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

Кафедра прикладної математики

ТЕХНОЛОГІЯ БЛОКЧЕЙНУ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	***Другий (магістерський)***
Галузь знань	*F Інформаційні технології*
Спеціальність	*F1 Прикладна математика*
Освітня програма	*Машинне навчання та математичне моделювання*
Статус дисципліни	*Вибіркова*
Форма навчання	*Очна (денна)*
Рік підготовки, семестр	*1 курс, весняний семестр*
Обсяг дисципліни	*4 кредити ЄКТС / 120 год.: (30 год. – лекції, 16 год. — практичні заняття, 74 год. – СРС)*
Семестровий контроль/ контрольні заходи	*Залік, РГР, МКР*
Розклад занять	*http://rozklad.kpi.ua* *лекція – 1 раз на тиждень* *практичні заняття — 1 раз на два тижні*
Мова викладання	*Українська*
Інформація про керівника курсу / викладачів	Лектор: канд. техн. наук Кобринець Антон Костянтинович e-mail: info@kobrynets.com
Розміщення курсу	

Програма навчальної дисципліни

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

Метою навчальної дисципліни є засвоєння студентами концептуальних засад, математичного та криптографічного апарату, а також архітектурних принципів побудови технології блокчейну та розподілених реєстрів; формування навичок формалізації, моделювання та аналізу децентралізованих протоколів, механізмів консенсусу й смартконтрактів; розвиток здатності проєктувати та впроваджувати безпечні, стійкі до відмов розподілені системи у прикладних галузях.

Предмет навчальної дисципліни:

- криптографічні основи та структури даних блокчейну: криптографічні геш-функції, дерево Меркла, схеми цифрового підпису, асиметричне шифрування, модель UTXO та транзакційні моделі на основі облікових записів;
- розподілені системи та протоколи досягнення консенсусу: задачі візантійської відмовостійкості (BFT), класичні та накамотівські механізми досягнення консенсусу (Proof of Work, Proof of Stake, DPoS, PBFT), аналіз безпеки й стійкості до атак (атака 51%, selfish mining, long-range attacks);
- програмовані розподілені платформи та смартконтракти: віртуальні машини блокчейнів (EVM), мови опису логіки контрактів, шаблони проєктування децентралізованих додатків

(dApps), рішення з масштабування (Layer 2, rollups, sharding) та забезпечення конфіденційності (доведення з нульовим розголошенням – ZK-SNARKs).

Після засвоєння навчальної дисципліни студенти мають продемонструвати такі компетентності та програмні результати навчання за освітньою програмою:

Загальні компетентності

ЗК01. Здатність до самонавчання, пошуку, оброблення та інтелектуального аналізу інформації з різних джерел, вміння виявляти, ставити та вирішувати проблеми.

Фахові компетентності

ФК01. Здатність формалізувати та розв’язувати складні задачі й проблеми, які потребують оновлення й інтеграції знань, часто в умовах неповної, неточної чи недостатньої інформації та суперечливих вимог.

Програмні результати навчання

ПРН01. Використовувати та адаптовувати математичні теорії та моделі для забезпечення теоретичного підґрунтя розв'язання наукових та практичних задач.

ПРН02. Застосовувати існуючий математичний апарат, розробляти нові моделі, методи та алгоритми при вирішенні актуальних практичних задач широкого спектру.

2. Пререквізити та постреквізити дисципліни

Корисним для успішного опанування дисципліни будуть базові знання з баз даних, дискретної математики, уміння програмувати мовою Python.

3. Зміст навчальної дисципліни

РОЗДІЛ 1. КРИПТОГРАФІЧНІ ОСНОВИ, СТРУКТУРИ ДАНИХ ТА АРХІТЕКТУРА БЛОКЧЕЙНУ

Тема 1.1. Криптографічний фундамент: криптографічні хеш-функції, властивості односторонності та стійкості до колізій, схеми електронного цифрового підпису (ECDSA, Ed25519, Schnorr).

Тема 1.2. Структури даних у розподілених реєстрах: хеш-ланцюжки, бінарні дерева Меркла, префіксні дерева Меркла–Патрісії (Merkle Patricia Trie), верифікація доказів належності.

Тема 1.3. Організація мережі peer-to-peer (P2P), розповсюдження інформації (пліткарські протоколи / Gossip protocols) та моделі обліку транзакцій (UTXO проти Account-based).

РОЗДІЛ 2. МЕХАНІЗМИ ДОСЯГНЕННЯ КОНСЕНСУСУ ТА КРИПТОЕКОНОМІЧНА БЕЗПЕКА

Тема 2.1. Проблема розподіленого консенсусу: теорема FLP, CAP-теорема, класична задача візантійських генералів та протоколи візантійської відмовостійкості (PBFT).

Тема 2.2. Накамотівський консенсус та механізм Proof of Work (PoW): динамічне регулювання складності, форки, правило найдовшого/найважчого ланцюга (Nakamoto consensus).

Тема 2.3. Механізми Proof of Stake (PoS) та його варіації: проблема «nothing at stake», фіналізація за Каспером (Casper FFG/CBC), делегований консенсус (DPoS).

Тема 2.4. Криптоекономічний аналіз та вектори атак: атака 51%, егоїстичний майнінг (Selfish Mining), MEV (Maximal Extractable Value), Sybil-атаки та стійкість до них.

РОЗДІЛ 3. ПРОГРАМОВАНІ БЛОКЧЕЙНИ, СМАРТКОНТРАКТИ ТА МАСШТАБУВАННЯ

Тема 3.1. Архітектура децентралізованих обчислювальних середовищ: віртуальна машина Ethereum (EVM), механізм розрахунку газу (Gas model), життєвий цикл смартконтрактів.

Тема 3.2. Розробка та шаблони проєктування смартконтрактів: мова Solidity/Vyper, стандарти токенів (ERC-20, ERC-721, ERC-1155), оракули (Oracles) та зв'язок із зовнішнім світом.

Тема 3.3. Проблеми безпеки смартконтрактів: атаки повторного входу (Reentrancy), переповнення, флеш-кредити (Flash Loans), методи статичного аналізу та формальної верифікації коду.

Тема 3.4. Масштабування розподілених систем та конфіденційність: шардинг (Sharding), Layer 2 протоколи (State Channels, Optimistic Rollups, ZK-Rollups) та доведення з нульовим розголошенням (ZK-SNARKs / ZK-STARKs).

4. Навчальні матеріали та ресурси

Базова література

1. Narayanan A., Bonneau J., Felten E., Miller A., Goldfeder S. *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. Princeton University Press, Draft version, 2016. 308 p. URL: https://economics.princeton.edu/books/bitcoin-and-cryptocurrency-technologies-a-comprehensive-introduction/ (access date: 25.12.2024).

2. Antonopoulos A. M., Wood G. *Mastering Ethereum: Building Smart Contracts and DApps*. O'Reilly Media / Open Source Edition (CC BY-NC-ND 4.0 / CC BY-SA 4.0). URL: https://github.com/ethereumbook/ethereumbook (access date: 25.12.2024).

3. Shi E. *Foundations of Distributed Consensus and Blockchains*. Book manuscript, 2020. 297 p. URL: http://elaineshi.com/docs/blockchain-book.pdf (access date: 25.12.2024).

Допоміжна література

4. Nakamoto S. *Bitcoin: A Peer-to-Peer Electronic Cash System*. 2008. 9 p. URL: https://bitcoin.org/bitcoin.pdf (access date: 25.12.2024).

5. Buterin V. *Ethereum Whitepaper: A Next-Generation Smart Contract and Decentralized Application Platform*. 2014. URL: https://ethereum.org/en/whitepaper/ (access date: 25.12.2024).

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Навчання здійснюється на основі студентоцентрованого підходу та стратегії взаємодії викладача та студента для засвоєння студентами матеріалу та розвитку у них практичних навичок. Для лекційних занять використовуються пояснювально-ілюстративний метод та метод проблемного виконання.

За дистанційного режиму навчання заняття проводять за допомогою платформи для проведення онлайн-зустрічей Zoom.

Лекційні заняття

№ з/п	Назва теми лекції та перелік основних питань
1	**Вступ до технології блокчейну та децентралізованих систем** Основні питання: еволюція систем передачі цінності та електронних грошей; проблема подвійного витрачання (double-spending); архітектурні відмінності між централізованими, розподіленими та децентралізованими реєстрами.
2	**Криптографічні основи: хеш-функції** Основні питання: криптографічні хеш-функції (SHA-256, Keccak-256); властивості односторонності, лавинного ефекту та стійкості до колізій; застосування хешування для зв'язування блоків і цілісності даних.
3	**Розширені структури даних: Merkle Patricia Trie** Основні питання: префіксні дерева (Radix Trie); комбінована структура Merkle Patricia Trie (MPT); організація зберігання стану облікових записів, сховища контрактів та транзакцій у Ethereum.
4	**Моделі обліку транзакцій: UTXO та Account-based** Основні питання: модель нестрачених виходів транзакцій (UTXO, приклад Bitcoin); модель на основі облікових записів (Account-based, приклад Ethereum); порівняння простору станів, паралелізації та конфіденційності.

№ з/п	Назва теми лекції та перелік основних питань
5	**Мережева взаємодія: P2P та протоколи поширення даних** Основні питання: топологія однорангових мереж (P2P); протоколи типу пліткарства (Gossip protocols); структура та керування пулом непідтверджених транзакцій (mempool); синхронізація вузлів.
6	**Теорія розподіленого консенсусу та візантійська стійкість** Основні питання: модель розподілених обчислень; теорема FLP про неможливість консенсусу в асинхронних системах; CAP-теорема; класична задача візантійських генералів; протокол PBFT (Practical Byzantine Fault Tolerance).
7	**Механізм Proof of Stake (PoS) та проблема «Nothing at Stake»** Основні питання: перехід до алгоритмів доведення частки володіння (PoS); алгоритми псевдовипадкового вибору валідаторів; проблема «Nothing at Stake» та методи її усунення (slashing mechanisms).
8	**Сучасні варіації PoS та моделі фіналізації** Основні питання: консенсусні протоколи з явною фіналізацією (Casper FFG, Tendermint BFT); делегований консенсус (DPoS); механізми ліквідного стейкінгу (Liquid Staking) та їхній криптоекономічний вплив.
9	**Криптоекономіка та вектори атак на рівні протоколу** Основні питання: стимули та штрафи у розподілених протоколах; Sybil-атаки; Eclipse-атаки; Long-range attacks у PoS; проблема максимальної вилученої вартості (MEV — Maximal Extractable Value).
10	**Архітектура смартконтрактів та віртуальна машина EVM** Основні питання: детермінізм у розподілених обчисленнях; архітектура Ethereum Virtual Machine (EVM); модель пам’яті (Stack, Memory, Storage); виконання та опкоди байт-коду.
11	**Економіка виконання смартконтрактів (модель Gas)** Основні питання: вирішення проблеми зупинки (Halting Problem); обчислювальна складність та концепція газу (Gas); структура транзакційних комісій (EIP-1559: base fee, priority fee, gas limit); оптимізація витрат ресурсів.
12	Модульна контрольна робота.
13	**Оракули та взаємодія із зовнішніми даними** Основні питання: ізольованість блокчейну та «проблема оракула» (Oracle Problem); централізовані та децентралізовані мережі оракулів (Chainlink); маніпуляції ціновими даними на базі децентралізованих бірж (DEX).
14	**Безпека смартконтрактів та аналіз вразливостей** Основні питання: класичні вразливості: атака повторного входу (Reentrancy attack, кейс The DAO); переповнення цілих чисел (Underflow/Overflow); помилки керування доступом.
15	**Масштабування блокчейнів: архітектурні рішення Layer 1** Основні питання: трилема блокчейну (децентралізація, безпека, масштабованість); шардинг (Sharding); перехід до модульних архітектур (розділення рівнів виконання, консенсусу та доступності даних – Data Availability).

Практичні заняття

№ з/п	Назва теми заняття та перелік основних питань
1	**Асиметрична криптографія, хешування та генерація ключів у блокчейні** **Мета:** Опанувати базовий криптографічний стек розподілених систем, навчитися генерувати пари ключів, створювати адреси та верифікувати транзакційні підписи. **Основні завдання:** • Реалізація та дослідження криптографічних геш-функцій (SHA-256, Keccak-256), демонстрація лавинного ефекту. • Генерація ключів за алгоритмом ECDSA (крива secp256k1) та формування блокчейн-адрес. • Підписання довільних повідомлень і транзакцій приватним ключем та алгоритмічна перевірка підпису публічним ключем. • Порівняльний аналіз стандартів підписів: Schnorr-підписи та агрегація мультипідписів (BLS).
2	**Структури даних: побудова дерева Меркла та верифікація включення транзакцій** **Мета:** Розробити алгоритмічні моделі структур даних блокчейну та реалізувати механізм спрощеної верифікації платежів (SPV). **Основні завдання:** • Програмна реалізація бінарного дерева Меркла для довільного масиву транзакцій. • Алгоритм обчислення кореня Меркла (Merkle Root) та формування доказів включення (Merkle Proof). • Реалізація клієнтської функції валідації входження транзакції в блок без завантаження повного реєстру. • Моделювання префіксного дерева Меркла–Патрісії (Merkle Patricia Trie) для збереження стану акаунтів.
3	**Моделювання механізмів консенсусу: Proof of Work та динамічне коригування складності** **Мета:** Створити програмну симуляцію накамотівського консенсусу та дослідити поведінку мережі за зміни обчислювальної потужності. **Основні завдання:** • Програмування спрощеного PoW-рушія з пошуком валідного нонсу (nonce) за заданою маскою цільового гешу (target). • Реалізація алгоритму динамічного коригування складності (difficulty adjustment) залежно від швидкості появи блоків. • Симуляція появи та вирішення випадкових форків за правилом найдовшого/найважчого ланцюга. • Кількісний аналіз залежності часу генерації блоку від загального хешрейту симульованої мережі.
4	**Симуляція векторів атак та аналіз стійкості протоколу (Selfish Mining, атака 51%)** **Мета:** Дослідити вразливості консенсусних протоколів шляхом моделювання стратегічної поведінки нечесних майнерів/валідаторів. **Основні завдання:** • Програмна симуляція атаки егоїстичного майнінгу (Selfish Mining за моделлю Еяля–Сірара). • Обчислення граничної частки обчислювальної потужності, за якої стратегія егоїстичного майнінгу стає вигіднішою за чесну. • Моделювання сценарію атаки 51% та подвійного витрачання (double-spending attack) із розрахунком необхідної глибини підтверджень. • Оцінка ризиків та симуляція проблеми «Nothing at Stake» для наївного алгоритму Proof of Stake.

№ з/п	Назва теми заняття та перелік основних питань
5	**Розробка, компіляція та розгортання смартконтрактів мовою Solidity** **Мета:** Опанувати повний життєвий цикл розробки децентралізованої бізнес-логіки в середовищі віртуальної машини EVM. **Основні завдання:** • Створення базового смартконтракту з використанням структур даних, модифікаторів доступу та подій (events) у середовищі Remix / Hardhat. • Тестування функцій запису та читання стану, моделювання взаємодії кількох облікових записів. • Профілювання використання газу (Gas profiling) для різних операцій зберігання (Storage vs Memory). • Розгортання та взаємодія зі смартконтрактом у локальній тестовій мережі (Anvil / Ganache).
6	**Реалізація стандартів токенів (ERC-20/ERC-721) та інтеграція з оракулами** **Мета:** Навчитися створювати стандартизовані цифрові активи та підключати зовнішні канали даних до смартконтрактів. **Основні завдання:** • Розробка та модульне тестування смартконтракту взаємозамінного токена стандарту ERC-20 (із логікою mint, burn, transfer, approve). • Створення базового смартконтракту невзаємозамінного токена (NFT) стандарту ERC-721 із метаданими. • Інтеграція смартконтракту з децентралізованим оракулом (Chainlink Price Feeds) для отримання зовнішніх цін. • Моделювання захисту від застарілих даних (**stale data) та відмов оракула.**
7	**Аудит безпеки, виявлення та експлуатація вразливостей смартконтрактів** **Мета:** Опанувати методики аналізу коду на наявність критичних вразливостей та навчитися захищати децентралізовані протоколи від атак. **Основні завдання:** • Створення тестового сценарію та практична реалізація атаки повторного входу (Reentrancy attack) на вразливий контракт сховища. • Застосування патерну Checks-Effects-Interactions та блокувальників повторного входу (ReentrancyGuard) для усунення вразливості. • Дослідження логічних помилок керування доступом та ризиків переповнення/арифметичних операцій. • Запуск та аналіз звітів інструментів автоматизованого статичного аналізу (Slither / Mythril).
8	**Протоколи масштабування Layer 2 та криптографічні доведення з нульовим розголошенням (ZKP)** **Мета:** Опанувати принципи роботи рішень другого рівня для масштабування розподілених систем, навчитися створювати та верифікувати платіжні канали, а також генерувати і перевіряти базові доведення з нульовим розголошенням. **Основні завдання:** • Програмна симуляція двостороннього платіжного каналу (State Channel) із мультипідписом, оновленням балансів поза мережею (off-chain) та фінальним закриттям ончейн. • Моделювання механізму оскарження некоректного стану (Fraud Proof) для архітектури Optimistic Rollup. • Складання арифметичної схеми для простого твердження за допомогою спеціалізованого фреймворку (наприклад, Circom / SnarkJS). • Генерація криптографічного доведення (ZK-SNARK Proof) поза блокчейном без розкриття приватних вхідних даних (witness). • Розгортання смартконтракту-верифікатора в EVM-середовищі та ончейн-перевірка валідності згенерованого доведення з нульовим розголошенням.

6. Самостійна робота студента

Студент повинен завчасно готуватись до лекцій. Перед лекціями потрібно повторити вивчений раніше теоретичний матеріал, підготувати питання щодо нової теми для обговорення на лекції. Перед практичними заняттями потрібно повторити відповідний теоретичний матеріал.

№ з/п	Вид самостійної роботи	Кількість годин СРС
1.	Виконання РГР	30
2.	Підготовка до лекцій	15
3.	Підготовка до практичних занять	16
4.	Опрацювання лекційного матеріалу	5
5.	Підготовка до МКР	2
6.	Підготовка до заліку	6
	Загалом	74

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Відвідування занять

Відповідно до Наказу 1-273 від 14.09.2020 р. заборонено оцінювати присутність або відсутність здобувача на авдиторному занятті. Матеріал занять, які були з тих чи тих причин пропущено, потрібно опановувати самостійно. У будь-якому випадку студентам рекомендується відвідувати всі види занять, оскільки на них викладається теоретичний матеріал та розвиваються навички, потрібні для виконання завдань на практичних заняттях. Система оцінювання орієнтована на отримання балів за активність студента, а також виконання завдань, що розвивають практичні уміння та навички.

Правила написання розрахунково-графічної роботи

Студенти виконують РГР протягом семестру, розв'язуючи задачі, моделюючи протоколи та програмуючи ключові компоненти блокчейн-систем. Виконання РГР здійснюється індивідуально мовою програмування Python, Solidity (або іншими погодженими мовами/інструментами). Розрахунково-графічна робота виконується поетапно відповідно до змістових модулів курсу і складається з чотирьох етапів:

1. **Етап 1. Криптографічний базис та структури даних блокчейну:** побудова та оптимізація геш-ланцюгів, алгоритмічна реалізація дерева Меркла, формування та валідація Merkle proof, генерація та верифікація цифрових підписів.

2. **Етап 2. Розподілений консенсус та криптоекономічні симуляції:** моделювання роботи протоколів консенсусу (Proof of Work, Proof of Stake), симуляція динаміки складності та форків, кількісний аналіз стійкості мережі до атак (атака 51%, selfish mining).

3. **Етап 3. Розробка та тестування смартконтрактів:** проектування смартконтрактів мовою Solidity, реалізація логіки децентралізованої взаємодії, впровадження базових токенів (ERC-20/ERC-721), інтеграція з оракулами та оптимізація споживання газу (Gas profiling).

4. **Етап 4. Безпека смартконтрактів та аналіз вразливостей:** виявлення, експлуатація (у локальному тестовому середовищі) та усунення критичних вразливостей (Reentrancy, маніпуляції логікою викликів), застосування інструментів статичного аналізу коду.

Результати кожного етапу РГР здаються викладачу в електронному вигляді у встановлені календарним планом терміни. Під час виконання РГР допустиме використання допоміжних матеріалів, але суворо заборонено обговорювати задачі з колегами чи питати порад у третіх осіб.

Правила складання модульної контрольної роботи

Модульну контрольну роботу студенти виконують:

– в умовах очного навчання — під час відповідного лекційного заняття протягом фіксованого часу і здають у паперовому вигляді;

– в умовах дистанційного навчання — у позааудиторний час і надсилають викладачеві в електронній формі.

Під час виконання МКР допустиме використання допоміжних матеріалів, але суворо заборонено обговорювати задачі з колегами чи питати порад у третіх осіб.

Студент може спробувати переписати пропущену МКР тільки в тому випадку, якщо її було пропущено з поважної причини, що підтверджено документально. Переписування МКР відбувається за домовленістю з викладачем.

Пропущені контрольні заходи

Пропущений екзамен не зараховується; у такому випадку студент отримує запис у відомості «не з'явився» та повинен складати екзамен на додатковій сесії.

Академічна доброчесність

Політика та принципи академічної доброчесності визначено в розділі 3 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: https://kpi.ua/code.

Норми етичної поведінки

Норми етичної поведінки студентів визначено в розділі 2 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: https://kpi.ua/code.

Процедура оскарження результатів контрольних заходів

Студенти мають можливість поставити будь-яке питання, яке стосується процедури проведення та/або оцінювання контрольних заходів, та очікувати, що воно буде розглянуто згідно з наперед визначеними процедурами. Студенти мають право оскаржити результати контрольних заходів, але обов'язково аргументовано, пояснивши, із яким критерієм не погоджуються відповідно до оціночного листа та/або зауважень.

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

Поточний контроль: РГР, МКР.

Календарний контроль: проводиться двічі на семестр на 7-ому та на 13-ому тижнях, як моніторинг поточного стану виконання вимог силабуса; студент отримує «зараховано», якщо його поточний рейтинг не менший від 50% від максимальної кількості балів, яку може отримати студент до даного календарного контролю.

Семестровий контроль: залік.

Умови допуску до семестрового контролю: виконання МКР та РГР.

Рейтинг студента з кредитного модуля складається з:

1) балів за виконання та захист розрахунково-графічної роботи (РГР);
2) балів за виконання модульної контрольної роботи (МКР).

СИСТЕМА РЕЙТИНГОВИХ БАЛІВ

1. Розрахунково-графічна робота (РГР)

Контроль знань здійснюється на основі виконання РГР, яка розподілена на чотири етапи відповідно до змістових модулів курсу. Сумарна кількість балів за всі етапи становить **80 балів**.

Кожний етап містить 3 теоретичні та 2 програмні задачі. Кожна задача оцінюється в **4 бали**.

Критерії оцінювання кожної окремої задачі:

4 бали – задачу виконано повністю;

для теоретичних задач: отримано правильний аналітичний розв'язок, наведено логічне обґрунтування;

для програмних задач: код є робочим, алгоритм відповідає теоретичній моделі, результати симуляції коректно інтерпретовані;

3 бали – задачу виконано з незначними вадами;

для теоретичних задач: правильний хід розв'язання при наявності некритичних арифметичних помилок;

для програмних задач: алгоритм загалом правильний, але код потребує виправлення для граничних випадків або відсутня візуалізація результатів.

0 балів – задачу не виконано, або розв'язок містить принципові помилки, що свідчать про нерозуміння концепції.

2. Модульна контрольна робота (МКР)

Студенти пишуть модульну контрольну роботу під час лекційного заняття. Вона містить чотири задачі, кожна з яких оцінюється в 5 балів. Загальна сума балів за МКР становить **20 балів**.

Критерії оцінювання розв'язків кожної окремої задачі:

5 балів – задачу розв'язано правильно;

3-4 бали – задачу розв'язано майже правильно, можливі незначні вади чи арифметичні помилки;

0 балів – розв'язок відсутній.

Розрахунок шкали (R) рейтингу:

Рейтингова шкала з дисципліни складає

$$R = R_C = 80 \text{ балів} + 20 \text{ балів} = 100 \text{ балів}.$$

Із цієї суми за семестр студент може набрати певну кількість балів r_C.

Неодмінною умовою допуску до заліку є виконання модульної контрольної роботи та розрахунково-графічної роботи. Інакше студент у першу відомість отримує оцінку «не допущено», а потім повинен виконати ці умови (написати МКР, захистити РГР та набрати бали для допуску). Виконавши умови допуску до заліку, студент під час першого та/або другого перескладання пише залікову контрольну роботу.

Якщо наприкінці семестру студент набрав не менше 60 рейтингових балів, а також виконав умови допуску до заліку, він отримує залік «автоматом» відповідно до таблиці.

Студент може спробувати підвищити свою оцінку шляхом написання залікової контрольної роботи, при цьому його бали, отримані за семестр, зберігаються, а з двох отриманих студентом оцінок виставляється вища.

У випадку, якщо сума рейтингових балів менша від 60, але виконано умови допуску до семестрового контролю, студент виконує залікову контрольну роботу, при цьому його рейтинг анулюють, після чого бали нараховують за результатами виконання залікової контрольної роботи.

Залікова контрольна робота містить 4 задачі. Відповідь на кожну задачу оцінюють у 25 балів.

Критерії оцінювання відповідей на кожну задачу:

25 балів — правильна та змістовна відповідь;

23–24 бали — відповідь правильна, змістовна, але має незначні вади;

19–22 балів — відповідь містить незначні помилки чи є неповною;

15–18 балів — відповідь містить кричущі помилки чи є неповною;

0 балів — немає відповіді.

Максимальна кількість балів за модульний контроль:
25 балів х 4 задачі = **100 балів**.

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

Робочу програму навчальної дисципліни (силабус):

Склав доцент кафедри ПМА, канд. техн. наук Антон КОБРИНЕЦЬ

Ухвалено кафедрою ПМА (протокол № 10 від 18.02.2026)

Погоджено Методичною комісією факультету прикладної математики (протокол № 7 від 23.02.2026)