

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра програмних систем і технологій

«ЗАТВЕРДЖУЮ»

Заступник декана

з навчально-виховної роботи

Наталія ТМЄНОВА

« 29 » *серпня* 2026 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Методи і технології високопродуктивних обчислень

для студентів

галузь знань
спеціальність
освітній рівень
освітня програма

**F Інформаційні технології
F2 Інженерія програмного забезпечення
магістр
Інженерія програмного забезпечення**

вид дисципліни

обов'язкова

Форма навчання

денна

Навчальний рік

2026/2027

Семестр

1

Кількість кредитів ECTS

6

Мова викладання, навчання
та оцінювання

українська

Форма заключного контролю

іспит

Викладачі: **Меркулова Катерина Володимирівна**, кандидат технічних наук, доцент

Пролонговано: на 2027/2028 н.р. _____ (_____) «__» _____ 2027 р.
(підпис, ПІБ, дата)

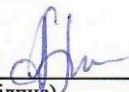
на 2028/2029 н.р. _____ (_____) «__» _____ 2028 р.
(підпис, ПІБ, дата)

на 2029/2030 н.р. _____ (_____) «__» _____ 2029 р.

КИЇВ – 2026

Розробник: **Меркулова Катерина Володимирівна**, кандидат технічних наук, доцент,
доцент кафедри програмних систем і технологій

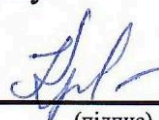
Завідувач кафедри
програмних систем і технологій

 проф. Олексій БИЧКОВ
(підпис)

Протокол № 18 від 17 червня 2026 року

Схвалено науково-методичною комісією факультету інформаційних
технологій

Протокол № 11 від 26 червня 2026 року

Голова науково-методичної комісії  Ганна КРАСОВСЬКА
(підпис)

© Катерина МЕРКУЛОВА, 2026
рік
© _____, 20__ рік
© _____, 20__ рік

1. Мета дисципліни – засвоєння основних методів та алгоритмів організації високопродуктивних обчислень, базових принципів побудови мультипроцесорних та мультикомп'ютерних систем, набуття початкових практичних навиків проектування та програмування таких засобів для розв'язання типових прикладних задач.

2. Попередні вимоги до опанування або вибору навчальної дисципліни:

1. Знати основні поняття та принципи створення додатків; основні методи класів; бібліотеки мови програмування C++.
2. Вміти створювати програмний код; проводити оцінку ефективності розроблених програмних засобів.
3. Володіти елементарними навичками програміста.

3. Анотація навчальної дисципліни:

Вивчаються технології високопродуктивних обчислень, що використовуються для розв'язання складних задач різних прикладних галузей, коли потрібно забезпечити високу продуктивність передачі інформації, застосування великих об'ємів оперативної і постійної пам'яті. Студенти ознайомлюються із сучасними парадигмами щодо програмування багатоядерних та кластерних систем, моделями паралельного програмування. Вивчаються методи ефективної паралельної обробки даних, що забезпечують максимальну високопродуктивності комп'ютерів, раціональний розподіл обсягів і структури пам'яті, пропускної здатності каналів зв'язку, ґрунтуються на мовах програмування, алгоритмах та чисельних методах, що дозволяють реалізувати технології паралельних та розподілених обчислень. Розглядаються технології розпаралелення та алгоритми розподілених обчислень. Значна увага приділена розробці паралельного та розподіленого програмного забезпечення для високопродуктивних комп'ютерних систем на основі технологій OLE, COM, DCOM, CORBA, засоби бібліотеки MPI та OpenMP.

4. Завдання вивчення дисципліни. В результаті вивчення курсу студенти набувають знання основних методів, алгоритмів і засобів високопродуктивної обробки інформації, засобів програмування паралельних та розподілених обчислень та їх реалізації, вміння застосування апаратного та програмного забезпечення обчислювальних систем з засобами паралельної та розподіленої обробки та класів мов програмування високого рівня.

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 ¹ (за необхідності)	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			

1.1	Знати поняття високопродуктивних систем;	лекція	бліц опитування, письмові модульні контрольні роботи, іспит	2%
1.2	знати типи високопродуктивних систем;	лекція		2%
1.3	знати технології високопродуктивних систем;	лекція		2%
1.4	знати основні прийоми виконання та аналізу високопродуктивних систем;	лекція		2%
1.5	знати суперскалярні процесори;	лекція		2%
1.6	знати архітектури ЕОМ для реалізації високопродуктивних обчислень;	лекція		2%
1.7	знати векторну обробку даних;	лекція		4%
1.8	знати векторні процесори;	лекція		2%
1.9	знати кластери робочих станцій;	лекція		2%
1.10	знати приклади високопродуктивної архітектури;	лекція		4%
1.11	знати організацію паралельних обчислень у локальних мережах;	лекція		2%
1.12	знати організацію паралельних обчислень у глобальних мережах;	лекція		4%
1.13	знати трансп'ютероподібні обчислювальні системи;	лекція		4%
1.14	знати підходи до реалізації обчислювальних систем.	лекція		4%
2.1	Вміти працювати з високопродуктивними системами;	лаборатор незаяття	бліц опитування, оцінювання лабораторн их робіт, виконання STEM проєктів,, іспит	до 32%
2.2	вміти працювати з платформами;	лаборатор незаяття		
2.3	вміти обробляти дані	лаборатор незаяття		
2.4	вміти працювати з чергами;	лаборатор незаяття		
2.5	вміти обробляти множини;	лаборатор незаяття		
2.6	вміти здійснювати роботу з потоками;	лаборатор незаяття		
2.7	вміти розв'язувати задачі з використанням паралельних обчислень;	лаборатор незаяття		
2.8	вміти здійснювати внутрішнє перенаправлення;	лаборатор незаяття		
2.9	вміти здійснювати зовнішнє керування;	лаборатор незаяття		
2.10	вміти обробляти дані;	лаборатор незаяття		
2.11	вміти застосовувати алгоритми високопродуктивних обчислень;	лаборатор незаяття		
2.12	вміти обробляти графи;	лаборатор незаяття		
2.13	вміти здійснювати перенаправлення.	лаборатор незаяття		

Модульна контрольна робота			9	18
Виконання та захист STEM-проєкту			15	40
Активна робота на лекціях та лабораторних роботах				5
Опанування окремих результатів навчання шляхом інформальної освіти або завдяки участі у програмах неформальної освіти з обов'язковим отриманням сертифікату (перелік рекомендованих курсів узгоджено на засіданні кафедри ПСТ протокол №1 від 30.08.2023 та включено до методичних рекомендацій з виконання лабораторних робіт) згідно ПОЛОЖЕННЯ про валідацію і визнання результатів навчання здобутих у процесі неформальної або інформальної освіти у програмах вищої або передвищої освіти Київського національного університету імені Тараса Шевченка П.1.6., 1.7, 2.8.		6		6
Участь у науковій конференції з публікацією за тематикою дисципліни				10

Студент самостійно обирає види діяльності за допомогою яких може заробити бали в семестрі , а саме максимально 60 балів.

Поточне семестрове оцінювання:

- оцінюванню підлягає модульна контрольна робота, звіти та захист лабораторних робіт, STEM-проєкт, активна робота на лекціях та лабораторних;
- упродовж семестру, після завершення лекційного матеріалу по змістовним модулям 1 та 2 (теми 2-6 та 7-8), орієнтовно на 9 тижні навчання проводиться письмова модульна контрольна робота у вигляді тесту;
- отримання балів, із застосуванням підходів неформальної освіти, а саме за проходження курсів та обов'язкове отримання сертифікату за темами дисципліни для досягнення РН дисципліни, а також ЗК та ФК відповідно до ОПП;
- виконання та захист STEM-проєкту отримане від викладача за темами дисципліни та для досягнення РН дисципліни, а також ЗК та ФК відповідно до ОНП;
- участь у науковій конференції з публікацією за темами дисципліни та для досягнення РН дисципліни, а також ЗК та ФК відповідно до ОНП.

Здобувач самостійно обирає види діяльності за допомогою яких може заробити бали в семестрі, але максимально можлива кількість балів складає 60 балів.

Кожна лабораторна робота оцінюється в 5 балів, якщо всі завдання виконано, продемонстровано їх працездатність, звіт оформлено відповідно до вимог. Терміном виконання кожної лабораторної роботи є кожен другий тиждень від початку семестру. Мінімальний

пороговий рівень оцінки, що може бути отримана протягом семестру, складає 50% від максимально можливої кількості балів, тобто – 30 балів. Для здобувачів, які набрали сумарно меншу кількість балів ніж пороговий мінімум, для отримання допуску до іспиту необхідно виконати та захистити лабораторні роботи та/або STEM-проект. Оцінювання STEM-проекту проводиться відповідно до «Рекомендацій щодо оцінювання освітніх STEM-проектів», затвердженого на засіданні кафедри програмних систем і технологій - <https://pst.knu.ua/academics/bachelor/evaluation-stem-projects/> (протокол № 16 від «01» травня 2026 р.). Модульна контрольна робота не перескладається. У випадку відсутності здобувача з поважних причин відпрацювання та перездача модульної контрольної роботи здійснюються у відповідності до «Положення про організацію освітнього процесу» від 11 квітня 2022 року.

Підсумкове оцінювання (у формі іспиту):

- екзаменаційний білет складається з теоретичних і практичних питань;
- студент не допускається до іспиту, якщо під час семестру набрав менше ніж мінімальний пороговий рівень оцінки (30 балів);
- для отримання загальної позитивної оцінки з дисципліни оцінка за іспит не може бути меншою 24 балів. При результаті підсумкового оцінювання, що складає менше 24 балів, виставляється оцінка незадовільно.
- максимальна кількість балів які можуть бути отримані студентом - 40 балів на іспиті.

7.2. Шкала відповідності оцінок

Відмінно / Excellent	90-100
Добре / Good	75-89
Задовільно / Satisfactory	60-74
Незадовільно / Fail	0-59

8. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ТЕМАТИЧНИЙ ПЛАН ЛЕКЦІЙ І ЛАБОРАТОРНИХ ЗАНЯТЬ

№ п/ п	Назва лекції	Кількість годин			
		Лекції	Лабора-т орні	СР	Конс уль-т ації
Змістовий модуль 1. Базові складові систем високопродуктивних обчислень.					
1.	Тема 1. Поняття та типи високопродуктивних систем	2		10	
2.	Тема 2. Архітектури для реалізації високопродуктивних обчислень.	2	2	10	
3.	Тема 3. Особливості побудови процесорів для підвищення продуктивності обчислень. Суперскалярні процесори.	2	2	10	
4.	Тема 4. Архітектури для реалізації високопродуктивних обчислень. Векторна обробка даних. Векторні процесори.	2	2	10	
5.	Тема 5. Організація оперативної пам'яті. Зв'язок між елементами високопродуктивних обчислювальних систем.	2	2	10	

6.	Тема 6. Кластери робочих станцій. Приклад високопродуктивної архітектури	2	2	10	
7.	Тема 7. Витрати й виграш при реалізації паралельних і векторних обчислень. Трудові витрати на розпаралелювання й векторизацію програми.	2	4	10	
Змістовий модуль 2. Застосування технологій високопродуктивних обчислень					
8.	Тема 8. Теоретична й практична швидкодія високопродуктивних програм.	4	2	10	
9.	Тема 9. Проблеми синхронізації й рівномірного завантаження процесорів.	2	4	10	
10.	Тема 10. Організація високопродуктивних обчислень у локальних мережах.	2	4	10	
11.	Тема 11. Методи й засоби підтримки мультипроцесорності на рівні операційної системи.	2	4	10	
12.	Тема 12. Трансп'ютероподібні обчислювальні системи.	2	2	11	1
	Модульний контроль	2			
	ВСЬОГО	28	30	121	1

Загальний обсяг 180 год., в тому числі:

- Лекції – 28 год.;
- Лабораторні заняття – 30 год.;
- Консультації – 1 год.
- Самостійна робота – 121 год.

Теми лабораторних занять та STEM проєктів:

1. Способи організації високопродуктивних обчислень.
2. Керування програмними потоками у C++.
3. Багатопотокове програмування у C++.
4. Використання потоків і об'єктів синхронізації (подій, семафорів, м'ютексів і критичних секцій).
5. Реалізація технологій "клієнт-сервер". Обчислення простої функції на сервері.
6. Методи і технології організації розподілених обчислень на основі процедур віддаленого виклику.
7. Розробка інтерфейсу користувача і методу авторизації користувача на Web-сервері через реалізацію методів багатопоточності на основі базової моделі.
8. Розв'язання транспортної задачі паралельними методами.
9. Паралельні методи розв'язання лінійних рівнянь.
10. Паралельні методи розв'язання диференціальних рівнянь.
11. Паралельна реалізація методу скінченних різниць.

Завдання для самостійної роботи та STEM проєктів:

Тема 1. Поняття розподіленої системи. Основні властивості розподілених систем. Основні властивості компонентної декомпозиції. Приклади розподілених систем. Специфіка побудови складних розподілених програмних систем. Загальні принципи побудови розподілених алгоритмів.

Тема 2. Архітектура розподілених систем. Проблеми її реалізації. Моделі розподілених систем. Моделі архітектури. Концепція слоїв (рівнів). Архітектура MPI та CUDA.

Тема 3. Роль проміжного програмного забезпечення при побудові розподілених систем. Місце проміжного програмного забезпечення у архітектурі розподіленої системи. Внутрішня архітектура проміжного програмного забезпечення.

Тема 4. Середовище передачі даних. Протоколи та сервіси. Багаторівнева модель протоколів та потоків. Еталонна модель OSI. Приклади її реалізації. Розподілена та загальна пам'ять. Принципи взаємодії компонентів розподіленої системи. Принципи мережевої адресації.

Тема 5. Призначення паралельних високопродуктивних систем та область їх застосування. Світові тенденції ринку високопродуктивної техніки. Область застосування високопродуктивної техніки в Україні, перелік основних задач та їх розв'язки.

Тема 6. Основні архітектури паралельних обчислювальних апаратних платформ. Їх недоліки та переваги.

Тема 7. Паралельна архітектура з загальною пам'яттю. Її основні компоненти, недоліки та переваги. Конфлікти у паралельних обчислювальних системах з загальною пам'яттю та шляхи їх розв'язання.

Тема 8. Паралельна архітектура з розподіленою пам'яттю. Її основні компоненти, недоліки та переваги. Конфлікти у паралельних обчислювальних системах з розподіленою пам'яттю та шляхи їх розв'язання. Систематика Флінна.

Тема 9. Підходи до паралелізму. Паралелізм на основі даних та програмного коду (задачі). Їх основні недоліки та переваги. Рівні підтримки паралелізму. Програмні засоби реалізації паралельних програм. C++.

9. Рекомендовані джерела:

Основна:

1. Організація паралельних обчислень: Навчальний посібник / Укладачі: Є. Ваврук, О. Лашко. – Львів: Національний університет “Львівська політехніка”, 2019. – 70 с.
2. Кузьменко Б.В., Чайковська О.А. Технологія розподілених систем та паралельних обчислень. (конспект лекцій, частина 1. Розподілені об'єктні системи, паралельні обчислювальні системи та паралельні обчислення, паралельне програмування на основі MPI) Навчальний посібник. – К.: Видавничий центр КНУКІМ, 2019 – 126 с.

Додаткова:

3. Ashwin Pajankar. Raspberry Pi Supercomputing and Scientific Programming. – Nashik, Maharashtra, India, 2019. – 171 p.
4. Gropp, William. Using MPI : portable parallel programming with the Message-Passing Interface / William Gropp, Ewing Lusk, and Anthony Skjellum. Third edition. –Massachusetts Institute of Technology, 2018. – 330 с.
5. Blaise Barney. Introduction to Parallel Computing. – Режим доступу: https://computing.llnl.gov/tutorials/parallel_comp/

Додаткові ресурси:

Інформаційні ресурси

1. Сайт Української команди розподілених обчислень. – Режим доступу: <http://distributed.org.ua/>.
2. Паралельна обробка і паралелізм в .NET Framework. – Режим доступу: [http://msdn.microsoft.com/library/hh156548\(v=vs.110\).aspx](http://msdn.microsoft.com/library/hh156548(v=vs.110).aspx) 10 Паралельні та розподілені обчислення
3. Проект Globus для наукових досліджень. – Режим доступу: <http://www.globus.org>.
4. Форум з використання ГРІД-технологій. – Режим доступу: <http://www.gridforum.org>.
5. Характеристики 500 найпотужніших комп'ютерів в світі. – Режим доступу: <http://www.top500.org>
6. Стандарти MPI. – Режим доступу: <http://www.mpiforum.org>

7. Netlib is a collection of mathematical software, papers, and databases. – Режим доступа: <http://netlib.org>