



Хмарні обчислення

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Другий (магістр)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>121 Інженерія програмного забезпечення</i>
Освітня програма	<i>Інженерія програмного забезпечення інформаційних систем</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>Очна (денна)/заочна</i>
Рік підготовки, семестр	<i>2 курс, 3 семестр</i>
Обсяг дисципліни	<i>Лекції: 36 год., лабораторні: 18 год., самостійна робота: 96 год. 5 кредитів</i>
Семестровий контроль/ контрольні заходи	<i>Екзамен</i>
Розклад занять	<i>Згідно розкладу на весняний семестр поточного навчального року (rozklad.kpi.ua)</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	Лектор: <i>професор Жаріков Е.В.</i> Практичні / Семінарські: <i>професор Жаріков Е.В.</i>
Розміщення курсу	<i>Moodle, https://do.ipk.kpi.ua/course/view.php?id=8063</i>

Програма навчальної дисципліни

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

Мета вивчення дисципліни – набуття ключових фахових компетентностей, теоретичних знань і практичних навичок з розроблення програмного забезпечення на базі хмарних технологій. Предметом вивчення дисципліни є методи і технології розроблення та експлуатації програмного забезпечення для віртуалізованих і хмарних систем, засоби реалізації циклу розроблення програмного забезпечення в ІТ-інфраструктурі центрів оброблення даних (ЦОД) та у хмарних середовищах.

Завдання вивчення дисципліни:

- освоєння теоретичних понять і практичних навичок у галузі розроблення і експлуатації програмного забезпечення для хмарних систем;
- ознайомлення з сучасними методами і технологіями розробки програмного забезпечення для віртуалізованих і хмарних систем;
- набуття навичок реалізації технологій аналізу, розробки та забезпечення якості програмного забезпечення в умовах хмарних обчислень.

Мета навчальної дисципліни.

Метою навчальної дисципліни є формування у здобувачів здатностей:

- проектувати та забезпечувати якість програмного забезпечення для хмарних обчислень;

- використовувати сучасні технології та програмні засоби для побудови застосунків у хмарних середовищах;
- використовувати сучасні технології балансування та кешування;
- розробляти архітектури приватних і гібридних хмарних середовищ для вирішення задач підприємства;
- використовувати хмарні сервіси, що надають провідні ІТ-компанії для вирішення задач підприємства;
- застосовувати концепції хмарної безпеки: конфіденційність, цілісність, доступність, автентифікацію, авторизацію, аудит.

Дисципліна формує та закріплює у здобувачів такі загальні та спеціальні (фахові) компетентності: ЗК 01. Здатність до абстрактного мислення, аналізу та синтезу. ЗК 02. Здатність спілкуватися іноземною мовою як усно, так і письмово. ЗК 05. Здатність генерувати нові ідеї (креативність). ФК02 Здатність розробляти і реалізовувати наукові та/або прикладні проекти у сфері інженерії програмного забезпечення. ФК03 Здатність проектувати архітектуру програмного забезпечення, моделювати процеси функціонування окремих підсистем і модулів. ФК05 Здатність розробляти, аналізувати та застосовувати специфікації, стандарти, правила і рекомендації в сфері інженерії програмного забезпечення. ФК16 Здатність розробляти програмне забезпечення оброблення даних в GRID та хмарних сервісах.

Основні завдання навчальної дисципліни.

Після засвоєння дисципліни «Хмарні обчислення» результатами навчання є:

1) знання:

- сучасних теоретичних, методичних і алгоритмічних основ розробки програмного забезпечення хмарних систем для їх використання під час розв'язання прикладних і наукових задач;
- визначальні характеристики, сервісні моделі та моделі розгортання хмарних обчислень;
- з моделювання стану та поведінки складних об'єктів комп'ютерної галузі в процесі тестування відповідних програмних і технічних систем при забезпеченні якості і надійності розроблених інформаційних систем і технологій;
- математичних моделей і методів аналізу та синтезу сучасних систем управління та їх фундаментальних властивостей при проектуванні програмного забезпечення для дослідження характеристик і поведінки складних об'єктів і систем;
- архітектури та стандартів компонентних моделей, комунікаційних засобів і розподілених обчислень, уміння розв'язувати проблеми масштабованості, підтримки віддалених компонентів і взаємодії різних програмних платформ у віртуалізованих і хмарних інформаційних системах.

2) уміння:

- використовувати методи та методики проведення наукових та прикладних досліджень;

- розв’язувати прикладні задачі в галузі проектування, налаштування та експлуатації програмного забезпечення хмарних систем;
- використовувати методи проведення досліджень та вміти аналізувати складність технічних систем, розуміти складність задач оптимізації цих систем та їх елементів, та вдосконалювати методики їх проведення;
- застосовувати різні парадигми програмування: структурне, об’єктно-орієнтоване, функціональне, логічне, з відповідними моделями, методами та алгоритмами обчислень, структурами даних і механізмами управління під час створення та удосконалення віртуалізованих і хмарних систем;
- застосовувати технічні і програмні комунікаційні засоби та засоби з’єднання ресурсів в гібридних та розподілених хмарних системах;
- проектувати архітектуру програмного забезпечення для підтримки хмарних обчислень з використанням сучасних апаратно-програмних засобів із забезпеченням якості, надійності, доступності та безпеки;
- освоювати інноваційні хмарні технології, оцінювати їх та використовувати з метою розробки архітектури та проектування хмарних послуг.

3) досвід:

- проектування архітектури хмарного програмного забезпечення, розроблення сучасних інформаційних технологій та програмних систем для реалізації хмарних обчислень;
- здатність використовувати професійні знання й уміння в галузі практичного використання хмарних технологій.

Результати навчання, на формування та покращення яких спрямована дисципліна: ПРН02 оцінювати і вибирати ефективні методи і моделі розроблення, впровадження, супроводу програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу; ПРН04 виявляти інформаційні потреби і класифікувати дані для проектування програмного забезпечення; ПРН07 аналізувати, оцінювати і застосовувати на системному рівні сучасні програмні та апаратні платформи для розв’язання складних задач інженерії програмного забезпечення.

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

Для успішного засвоєння дисципліни здобувач повинен володіти освітніми компонентами «Побудова компіляторів», «Інфраструктура інформаційних систем». Отримані в результаті засвоєння дисципліни «Хмарні обчислення» теоретичні знання та практичні уміння можуть бути корисні для проведення наукових досліджень за темою дисертації.

3. Зміст навчальної дисципліни

Тема 1. Екосистема хмарних обчислень.

Тема 2. Розподілені обчислення у хмарній інфраструктурі.

Тема 3. Віртуалізація ресурсів в інфраструктурі хмарних ЦОД.

Тема 4. Моделі зберігання даних, файлові системи, сховища та бази даних у хмарі.

Тема 5. Розробка рішень для зберігання даних у хмарі AWS.

Тема 6. Розробка REST API.

Тема 7. Розробка безсерверних рішень на основі подій.

Тема 8. Реалізація контейнерів та контейнерних сервісів.

Тема 9. Кешування інформації для забезпечення масштабованості.

Тема 10. Розробка за допомогою служб обміну повідомленнями.

Тема 11. Консенсус, координація, балансування навантаження та робочі процеси у хмарних обчисленнях.

Тема 12. Розробка безпечних застосунків для хмарної інфраструктури.

Тема 13. Автоматизація розгортання за допомогою конвеєрів CI/CD.

4. Навчальні матеріали та ресурси

Основна література

1. Nayyar, Anand. Handbook of Cloud Computing: Basic to Advance research on the concepts and design of Cloud Computing. BPB Publications, 2019.
2. Simmon E. Evaluation of Cloud Computing Services Based on NIST SP 800-145 //NIST Special Publication. – 2018. – Т. 500. – С. 322.
3. Marinescu, D. C. (2022). Cloud computing: theory and practice. Morgan Kaufmann.
4. Barroso, L. A., Hölzle, U., & Ranganathan, P. (2018). The datacenter as a computer: Designing warehouse-scale machines. *Synthesis Lectures on Computer Architecture*, 13(3), i-189.
5. a, Y., Shen, Z., & Shen, J. (2024). Cloud Computing and Hyperscale Data Centers: A Comparative Study of Usage Patterns. *Journal of Theory and Practice of Engineering Science*, 4(06), 11-19.
6. Rittinghouse, J. W., & Ransome, J. F. (2017). Cloud computing: implementation, management, and security. CRC press.
7. Erl, Thomas, Robert Cope, and Amin Naserpour. Cloud computing design patterns. New York, NY: Prentice Hall, 2015.
8. Kavis, Michael J. Architecting the cloud: design decisions for cloud computing service models (SaaS, PaaS, and IaaS). John Wiley & Sons, 2014.
9. Alonso, J., Orue-Echevarria, L., Casola, V., Torre, A. I., Huarte, M., Osaba, E., & Lobo, J. L. (2023). Understanding the challenges and novel architectural models of multi-cloud native applications—a systematic literature review. *Journal of Cloud Computing*, 12(1), 6.

Додаткова література

1. Kachris, C., Falsafi, B., & Soudris, D. (Eds.). (2019). *Hardware Accelerators in Data Centers*. Springer International Publishing.

2. Mangalampalli, S., Sree, P. K., Swain, S. K., & Karri, G. R. (2023). Cloud computing and virtualization. *Convergence of Cloud with AI for Big Data Analytics: Foundations and Innovation*, 13-40.
3. Gupta, N., & Sohal, A. (2022). Cloud Computing: Evolution, Research Issues, and Challenges. *Emerging Computing Paradigms: Principles, Advances and Applications*, 1-17.
4. Longbottom, Clive. *The Evolution of Cloud Computing: How to plan for change*. BCS Learning & Development Limited, 2017.
5. Press, Cisco. "Data Center Virtualization Fundamentals.", 2014.
6. Zbakh, Mostapha, et al. "Cloud Computing and Big Data: Technologies, Applications and Security.", 2018.
7. Agboola, T. O., Mezue, F. C. T., Adebayo, S. B., Adegede, J., & Oyeniran, O. C. (2024). Technical Challenges and Solutions to TCP in Data Center.
8. Patnaik, S., Yang, X. S., Tavana, M., Popentiu-Vlădicescu, F., & Qiao, F. (Eds.). (2018). *Digital business: business algorithms, cloud computing and data engineering* (Vol. 21). Springer.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Структура

Назви розділів і тем	Всього	Лабораторні (комп'ютерний практикум)		
		Лекції		СРС
	150	36	18	96
Тема 1. Екосистема хмарних обчислень.		2		
Тема 2. Розподілені обчислення у хмарній інфраструктурі.		2	1	
Тема 3. Віртуалізація ресурсів в інфраструктурі хмарних ЦОД.		3	1	
Тема 4. Моделі зберігання даних, файлові системи, сховища та бази даних у хмарі.		3		
Тема 5. Розробка рішень для зберігання даних у хмарі AWS.		3	2	
Тема 6. Розробка REST API.		3	2	
Тема 7. Розробка безсерверних рішень на основі подій.		3	2	
Тема 8. Реалізація контейнерів та контейнерних сервісів.		3	2	
Тема 9. Кешування інформації для забезпечення масштабованості.		3	2	
Тема 10. Розробка за допомогою служб обміну повідомленнями.		3	2	
Тема 11. Консенсус, координація, балансування навантаження та робочі процеси у хмарних обчисленнях.		3	1	
Тема 12. Розробка безпечних		3	1	

застосунків для хмарної інфраструктури.					
Тема 13. Автоматизація розгортання за допомогою конвеєрів CI/CD.		2		2	

Лекційні заняття

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, посилання на літературу та завдання на СРС)
1.	<p><i>Лекція 1. Екосистема хмарних обчислень.</i></p> <ul style="list-style-type: none"> – Огляд інфраструктури AWS. – Запуск програм у хмарі. – Вступ до Amazon EC2, еластичного балансування навантаження та автоматичного масштабування Amazon EC2. – Вступ до безсерверних обчислень з AWS Lambda. – • Від монолітів до мікросервісів.
2.	<p><i>Лекція 2. Розподілені обчислення у хмарній інфраструктурі.</i></p> <ul style="list-style-type: none"> – Життєвий цикл розробки систем. – Паралелізм на рівні даних, потоків та завдань. – «М'яка» модульність та «примусова» модульність. – Основи роботи з AWS SDK.
3.	<p><i>Лекція 3. Віртуалізація ресурсів в інфраструктурі хмарних ЦОД.</i></p> <ul style="list-style-type: none"> – Віртуалізація ресурсів та гіпервізори. – Повна віртуалізація, вкладена та паравіртуалізація. – Апаратна підтримка віртуалізації. – Програмне забезпечення для віртуалізації.
4.	<p><i>Лекція 4. Моделі зберігання даних, файлові системи, сховища та бази даних у хмарі.</i></p> <ul style="list-style-type: none"> – Моделі зберігання, файлові системи та бази даних. – Помилки у розподілених обчисленнях та теорема CAP. – Бази даних NoSQL. – Надійність сховища у великих масштабах ЦОД. – Локальність диска проти локальності даних у комп'ютерних хмарах.
5.	<p><i>Лекція 5. Розробка рішень для зберігання даних у хмарі AWS.</i></p> <ul style="list-style-type: none"> – Знайомство з опціями бази даних AWS. – Ключові концепції Amazon DynamoDB. – Розділи та розподіл даних. – Вторинні індекси. – Пропускна здатність читання/запису. – Активність: обчислення RCU та WCU. – Потoki та глобальні таблиці. – Резервне копіювання та відновлення. – Базові операції для таблиць Amazon DynamoDB.
6.	<p><i>Лекція 6. Розробка REST API.</i></p> <ul style="list-style-type: none"> – Знайомство з API. – Знайомство з Amazon API Gateway. – Створення REST API. – Інтеграція з API Gateway. – Розгортання API.

	<ul style="list-style-type: none"> –Контроль доступу до REST API. –Моніторинг REST API. –Оптимізація API Gateway.
7.	<p><i>Лекція 7. Розробка безсерверних рішень на основі подій.</i></p> <ul style="list-style-type: none"> –Знайомство з безсерверними обчисленнями. –Знайомство з AWS Lambda. –Виклик лямбда-функцій. –Налаштування дозволів для лямбда. –Створення та налаштування лямбда-функцій. –Розгортання лямбда-функцій. –Інструменти моніторингу та налагодження для розробників додатків. –Демонстрація: використання AWS X-Ray з Lambda.
8.	<p><i>Лекція 8. Реалізація контейнерів та контейнерних сервісів.</i></p> <ul style="list-style-type: none"> –Знайомство з контейнерами. –Знайомство з контейнерами Docker. –Використання контейнерів для мікросервісів. –Знайомство з контейнерними сервісами AWS. –Розгортання застосунків за допомогою Elastic Beanstalk.
9.	<p><i>Лекція 9. Кешування інформації для забезпечення масштабованості.</i></p> <ul style="list-style-type: none"> –Огляд кешування. –Кешування за допомогою Amazon ElastiCache. –Кешування за допомогою Amazon CloudFront. –Стратегії кешування.
10.	<p><i>Лекція 10. Розробка за допомогою служб обміну повідомленнями.</i></p> <ul style="list-style-type: none"> –Асинхронна обробка запитів. –Знайомство з Amazon SQS. –Робота з повідомленнями Amazon SQS. –Налаштування черг Amazon SQS. –Знайомство з Amazon SNS. –Розробка за допомогою Amazon SNS. –Демонстрація: робота з Amazon Messaging Services. –Знайомство з потоками даних Kinesis.
11.	<p><i>Лекція 11. Консенсус, координація, балансування навантаження та робочі процеси у хмарних обчисленнях.</i></p> <ul style="list-style-type: none"> –Координація завдань у розподілених застосунках. –Знайомство з функціями AWS Step Functions. –Протоколи консенсусу. –Варіанти використання функцій Step Functions. –Балансування навантаження. –Шаблони робочих процесів. –Координація на основі моделі кінцевого автомата, zookeeper.
12.	<p><i>Лекція 12. Розробка безпечних застосунків для хмарної інфраструктури.</i></p> <ul style="list-style-type: none"> –Захист мережевих підключень. –Автентифікація за допомогою AWS STS та Amazon Cognito. –Безпека віртуалізації, віртуальних машин та мобільних пристроїв. –Забезпечення безпеки для користувачів хмарних технологій.
13.	<p><i>Лекція 13. Автоматизація розгортання за допомогою конвеєрів CI/CD.</i></p> <ul style="list-style-type: none"> –Знайомство з DevOps. –Використання AWS Code Services для CI/CD. –Розгортання програм за допомогою AWS CloudFormation. –Розгортання програм за допомогою AWS SAM.

Матеріали для вивчення дисципліни розміщені викладачем у системі Moodle платформи дистанційного навчання «Сікорський». Матеріали доступні з мережі Інтернет зареєстрованому на курс слухачеві. Крім того, матеріали розміщені у хмарному сховищі AWS Academy, посилання на яке слухачі отримують на початку семестру. Навчальний курс у системі Moodle складається з розділів, кожний з яких містить презентації лекцій, завдання та теоретичний матеріал для виконання комп'ютерного практикуму, URL-посилання на документацію з програмного забезпечення та навчальний матеріал для самостійної роботи відповідно до теми, яка вивчається.

Лекції по дисципліні проводяться викладачем із використанням сучасних мультимедійних презентаційних технологій. На лекціях проводяться експрес-опитування, які дають можливість лектору отримати інформацію про якість засвоєння матеріалу та, за необхідності, розглянути більш детально складний матеріал. Обговорення лекційного матеріалу відбувається також в чаті дисципліни в системі Moodle.

Комп'ютерний практикум

Цикл комп'ютерного практикуму дозволяє одержати практичні навички з розроблення, проектування та створення програмного забезпечення для реалізації хмарних послуг.

№ з/п	Назва лабораторної роботи	Кількість ауд. годин
1.	Лабораторна робота 1. Ознайомлення з AWS CloudShell та IDE.	1
2.	Лабораторна робота 2. Робота з Amazon S3.	1
3.	Лабораторна робота 3. Робота з DynamoDB.	1
4.	Лабораторна робота 4. Розробка REST API за допомогою API Gateway.	2
5.	Лабораторна робота 5. Створення лямбда-функцій за допомогою AWS SDK для Python.	2
6.	Лабораторна робота 6. Міграція веб-застосунку до контейнерів Docker.	2
7.	Лабораторна робота 7. Запуск контейнерів на основі Managed Service.	1
8.	Лабораторна робота 8. Кешування даних застосунку за допомогою ElastiCache.	2
9.	Лабораторна робота 9. Реалізація CloudFront для кешування та безпеки застосунку.	2
10.	Лабораторна робота 10. Реалізація системи обміну повідомленнями за допомогою Amazon SNS та Amazon SQS.	1
11.	Лабораторна робота 11. Оркестрація безсерверних функцій за допомогою Step Functions.	1

12.	Лабораторна робота 12. Реалізація автентифікації застосунку за допомогою Amazon Cognito.	1
13.	Лабораторна робота 13. Автоматизація розгортання застосунку за допомогою конвеєра CI/CD.	1

Лабораторні роботи виконуються з використанням спеціалізованих засобів і технологій, що надаються компанією Amazon. Під час їх проведення використовуються методичні вказівки до комп'ютерних практикумів по дисципліні. Для виконання кожного завдання комп'ютерного практикуму слухачеві надається не більше 4 тижнів. Здобувач на початку семестру отримує календарний план початку та завершення захисту завдань комп'ютерного практикуму.

6. Самостійна робота студента/здобувача

№ з/п	Назва теми, що виноситься на самостійне опрацювання	Кількість годин СРС
1.	<i>Екосистема хмарних обчислень.</i>	16
2.	<i>Розподілені обчислення в інфраструктурі хмарних обчислень.</i>	14
3.	<i>Віртуалізація ресурсів в інфраструктурі хмарних обчислень.</i>	16
4.	<i>Моделі зберігання даних, файлові системи, сховища та бази даних.</i>	16
5.	<i>Сучасні хмарні сервіси та оркестрація їх роботи.</i>	10
6.	<i>Безсерверні обчислення та контейнери.</i>	10
7.	<i>Консенсус, координація, балансування навантаження та робочі процеси у хмарних обчисленнях.</i>	14

Матеріали для самостійного вивчення дисципліни розміщені викладачем в електронному вигляді в системі Moodle платформи дистанційного навчання «Сікорський». До самостійної роботи здобувача відноситься, в основному, виконання завдання комп'ютерного практикуму, робота з документацією, а також опрацювання лекційного та додаткового теоретичного матеріалу за наданими презентаціями лекцій та додатковою літературою.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Форми організації освітнього процесу, види навчальних занять і оцінювання результатів навчання регламентуються Положенням про організацію освітнього процесу в КПІ ім. Ігоря Сікорського.

Політика виставлення оцінок: кожна оцінка виставляється відповідно до розроблених викладачем та заздалегідь оголошених здобувачам критеріїв, а також мотивується в індивідуальному порядку на вимогу здобувача; у випадку не виконання здобувачем усіх передбачених навчальним планом видів занять (лабораторних робіт, МКР) до екзамену він не допускається.

Відвідування є обов'язковим (за винятком випадків, коли існує поважна причина). Якщо здобувач не може бути присутнім на заняттях, він все одно несе відповідальність за вивчення матеріалу та виконання лабораторних робіт.

Порядок зарахування пропущених занять. Відпрацювання пропущеного заняття з лекційного курсу здійснюється шляхом підготовки і захисту реферату за відповідною темою у вигляді презентації. Захист реферату відбувається відповідно до графіку консультацій викладача, з яким можна ознайомитись на кафедрі. Відпрацювання пропущеного лабораторного заняття здійснюється шляхом самостійного виконання завдання і його захисту відповідно до графіку консультацій викладача.

Політика академічної поведінки та доброчесності: конфліктні ситуації мають відкрито обговорюватись в академічних групах з викладачем, необхідно бути взаємно толерантним, поважати думку іншого. Плагіат та інші форми нечесної роботи неприпустимі. Всі лабораторні завдання здобувач має виконати самостійно із використанням рекомендованої літератури й отриманих знань та навичок. Цитування в письмових роботах допускається тільки із відповідним посиланням на авторський текст. Недопустимі підказки і списування у ході захисту лабораторних робіт, на модульних контрольних роботах, на екзамені.

Норми академічної етики: дисциплінованість; дотримання субординації; чесність; відповідальність; робота в аудиторії з відключеними мобільними телефонами. Повага один до одного дає можливість ефективніше досягати поставлених командних результатів. При виконанні лабораторних робіт здобувачу потрібен ноутбук, підключений до Інтернет. Якщо ви використовуєте ноутбук чи телефон для аудіо- чи відеозапису, необхідно заздалегідь отримати дозвіл викладача.

Дотримання академічної доброчесності здобувачами та викладачами регламентується кодексом честі КПІ ім. Ігоря Сікорського (https://kpi.ua/files/honorcode_2021.pdf) (<https://osvita.kpi.ua/code>), Положення про систему запобігання академічному плагіату в КПІ ім. Ігоря Сікорського (<https://osvita.kpi.ua/node/47>) та Положенням про організацію освітнього процесу в КПІ ім. Ігоря Сікорського (<https://osvita.kpi.ua/node/39>).

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

Поточний контроль: експрес-опитування, завдання комп'ютерного практикуму (лабораторних робіт).

Календарний контроль: провадиться двічі на семестр як моніторинг поточного стану виконання вимог робочої програми.

Семестровий контроль: екзамен

Семестровий рейтинг здобувача складається з балів, які він отримує за результатами виконання та захисту лабораторних робіт, МКР та екзамену.

Умови допуску до семестрового контролю: виконані усі лабораторні роботи, тести та активності у системі AWS Academy в рамках дисципліни хмарні обчислення до початку екзаменаційної сесії.

Студенти, які не виконали умови допуску отримують академічну заборгованість з дисципліни і мають право її ліквідувати під час канікул.

Усі види діяльності в системі AWS Academy в рамках дисципліни хмарні обчислення повинні бути завершені до початку екзаменаційної сесії.

Під час екзаменаційної сесії виконання лабораторних робіт в системі AWS Academy не проводяться, і студент більше не може отримати бал за виконання лабораторних робіт.

Система рейтингових балів

Практичні навички здобувача оцінюються за результатами захисту виконаних завдань лабораторних робіт. Сумарна оцінка практичних навичок визначається за формулою: $P = \sum D_i$, де D_i – бал за i -ту лабораторну роботу.

Сумарний бал за виконання лабораторних робіт складає **30 балів**. Критерії оцінювання лабораторних робіт включають якість її виконання, захисту та відповідь на запитання.

Екзамен у формі тесту оцінюється 50 балами. МКР складається з результатів тестування після кожного модуля і оцінюється 20 балами. Умови проведення МКР та екзамену визначаються додатково і оголошуються на консультації. МКР та екзамен не переносяться та повторно не проводяться.

Підсумкова оцінка формується за результатами оцінювання знань та навичок студента в семестрі та на екзамені за формулою: $S = P + E + МКР$, де P – сумарна оцінка практичних навичок, E – бал за екзамен.

Для допуску до поточної лабораторної роботи необхідно мати звіт, оформлений відповідно до норм оформлення технічної документації, який має містити всі необхідні пункти, відповідно до методичних вказівок. Також для допуску до лабораторної роботи необхідно захистити попередню роботу. Здобувачі, що не захистили попередню лабораторну роботу, можуть бути не допущені до виконання наступної. Лабораторні роботи виконуються кожним здобувачем індивідуально.

Критерії оцінювання лабораторної роботи (табл. 1):

«відмінно» – повна відповідь на питання (не менше 90% потрібної інформації) та оформлений належним чином протокол до лабораторної роботи;

«добре» – достатньо повна відповідь на питання (не менше 75% потрібної інформації) та оформлений належним чином протокол до лабораторної роботи;

«задовільно» – неповна відповідь на питання (не менше 60% потрібної інформації), незначні помилки та оформлений належним чином протокол до лабораторної роботи;

«незадовільно» – незадовільна відповідь та/або не оформлений належним чином протокол до лабораторної роботи – 0 балів (повертається на відпрацювання або доопрацювання).

Таблиця 1 – Відповідність 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

Для допуску до перескладання екзамену треба у визначений викладачем термін здати всі заборгованості з лабораторних робіт.

Календарний контроль

Календарний контроль (атестація) базується на поточній рейтинговій оцінці. Умовою позитивної атестації є значення поточного рейтингу здобувача не менше 50% від максимального можливого на час атестації. Бал, необхідний для отримання позитивного календарного контролю доводиться до відома здобувачів викладачем не пізніше ніж за 2 тижні до початку календарного контролю.

Здобувачі, які виконали всі умови допуску до екзамену та мають рейтингову оцінку 60 і більше балів, отримують відповідну до набраного рейтингу оцінку без додаткових випробувань.

До відомості семестрового контролю викладач заносить рейтингові бали, отримані здобувачем у семестрі.

Робочу програму навчальної дисципліни (силабус):

Складено: д.т.н., професор Жаріков Е.В.

Ухвалено кафедрою інформатики та програмної інженерії (протокол № 16 від 23.06.2025 р.)

Погоджено методичною комісією факультету (протокол № 11 від 27.06.2025 р.)