



ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ВИСОКОПРОДУКТИВНИХ КОМП'ЮТЕРНИХ СИСТЕМ Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	12 Інформаційні технології
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення комп'ютерних систем
Статус дисципліни	Нормативна
Форма навчання	очна(денна)
Рік підготовки, семестр	3 курс, весняний семестр
Обсяг дисципліни	4 кредити 120 годин,
Семестровий контроль/ контрольні заходи	Екзамен МКР
Розклад занять	//rozklad.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: к.т.н, доцент, Корочкін Олександр Володимирович avcora@gmail.com Лабораторні: к.т.н, доцент, Корочкін Олександр Володимирович avcora@gmail.com
Розміщення курсу	//comsys.kpi.ua

Програма навчальної дисципліни

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

Дисципліна "Програмне забезпечення високопродуктивних комп'ютерних систем" спрямована на опанування студентами методів та засобів побудови програмного забезпечення для ВКС з використанням технологій паралельного програмування.

Дисципліна забезпечує наступні програмні результати навчання освітньо-професійної програми Інженерія програмного забезпечення комп'ютерних систем: ФК17, ФК18, ПРН25, ПРН 27.

Метою викладання дисципліни є отримання студентом знань та навичок у сфері створення програмного забезпечення для ВКС, зокрема концепції потоку, реалізації потоків в сучасних мовах і бібліотеках паралельного програмування, методів та засобів організації взаємодії потоків, рішення завдання взаємного виключення та синхронізації потоків, засобів програмування взаємодії потоків: семафорів, мютексів, подій, критичних секцій, моніторів, повідомлень, аналізу та побудови паралельних алгоритмів. Метою навчальної дисципліни є формування у студентів компетентності розробки програмного забезпечення для високопродуктивних комп'ютерних систем з паралельною архітектурою на основі володіння засобами сучасних мов та бібліотек паралельного програмування на робочих місцях і підрозділах під час майбутньої професійної діяльності на першій посаді. За результатами вивчення дисципліни студент має бути здатним вирішувати професійні завдання та володіти таким:: оцінювати особливості ВКС, що застосовуються; використовувати методи створення та оцінювання паралельних

алгоритмів; використовувати засоби сучасних мов та бібліотек паралельного програмування для створення програмного забезпечення сучасних ВКС, володіти основними методами формування програм на основі концепції процесів.

ЗДАТНІСТЬ:

- розробки програмного забезпечення для комп'ютерних систем з паралельною або розподіленою архітектурою на основі володіння засобами сучасних мов та бібліотек паралельного програмування
- аналізувати структуру паралельної комп'ютерної системи
- аналізувати паралельні властивості задачі, що вирішується
 - розробляти та оцінювати паралельні алгоритми
 - здійснювати організацію обчислень в паралельних та розподілених комп'ютерних системах

ЗНАННЯ:

- методів та засобів організації обчислень у високопродуктивних комп'ютерних системах
- структур сучасних паралельних комп'ютерних систем, -організації пам'яті та зв'язку процесорів,
- методів аналізу, оцінювання та представлення паралельних алгоритмів, концепцій розробки паралельних програм
- методів програмування процесів, концепції потоку, операцій з потоками, -методів та засобів організації взаємодії потоків ,
- засобів програмування процесів за допомогою сучасних мов та бібліотек паралельного програмування,
- моделей взаємодії процесів,
- постанови та вирішення завдання взаємного виключення та синхронізації процесів,
- методів налагодження паралельних програм.

УМІННЯ:

- організації обчислювальних процесів в сучасних комп'ютерних системах ,
- створювати програмне забезпечення для високопродуктивних комп'ютерних систем з різною архітектурою ,
- виконувати побудову паралельного алгоритму,
- аналізувати ефективність паралельного алгоритму, формувати алгоритми процесів, - програмувати процеси та створювати паралельну програму,
- організовувати ефективну взаємодію процесів в залежності від структури ВКС,
- розміщувати процеси по процесорах ВКС,
- виконувати налагодження та виконання паралельної програми

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

Необхідні дисципліни: "Основи програмування", "Об'єктна - орієнтоване програмування", "Алгоритми і структури даних", "Компоненти програмної інженерії", "Основи комп'ютерних систем і мереж".

Дисципліни, які базуються на результатах навчання з даної дисципліни: "Безпека програмного забезпечення", "Програмування складних систем"

3. Зміст навчальної дисципліни

Розділ 1. Структури високопродуктивних комп'ютерних систем (ВКС)

Тема 1.1 Базові структури ВКС

Тема 1.2 Організація пам'яті та зв'язків в ВКС

Розділ 2. Програмування паралельних обчислень

Тема 2.1 Особливості програмування паралельних обчислень.

Тема 2.2 Життєвий цикл розробки паралельних програм.

Розділ 3. Аналіз та побудова паралельних алгоритмів

Тема 3.1 Концепція необмеженого паралелізму

Тема 3.2 Ярусно-паралельна форма представлення паралельного алгоритму. Лема Брента. Теорема Мунро-Петерсона.

Тема 3.4 Паралельні алгоритми рішення основних задач лінійної алгебри

Розділ 4. Паралельні потоки

Тема 4.1 Поняття потоку. Стани потоку. Операції з потоками

Тема 4.2 Потоки в сучасних мовах та бібліотеках паралельного програмування

Розділ 5. Комунікація процесів

Тема 5.1 Види комунікації потоків

Тема 5.2 Моделі взаємодії потоків.

Розділ 6. Задачі взаємного виключення і синхронізації

Тема 6.1 Постанова та загальна схема рішення задачі взаємного виключення і синхронізації

Тема 6.2 Алгоритми Деккера

Тема 6.3 Семафори.

Тема 6.4 Мютекси.

Тема 6.5 Події.

Тема 6.6 Критичні секції.

Розділ 7. Монітори

Тема 7.1 Концепція моніторів

Тема 7.2 Реалізація і застосування моніторів

Розділ 8. Взаємодія процесів через посилання повідомлень.

Тема 8.1 Загальна концепція механізму повідомлень

Тема 8.2 Особливості реалізації і використання механізму повідомлень.

Розділ 9. Розробка програмного забезпечення для ВКС

Тема 9.1 Розробка програмного забезпечення для ВКС зі спільною пам'яттю

Тема 9.2 Розробка програмного забезпечення для ВКС зі локальною пам'яттю

Методичні рекомендації

Методика вивчення дисципліни потребує використання сучасних апаратних і програмних засобів, що пов'язано з паралельною обробкою інформації. Що стосовно апаратної складової, то в першу чергу слід використовувати паралельні комп'ютерні системи з багатоядерною архітектурою. Слід приділити увагу структурної організації сучасних багатоядерних процесорів, організації багаторівневої кеш-пам'яті та зв'язку ядер. Лабораторні роботи повинні виконуватися в класі, які обладнано таким багатоядерними комп'ютерами.

Сучасні програмні засоби, що дозволяють програмувати для паралельних (розподілених) комп'ютерних систем базуються на використанні мов та бібліотек паралельного програмування. До таких відносяться мови Java, C#, Ada. OpenMP, MPI. Слід розглянути всітаки системи, бо вони відрізняються як по концепції створення процесів, так і по засобам організації взаємодії процесів.

В якості базової мови обрана мова Ада, яка підтримує класичний підхід до створення процесів через спеціальні модулі task, а також реалізує обидві моделі організації взаємодії процесів як через спільні змінні, так і через посилання повідомлень. Крім того, ресурси мови Ада широко представлено в Інтернеті.

4. Навчальні матеріали та ресурси

Базова:

1. Жуков І.А., Корочкін О.В. Паралельні та розподілені обчислення. Навч. посібник. Друге видання. – К.: Корнійчук, 2014. – 284 с. Гриф надано МОН //comsys.kpi.ua
2. Loutsky G., Zhukov I., Korochkin A. Parallel Computing. – Kyiv, Kornechuk, 2007. -216 pp. Гриф надано МОН //comsys.kpi.ua
3. Багатоядерне програмування на мові Ада: Англ мовою [Електронний ресурс] : навч. посіб. для студ. спеціальності 123 «Комп'ютерна інженерія», спеціалізації «Комп'ютерні системи та мережі», «Технології програмування для комп'ютерних системи та мереж» О.В.Корочкін ; КПІ ім. Ігоря Сікорського. – Електронні текстові данні (1 файл: 2,44 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2018. – 114 с. Гриф надано Методичною радою КПІ ім. Ігоря Сікорського.
4. Паралельні та розподілені обчислення. Вибрані розділи: Навч. посібник для здобувачів ступеня бакалавр за спеціальністю 123 «Комп'ютерні системи та мережі» / Корочкін О.В., Русанова О.В. – Київ : КПІ ім. Ігоря Сікорського, 2020. – 123с Електронний ресурс. Гриф надано Методичною радою КПІ ім. Ігоря Сікорського (протокол № 6 від 31.01.2020 р.) за поданням Вченої ради факультету інформатики та обчислювальної техніки (протокол № 4 від 25.11.2019 р.) <http://ela.kpi.ua/handle/123456789/48224>
5. Програмне забезпечення високопродуктивних комп'ютерних систем. Лабораторний практикум. /Корочкін О.В., – Київ : КПІ ім. Ігоря Сікорського, 2022. – 32 с. Електронний ресурс. //comsys.kpi.ua

Додаткова:

6. Жуков І.А., Корочкін О.В. Паралельні та розподілені обчислення. Навч. посібник. – К.: Корнійчук, 2005. – 226 с. Гриф надано МОН//comsys.kpi.ua
7. Семеренко, В. П. Технології паралельних обчислень : навчальний посібник. – Вінниця : ВНТУ, 2018. – 104 с. ([Електронний ресурс] - Режим доступу: https://www.researchgate.net/publication/334710599_V_P_Semerenko_TEHNOLO..._OBCISLEN_M_inisterstvo_osviti_i_nauki_Ukraini_Vinnickij_nacionalnij_tehnicnij_universitet.)
8. Ringler R. C# Multithreaded and Parallel Programming. Birmingham: Packt Publishing, 2014. – 323с. ([Електронний ресурс] - Режим доступу: <https://www.pdfdrive.com/c-multithreaded-and-parallel-programming-develop-powerful-c-applications-to-take-advantage-of-todays-multicore-hardware-e187077213.html>)
9. Terrell R. Concurrency in .NET. Shelter Island: Manning, 2018. – 534с. ([Електронний ресурс] - Режим доступу: [https://www.pdfdrive.com/concurrency-in-net-modern-patterns-of-concurren....](https://www.pdfdrive.com/concurrency-in-net-modern-patterns-of-concurren...))

Навчальний контент

6. Методика опанування навчальної дисципліни (освітнього компонента)

Назви розділів і тем	Кількість годин				
	Всього	у тому числі			
		Лекції	Практичні	Лабораторні роботи	СРС
1	2	3	4	5	6

Розділ 1. Структури паралельних комп'ютерних систем (ПКС)					
Тема 1.1 Базові структури ПКС	1,5	1	-	-	0,5
Тема 1.2 Організація пам'яті та зв'язків в ПКС	1,5	1	-	-	0,5
Разом за розділом 1	3	2	-	-	1
Розділ 2. Програмування паралельних обчислень					
Тема 2.1 Особливості програмування паралельних обчислень.	1,5	1	-	-	0,5
Тема 2.2 Життєвий цикл розробки паралельних програм	1,5	1	-	-	0,5
Разом за розділом 2	3	2	-	-	1
Розділ 3. Аналіз та побудова паралельних алгоритмів					
Тема 3.1 Концепція необмеженого паралелізму	1,5	1	-	-	0,5
Тема 3.2 Ярусно-паралельна форма представлення паралельного алгоритму.	1,5	1	-	-	0,5
Тема 3.3 Паралельні алгоритми рішення основних задач лінійної алгебри	3	2	-	-	1
Разом за розділом 3	6	4	-	-	2
Розділ 4. Паралельні процеси(потоки)					
Тема 4.1 Концепція процесу. Стан процесу. Операції з процесами	3	2			1
Тема 4.2 Процеси в сучасних мовах і бібліотеках паралельного програмування	10	4	-	2	4
Разом за розділом 4	13	6	-	2	5
Розділ 5. Комунікація процесів					
Тема 5.1 Види комунікації процесів	1,5	1	-	-	0,5
Тема 5.2 Моделі взаємодії процесів.	1,5	1	-	-	0,5
Разом за розділом 5	3	2	-	-	1
Розділ 6. Завдання взаємного виключення і синхронізації					
Тема 6.1 Постановка та загальна схема рішення задачі взаємного виключення і синхронізації	1,5	1			0.5
Тема 6.2 Алгоритми Деккера	1,5	1	-	-	0.5
Тема 6.3 Семафори	1,5	2	-		1

Тема 6.4 Мьютекси	1,5	1			0,5
Тема 6.5 Події	1,5	1	-		1
Тема 6.6 Критичні секції	3	2	-		1
Разом за розділом 6	12,5	8	-		4,5
Розділ 7. Монітори					
Тема 7.1 Концепція моніторів	3	2	-	-	1
Тема 7.2 Реалізація і застосування моніторів	3	2	-		1
Разом за розділом 7	6	4	-		2
Розділ 8. Взаємодія процесів через посилання повідомлень					
Тема 8.1 Загальна концепція механізму повідомлень	1,5	1	-	-	0,5
Тема 8.2 Особливості реалізації механізму повідомлень	1,5	1	-		0,5
Разом за розділом 8	3	2	-		1
Розділ 9 . Розробка програмного забезпечення для ВКС					
Тема 9.1 Розробка програмного забезпечення для ВКС зі спільною пам'яттю	30,5	4		12	14,5
Тема 9.2 Розробка програмного забезпечення для ВКС зі локальною пам'яттю	10	2		4	4
Разом за розділом 9	40,5	6		16	18,5
МКР	1				1
Екзамен	30				30
Всього в семестрі:	120	36	-	18	66

Лекційні заняття

№ лекції	Назва теми лекції та перелік основних питань
1	<p>Структури паралельних комп'ютерних систем (ПКС) Базові структури ПКС. Організація пам'яті та зв'язків в ПКС. ПКС з локальною пам'яттю. ПКС зі спільною пам'яттю . СРС: Виконати розрахунки топологічних характеристик ПКС ЛП (лінійна, кільцева, зірка, решітка, гіперкуб) .</p>

2	<p>Програмування паралельних обчислень</p> <p>Особливості завдання паралельного програмування: дослідження задач на паралелізм, розробка паралельних алгоритмів, програмування паралельних процесів, взаємодія процесів, рішення задачі взаємного виключення, синхронізація процесів, розміщення процесів по процесорах, планування виконання процесів, налагодження та виконання</p> <p>Особливості життєвого циклу розробки програм для паралельних систем. Етапи розробки: аналіз вимог, проектування, реалізація, тестування, виробництво, модифікація, супровід.</p> <p>СРС: Виконати аналіз життєвого циклу розробки програм для операції пошуку максимального елемента матриці .</p>
3	<p>Аналіз та побудова паралельних алгоритмів</p> <p>Концепція необмеженого паралелізму. Призначення та застосування [Ярусно-паралельна форма (ЯПФ) представлення паралельного алгоритму. Визначення та призначення ЯПФ. Правила будування ЯПФ. Параметри ЯПФ: ранг, ширина, висота .Застосування. Теорема Мунро-Петерсона. Лема Брента для встановлення зв'язку між обмеженим та необмеженим</p>
	<p>паралелізмом. Призначення. Приклади застосування. Теорема Мунро-Петерсона для оцінки арифметичних виразів. Застосування.</p> <p>СРС: Побудувати паралельний алгоритм для заданого виразу, визначити T_0, T_p. K_u</p>
4	<p>Паралельні алгоритми рішення основних задач лінійної алгебри</p> <p>Паралельні алгоритми для операцій над векторами та матрицями. Сортування. Знаходження максимального елемента. Рішення систем алгебраїчних рівнянь</p> <p>СРС: Розробити паралельні алгоритми для заданих векторно-матричних операцій</p>
5	<p>Концепція процесу.</p> <p>Види взаємодії процесів. Визначення процесу. Види стану процесу. Операції над процесом. Види процесів.</p> <p>Представлення процесів в ОС. Блок керування процесом. Процес та ресурс - фундаментальні поняття сучасних операційних систем. Керування процесом в ОС. Блок керування процесом</p> <p>Види взаємодії процесів. Взаємодія процесів. Комунікація процесів. Синхронізація процесів</p> <p>СРС: Виконати аналіз різних підходів до програмування процесів (потоків) в сучасних мовах та бібліотеках паралельного програмування</p>
6	<p>Процеси в сучасних мовах програмування. Мова Ада. Мова Java. Мова C# . СРС: Програмування потоків в мові Python</p>
7	<p>Процеси у спеціальних бібліотеках паралельного програмування. Бібліотека Win32. Бібліотека MPI. Бібліотека PVM. Бібліотека OpenMP</p> <p>СРС: Програмування потоків в бібліотеці PVM</p>

8	<p>Види синхронізації та взаємодії процесів Задачі взаємодії процесів: синхронізація та обмін даними. Особливості механізмів синхронізації. СРС: Особливості взаємодії процесів</p>
9	<p>Класичні задачі взаємодії процесів. Взаємне виключення. Виробник-Споживач. Читач - Письменник. Філософи що обідають. СРС: Завдання Завод-Магазін</p>
10	<p>Постанова та загальна схема рішення задачі взаємного виключення Постанова задачі .Загальна схема рішення. Критичні ділянки. Конструкції ВХОДКУ ті ВИХІДКУ. СРС: Алгоритми Декера 2 - 4</p>
11	<p>Семафори Загальна концепція семафорів. Семафорний тип. Операції з семафорами. Застосування семафорів. Реалізація. Приклади. СРС: Реалізація та застосування множинних семафорів</p>
12	<p>Мютекси Загальна концепція мютексів. Операції з мютексами. Застосування мютексів. Реалізація. Приклади. СРС: Реалізація та застосування мютексів в мові Python</p>
13	<p>Події Загальна концепція подій. Операції з полями. Застосування подій. Реалізація. Приклади. СРС: Реалізація та застосування подій в мові Python</p>
14	<p>Критичні секції Концепція критичних секцій. Застосування. Реалізація в мові C# та бібліотеці Win32 [1, с.100-101]. Розподілені змінні. Конструкції shared, atomic, volatile. СРС: Реалізація механізму критичних секцій в бібліотеці OpenMP [1, с. 127]</p>
15	<p>Концепція моніторів Загальна концепція моніторів. Дані та процедури. Особливості процедур. Рішення задачі взаємного виключення та синхронізації. СРС: Порівняння реалізацій.</p>
16	<p>Реалізація моніторів. Мова Ада: захищені модулі. Монітори в мові Java. Приклади застосування. [1, с. 104-110] СРС: Завдання Завод-Магазін]</p>
16	<p>Загальна концепція механізму повідомлень Операції Send та Receive. Блоковані та неблоковані операції передавання даних. СРС: види операцій передавання в бібліотеці MPI</p>
17	<p>Реалізації механізму повідомлень [Механізм рандеву у мовах Ада та Оккам. Бібліотека MPI СРС: Бібліотека PVM</p>

18	Розробка програмного забезпечення для ВКС Розробка програмного забезпечення для ВКС зі спільною пам'яттю. Розробка програмного забезпечення для ВКС зі локальною пам'яттю
----	---

Лабораторні заняття

Основні завдання циклу лабораторних занять - придбання студентами необхідних практичних навиків розробки паралельних програм, що базуються на концепції процесів, з застосуванням мов і бібліотек паралельного програмування Java, Ада, С#, WinAPI, OpenMP, MPI на базі сучасного комп'ютерного обладнання у вигляді потужних багатоядерних систем.

№ з/п	Назва лабораторної роботи	Кількість ауд. годин
1	Програмування потоків.	2
2	Розробка паралельних програм для ВКС з СП. Низкорівневі засоби	4
3	Розробка паралельних програм для ВКС з СП. Монітори.	4
4	Розробка паралельних програм для ВКС з СП. Open MP	4
5	Розробка паралельних програм для ВКС з ЛП.	4
	Разом:	18

7. Самостійна робота студента

№ з/п	Назва теми, що виноситься на самостійне опрацювання	Кількість годин СРС
1	Трансп'ютерні системи. Апаратне та програмне забезпечення. Мова Оккам	2
2	Сучасні розподілені (класстерні) системи. Структурна організація. Програмне забезпечення [www.top500.com]	2
3	Бібліотека PVM. Програмування потоків. Організація взаємодії потоків	2
4	Підготовка до лекційних та лабораторних занять	30
5	Підготовка до екзамену	30
	Разом	66

Політика та контроль

8. Політика навчальної дисципліни (освітнього компонента)

При зарахування та оцінювання лабораторних робіт беруться до уваги наступні чинники:

- Повнота виконання завдання на лабораторну роботу за індивідуальним варіантом;
- Своєчасність виконання лабораторної роботи згідно графіку;
- Самостійність виконання лабораторної роботи та відсутність ознак плагіату;
- Відповіді на питання щодо змісту лабораторної роботи під час її захисту.

При оцінюванні контрольних робіт до уваги приймаються:

- Правильність та повнота виконання завдань;
- Кількість виконаних завдань в умовах обмеженого часу;
- Самостійність виконання завдань та відсутність ознак плагіату;
- Кількість спроб виконання контрольних, які передують тій, що оцінюється.

Для підготовки до контрольних студенти отримують перелік теоретичних питань та зміст типових задач, які будуть у завданнях на контрольних.

При першій та другій атестації до уваги приймається кількість лабораторних робіт та контрольних робіт зарахованих на час проведення атестації.

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

PCO з дисципліни, семестровий контроль з якої передбачений у вигляді екзамену, розробляється за типом PCO-2 і складається з двох складових:

- стартової – призначена для оцінювання заходів поточного контролю впродовж семестру (**Rc**)
- екзаменаційної – призначена для оцінювання окремих запитань (завдань) на екзамені. (**Re**)

Рекомендований розмір стартової складової PCO дорівнює 60 балів, екзаменаційної складової – 40 балів. Якщо значення стартової складової >60, студент може не здавати екзамен. І цьому разі $R = Rc$.

Стартові бали формуються як сума рейтингових балів, отриманих здобувачем за результатами заходів поточного контролю, заохочувальних та штрафних балів.

Після оцінювання відповідей здобувача на екзамені (виконання екзаменаційної контрольної роботи) викладач підсумовує стартові бали та бали за екзамен, зводить до рейтингової оцінки та переводить до оцінок за університетською шкалою.

$$R = Rc + Re$$

Види контролю з навчальної дисципліни:

Лабораторні роботи:

Заплановано самостійне виконання від трьох до п'яти лабораторних робіт (за вибором). Темі лабораторних робіт узгоджені у часі та за змістом з темами лекцій. Виконання лабораторних робіт у повному обсязі дозволяє набути практичних навичок програмування потоків за допомогою сучасних мов та бібліотек паралельного програмування.

Поточний контроль:

Передбачено проведення модульної контрольної роботи (МКР)

Семестровий контроль

Екзамен проводиться для об'єктивного визначення додаткових знань, умінь та практичних навичок, отриманих за семестр

Семестровий рейтинг студента складається з балів, які він отримує за види робіт відповідно до таблиці 1.

Таблиця 1

Оцінювання окремих видів навчальної роботи студента (у балах)

Вид навчальної роботи	Всього за видом роботи
Виконання та захист лабораторної роботи № 1	10
Виконання та захист лабораторної роботи № 2	10
Виконання та захист лабораторної роботи № 3	10
Виконання та захист лабораторної роботи № 4	10
Виконання та захист лабораторної роботи № 5	10
Виконання лабораторних робіт	Rл 50
Модульна контрольна робота (МКР)	Rк 10
Усього за семестр	Rп= Rл + Rк 60
Екзамен	Re 40
Усього за семестр (R = Rп + Re)	100

Індивідуальний поточний рейтинг студента (**Rп**) складається з балів, які він отримує за виконання лабораторних робіт і МКР. Протягом семестру студенти виконують обрану кількість (3-5) лабораторних робіт. Максимальна кількість балів за кожну лабораторну роботу – 10. Бали нараховуються за:

- теоретична складова – 5 балів
- практична складова – 5 балів.

Максимальний можливий бал за лабораторну роботу – 10 балів.

Максимальна кількість за всі лабораторні роботи 50 балів.

Розрахунок розміру шкали (**R**) рейтингу.

Сума вагових балів контрольних заходів протягом семестру становить:

Rп= Rл + Rк , де Rп – семестровий рейтинг студента (МКР, лабораторні роботи).

Необхідною умовою допуску студента до екзамену є його індивідуальний семестровий рейтинг (**Rп**), не менший, ніж 60 балів, та відсутність заборгованості з лабораторних робіт та МКР. При невиконанні згаданих вимог студент до заліку не допускається.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

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

В рамках вивчення дисципліни “Програмне забезпечення високопродуктивних комп’ютерних систем” допускається зарахування додаткових балів (5 балів), одержаних в результаті дистанційних курсів на платформі “Coursera”, за умови попереднього погодження програми даного курсу з викладачем та за умови отримання офіційного сертифікату.

Робочу програму навчальної дисципліни (силабус):

Складено доцент, к.т.н, доцент, Корочкін Олександр Володимирович

Ухвалено кафедрою обчислювальної техніки (протокол № 13 від 10.05.2023)

Погоджено Методичною комісією факультету (протокол № 1 від 29.06.2023)

....