Політика щодо академічної доброчесності. Обов'язковою умовою виконання завдань з освітньої компоненти є дотримання політики та принципів академічної доброчесності (https://kpi.ua/academic-integrity), які, у тому числі, викладено у Кодексі честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» (https://kpi.ua/code), Положенні про систему запобігання академічному плагіату в КПІ ім. Ігоря Сікорського (https://osvita.kpi.ua/node/47). У разі виявлення дублювання робіт, плагіату роботи здобувачі отримують нульовий рейтинг.

Політика використання штучного інтелекту. Використання штучного інтелекту регламентується «Політикою використання штучного інтелекту для академічної діяльності в КПІ ім. Ігоря Сікорського» (https://osvita.kpi.ua/node/1225). Усі навчальні завдання з дисципліни мають бути результатом власної оригінальної роботи здобувача.

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

На першому занятті студенти знайомляться з політикою оцінювання, яка ґрунтується на Положенні про систему оцінювання результатів навчання https://document.kpi.ua/files/2020_1-273.pdf. Рейтинг студента на курсі складається з балів, які він/вона отримує за захищені лабораторні роботи (R1), модульну контрольну роботу (R2) та оцінку на екзамені (R3).

Rs = R1+R2+R3= 100 балів.

В результаті максимальний середній бал дорівнює:

4 лабораторних роботи х 10 балів = 40 балів,

модульна контрольна робота = 20 балів.

Згідно з університетськими правилами щодо моніторингу успішності студента (https://kpi.ua/document_control), існує два тижні оцінювання (атестація), зазвичай протягом 7/8 та 14/15 тижнів семестру, коли оцінюється прогрес у навчанні студентів та відповідність до критеріїв політики оцінювання курсу.

Умовою першої атестації є отримання не менше 10 балів (на момент атестації). Умова другої атестації - набрати не менше 30 балів (на момент атестації).

Критеріями оцінки є:

• Виконання лабораторних робіт:

- бездоганна робота оцінюється у 9 балів;
- є певні недоліки в оформленій роботі - 8-7 балів;
- є деякі недоліки у виконанні робочої програми - 6-5 балів;

Робота не виконана або не захищена - 0 балів.

За роботу, подану вчасно, діє заохочення - 1 бал (сумарно не більше - 6 балів).

• Модульна контрольна робота оцінюється 20 балами. Контрольна робота складається з 20 тестових питань.

За кожну правильну відповідь на тестове питання нараховується 1 бал.

Студенти, які нарешті наберуть необхідну кількість балів (≥60), можуть:

- отримати остаточну залікову оцінку відповідно до рейтингового балу;
- виконатизалікову контрольну роботу, щоб підвищити оцінку.

Студенти можуть отримати до 6 заохочувальних балів за виконання творчих робіт з кредитного модуля (складання тез до конференції, участь у конкурсах, у дослідженнях тощо).

Студенти, чий бал нижче 30, не допускаються до складання іспиту.

На іспиті студенти виконують письмовий тест. Кожне завдання містить два теоретичних питання (завдання) та одне практичне. Перелік питань наведено у Рекомендаціях щодо засвоєння кредитного модуля. Кожне теоретичне запитання (завдання) оцінюється 15 балами за такими критеріями:

- "відмінно", повна відповідь, не менше 90% необхідної інформації), виконана відповідно до вимог до рівня "кваліфікації" - 15-14 балів;

- «добре», достатньо повна відповідь, не менше 75% необхідної інформації, виконана відповідно до вимог до рівня кваліфікації або незначних неточностей - 13-11 балів;

- «задовільна», неповна відповідь, не менше 60% необхідної інформації, виконана відповідно до вимог до «стереотипного» рівня та деяких помилок - 10-8 балів;

- "незадовільно", відповідь не відповідає умовам "задовільно" - 0 балів.

Практичне завдання оцінюється на 10 балів за такими критеріями:

- "відмінно", повне, безпомилкове вирішення завдань - 10 балів;

- «добре», повне вирішення проблем з незначними неточностями або недоліками - 9-8 балів;

- «задовільно», завдання виконується з помилками - 7-5 балів;

- "незадовільно", відповідь не відповідає умовам "задовільно" - 0 балів.

Остаточна оцінка результативності RS приймається за системою оцінювання університету за наступною таблицею.

Таблиця. Остаточна оцінка RS приймається за системою оцінювання університету

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

Робочу програму навчальної дисципліни (силабус):

Складено д.т.н., с.н.с, А. М. Сергієнко

Ухвалено кафедрою обчислювальної техніки (протокол № 10 , від 25.05.2025)

Погоджено Методичною комісією ФІОТ (протокол № 10 , 13.06. 2025)