

Затверджено

Завідувач кафедри ICT

 Каміла СТОРЧАК

31 серпня 2026 р.

**СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«ТЕОРІЯ ТА ТЕХНОЛОГІЯ СТВОРЕННЯ ПРОГРАМНИХ СИСТЕМ»**

Лектор курсу			Аронов Андрій Олексійович, кандидат технічних наук		Контактна інформація лектора (e-mail), сторінка курсу в Classroom		e-mail: a.aronov@duikt.edu.ua сторінка курсу в Classroom: https://classroom.google.com/c/ODkyNTc4Nzc0MDg4?cjc=qwj4carp код – qwj4carp	
Галузь знань			F Інформаційні технології		Рівень вищої освіти		PhD (доктор філософії)	
Спеціальність			F6 Інформаційні системи і технології		Семестр		2	
Освітня програма			Інформаційні системи та технології		Тип дисципліни		Обов'язкова	
Форма навчання			Денна		Підсумковий контроль		Екзамен	
Обсяг:	Кредитів ECTS	Годин	За видами занять:					
			Лекцій	Семінарських занять	Практичних занять	Лабораторних занять	Самостійна робота	
	4	120	18	-	36	-	66	
АНОТАЦІЯ КУРСУ								
Взаємозв'язок у структурно-логічній схемі								
Освітні компоненти, які передують вивченню			OK02. Методологія наукових досліджень в інформаційних технологіях. OK05. Англійська мова для наукової комунікації.					
Освітні компоненти, для яких є базовою			OK3. Управління науковими дослідженнями.					
Мета курсу:		Формування здатності науково обґрунтовувати, моделювати, проектувати і створювати програмні системи; розробляти сервісно-орієнтоване програмне забезпечення інформаційних систем, виконувати його реінжиніринг та експериментально досліджувати ефективність запропонованих рішень із дотриманням принципів академічної доброчесності.						
Компетентності відповідно до освітньої програми								
Soft- skills / Загальні компетентності (ЗК)					Hard-skills / Спеціальні (фахові) компетентності (СК)			
ЗК01. Здатність до абстрактного мислення, аналізу та синтезу. ЗК04. Здатність розв'язувати комплексні проблеми у сфері інформаційних систем і технологій та з дотичних до міждисциплінарних напрямів на основі системного наукового світогляду та загального культурного кругозору із дотриманням принципів професійної етики та академічної доброчесності.					СК03. Здатність створювати і застосовувати сучасні інформаційні технології, архітектури і спеціалізоване програмне забезпечення у науковій та освітній діяльності, керувати інформаційними ресурсами, інформаційними системами та цифровими сервісами. СК06. Здатність застосовувати сучасні методи дослідження, синтезу, проектування інформаційних систем і технологій у науковій та науково-педагогічній діяльності.			

Програмні результати навчання (ПР)

РН04. Розробляти та досліджувати концептуальні, математичні і комп'ютерні моделі процесів і систем, використовувати їх для отримання нових знань та/або створення інноваційних продуктів у сфері ІСТ та дотичних міждисциплінарних напрямках.

РН05. Планувати і виконувати експериментальні та/або теоретичні дослідження інформаційних систем і технологій з використанням сучасних методів проведення досліджень, а також з використанням технічних і програмних засобів та з дотриманням норм академічної і професійної етики.

РН08. Розробляти програмне забезпечення інформаційних систем у відповідності з принципами сервіс-орієнтованої архітектури розподілених програмних систем, проводити реінжиніринг прикладного інформаційного забезпечення.

ОРГАНІЗАЦІЯ НАВЧАННЯ

Тема, опис теми	Вид заняття	Оцінювання за тему	Форми і методи навчання/питання до самостійної роботи
Наскрізне завдання курсу: створити і дослідити програмний прототип за тематикою дисертації або погодженою науково-прикладною задачею. Результати занять послідовно формують специфікацію, моделі, код, тести та звіт. Внесок кожного здобувача оцінюється індивідуально. Тема 1. Теоретичні засади та наукові підходи до створення програмних систем <i>Рекомендовані джерела: 1, 2, 3.</i> <i>Результати навчання: РН05.</i>			
Заняття 1.1. Програмна система як об'єкт дослідження. Життєвий цикл, парадигми розроблення, складність, доказовий вибір технології. Дослідницькі питання, гіпотези та критерії порівняння рішень.	Лекція 1 2 год		Проведення лекції, візуалізація моделей, критичний аналіз прикладів і наукова дискусія.
Заняття 1.2. Постановка науково-прикладної задачі створення програмної системи; визначення об'єкта, обмежень і вимірюваних критеріїв якості.	Практичне заняття 1 2 год	2 бали	Проблемне навчання, аналіз кейсу. Результат: паспорт задачі та критерії оцінювання.
Заняття 1.3. Порівняння двох підходів до розроблення системи. Обґрунтування базового рішення, гіпотези та плану перевірки запропонованого удосконалення.	Практичне заняття 2 2 год	3 бали	Критичний аналіз публікацій, дискусія. Результат: порівняльна матриця та гіпотеза.
Тема 2. Вимоги, формальні специфікації та моделі поведінки програмних систем <i>Рекомендовані джерела: 1, 5, 6, 8.</i> <i>Результати навчання: РН04.</i>			
Заняття 2.1. Функціональні та нефункціональні вимоги. Передумови, післяумови та інваріанти. Системи переходів, автомати станів; узгодженість і перевірюваність специфікацій.	Лекція 2 2 год		Проведення лекції, візуалізація моделей, критичний аналіз прикладів і наукова дискусія.
Заняття 2.2. Побудова специфікації вимог, моделі предметної області та моделі станів обраної програмної системи.	Практичне заняття 3 2 год	2 бали	Моделювання, взаємне рецензування. Результат: специфікація та UML-моделі.
Заняття 2.3. Формалізація інваріантів і перевірка властивостей моделі засобами Alloy або еквівалентного інструменту; аналіз контрприкладів.	Практичне заняття 4 2 год	3 бали	Комп'ютерний експеримент. Результат: формальна модель і протокол перевірки.

ОРГАНІЗАЦІЯ НАВЧАННЯ

Тема, опис теми	Вид заняття	Оцінювання за тему	Форми і методи навчання/питання до самостійної роботи
Тема 3. Архітектура програмних систем та обґрунтування проєктних рішень <i>Рекомендовані джерела: 1, 2, 5.</i> <i>Результати навчання: РН04.</i>			
Заняття 3.1. Архітектурні стилі, компоненти, інтерфейси та залежності. Атрибути якості, архітектурні компроміси, сценарії та метрики. Документування архітектурних рішень.	Лекція 3 2 год		Проведення лекції, візуалізація моделей, критичний аналіз прикладів і наукова дискусія.
Заняття 3.2. Розроблення двох архітектурних варіантів системи, діаграм компонентів і розгортання; визначення сценаріїв якості.	Практичне заняття 5 2 год	2 бали	Проєктний метод, архітектурне моделювання. Результат: два варіанти архітектури.
Заняття 3.3. Порівняльне оцінювання архітектур за продуктивністю, модифікованістю та надійністю; обґрунтування вибору і меж застосовності.	Практичне заняття 6 2 год	3 бали	Сценарний аналіз, експертне обговорення. Результат: оцінка компромісів та запис рішення.
Тема 4. Сервіс-орієнтовані та розподілені програмні системи <i>Рекомендовані джерела: 1, 2, 7.</i> <i>Результати навчання: РН08.</i>			
Заняття 4.1. Принципи SOA та мікросервісної архітектури. Контракти API, синхронна і асинхронна взаємодія, узгодженість даних, ідемпотентність і відмовостійкість.	Лекція 4 2 год		Проведення лекції, візуалізація моделей, критичний аналіз прикладів і наукова дискусія.
Заняття 4.2. Проєктування сервісних меж і контрактів взаємодії. Розроблення специфікації API та сценаріїв обміну даними.	Практичне заняття 7 2 год	2 бали	Аналіз предметної області, моделювання взаємодій. Результат: контракти сервісів.
Заняття 4.3. Реалізація прототипу щонайменше двох взаємодіючих сервісів; перевірка контрактів та поведінки у разі затримки або відмови.	Практичне заняття 8 2 год	3 бали	Програмний експеримент, демонстрація. Результат: прототип і протокол перевірки взаємодії.
Тема 5. Модельно-орієнтоване розроблення та повторне використання <i>Рекомендовані джерела: 1, 2, 5, 6.</i> <i>Результати навчання: РН04.</i>			
Заняття 5.1. Моделі і метамоделі, рівні абстракції, перетворення моделей у код. Предметно-орієнтовані мови, шаблони, компоненти та варіативність програмних продуктів.	Лекція 5 2 год		Проведення лекції, візуалізація моделей, критичний аналіз прикладів і наукова дискусія.
Заняття 5.2. Побудова моделі компонентів або метамоделі фрагмента системи; визначення правил перетворення моделі на програмні конструкції.	Практичне заняття 9 2 год	2 бали	Дослідницьке моделювання. Результат: модель, обмеження та правила перетворення.
Заняття 5.3. Реалізація простого генератора або повторно використовованого компонента; дослідження узгодженості моделі і коду та трудомісткості змін.	Практичне заняття 10 2 год	3 бали	Прототипування, порівняльний аналіз. Результат: програмний артефакт і оцінка застосовності.

ОРГАНІЗАЦІЯ НАВЧАННЯ

Тема, опис теми	Вид заняття	Оцінювання за тему	Форми і методи навчання/питання до самостійної роботи
Тема 6. Технології конструювання та відтворюване збирання програмних систем <i>Рекомендовані джерела: 1, 2, 6.</i> <i>Результати навчання: РН05.</i>			
Заняття 6.1. Інкрементальне конструювання, керування версіями та залежностями. Автоматизоване збирання, інтеграція, контейнеризація; відтворюваність дослідницького середовища.	Лекція 6 2 год		Проведення лекції, візуалізація моделей, критичний аналіз прикладів і наукова дискусія.
Заняття 6.2. Організація репозиторію прототипу, конфігурації залежностей і сценарію автоматизованого збирання та перевірки.	Практичне заняття 11 2 год	2 бали	Практичне конструювання, аналіз змін. Результат: репозиторій і сценарій збирання.
Заняття 6.3. Створення конвеєра CI та контейнерного опису; повторне розгортання прототипу в чистому середовищі; порівняння результатів.	Практичне заняття 12 2 год	3 бали	Відтворення експерименту. Результат: конфігурація CI, образ і журнал запусків.
Тема 7. Верифікація, валідація та забезпечення якості програмних систем <i>Рекомендовані джерела: 1, 3, 4, 6.</i> <i>Результати навчання: РН05.</i>			
Заняття 7.1. Рівні тестування, тестування властивостей і контрактів, статичний аналіз. Покриття та мутаційне тестування. Надійність вимірювань і обмеження метрик якості.	Лекція 7 2 год		Проведення лекції, візуалізація моделей, критичний аналіз прикладів і наукова дискусія.
Заняття 7.2. Розроблення модульних, інтеграційних та контрактних тестів прототипу; встановлення зв'язку «вимога - перевірка».	Практичне заняття 13 2 год	2 бали	Тестування за специфікацією, взаємне рецензування. Результат: тести і матриця простежуваності.
Заняття 7.3. Порівняння ефективності тестових наборів на навмисно внесених дефектах; аналіз покриття, статичних попереджень і пропущених дефектів.	Практичне заняття 14 2 год	3 бали	Керований експеримент. Результат: звіт про виявлення дефектів та обмеження оцінок.
Тема 8. Еволюція, рефакторинг та реінжиніринг програмних систем <i>Рекомендовані джерела: 1, 2, 4.</i> <i>Результати навчання: РН08.</i>			
Заняття 8.1. Зворотна інженерія, технічний борг, реконструкція архітектури. Рефакторинг, міграція даних та модернізація успадкованих систем. Збереження поведінки і керування ризиками.	Лекція 8 2 год		Проведення лекції, візуалізація моделей, критичний аналіз прикладів і наукова дискусія.
Заняття 8.2. Аналіз успадкованого модуля або початкової версії прототипу: відновлення залежностей, виявлення проблем і підготовка плану реінжинірингу.	Практичне заняття 15 2 год	2 бали	Аналіз коду та архітектури. Результат: модель наявної системи та план змін.
Заняття 8.3. Реінжиніринг обраного фрагмента із виділенням сервісу або зміною структури; регресійна перевірка та порівняння метрик до і після змін.	Практичне заняття 16 2 год	3 бали	Рефакторинг під контролем тестів. Результат: оновлений прототип і докази збереження поведінки.

ОРГАНІЗАЦІЯ НАВЧАННЯ

Тема, опис теми	Вид заняття	Оцінювання за тему	Форми і методи навчання/питання до самостійної роботи
Тема 9. Експериментальне дослідження та наукове обґрунтування програмних рішень <i>Рекомендовані джерела: 1, 2, 3.</i> <i>Результати навчання: PH04, PH05, PH08.</i>			
Заняття 9.1. План експерименту, базовий метод, фактори і метрики. Повторні вимірювання, статистичне оцінювання та загрози валідності. Відтворювані артефакти та аргументація висновків.	Лекція 9 2 год		Проведення лекції, візуалізація моделей, критичний аналіз прикладів і наукова дискусія.
Заняття 9.2. Проведення порівняльного експерименту для базового і удосконаленого прототипів: навантаження, час відгуку, ресурси або надійність.	Практичне заняття 17 2 год	2 бали	Експериментальне дослідження. Результат: план, первинні дані та сценарії запуску.
Заняття 9.3. Статистичне опрацювання результатів і захист дослідницького прототипу: оцінка ефекту, обмеження, відтворюваність та можливий внесок у дисертацію.	Практичне заняття 18 2 год	3 бали	Наукова дискусія, демонстрація, рецензування. Результат: звіт, репозиторій та презентація.
САМОСТІЙНА РОБОТА			
Тема самостійної роботи	Обсяг	Максимум балів	Питання, завдання та форма подання результату
1. Критичний огляд підходів до створення програмної системи	6 год	1 бал	Опрацювати щонайменше 5 наукових джерел за обраною задачею; виділити обмеження наявних підходів. Подати стислий огляд із посиланнями та уточнити гіпотезу.
2. Формальна модель і специфікація вимог	8 год	2 бали	Уточнити вимоги та інваріанти прототипу; опрацювати формальні методи і пояснити виявлений контрприклад. Подати специфікацію та коментар до результатів перевірки.
3. Аналіз архітектурних компромісів	8 год	2 бали	Дослідити щонайменше два архітектурні рішення, сформулювати сценарії якості і обґрунтувати вибір метрик. Подати запис архітектурного рішення з альтернативами та ризиками.
4. Надійна взаємодія розподілених сервісів	8 год	1 бал	Опрацювати ідемпотентність, повторні запити та узгодженість даних. Уточнити контракт API і сценарії оброблення відмов; подати схему взаємодії з поясненням.
5. Трансформація моделей і повторне використання	6 год	2 бали	Дослідити правила генерації коду або компонентного повторного використання. Перевірити узгодженість моделі і реалізації; подати приклад перетворення та аналіз його обмежень.
6. Відтворюване середовище розробки	6 год	1 бал	Уточнити залежності, контейнерний опис і сценарії збирання. Підготувати інструкцію відтворення прототипу та журнал повторного запуску в чистому середовищі.

ОРГАНІЗАЦІЯ НАВЧАННЯ

Тема, опис теми	Вид заняття	Оцінювання за тему	Форми і методи навчання/питання до самостійної роботи
7. Докази якості та валідності перевірок	8 год	2 бали	Опрацювати критерії якості тестів, доповнити набір перевірок та пояснити межі показника покриття. Подати аналіз дефектів і матрицю «вимога - тест - результат».
8. Реінжиніринг та еволюція програмного забезпечення	8 год	2 бали	Обґрунтувати план модернізації, уточнити метрики змін і ризики. Підготувати порівняльний звіт до/після реінжинірингу з результатами регресійного тестування.
9. Узагальнення результатів і підготовка до екзамену	8 год	2 бали	4 год: оформити висновки, загрози валідності та пакет відтворення експерименту. 4 год: систематизувати матеріал тем 1-9 і підготуватися до теоретичних та прикладних завдань екзамену.

МАТЕРІАЛЬНО-ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ ДИСЦИПЛІНИ

Для проведення занять: аудиторія з мультимедійним проектором або дисплеєм; комп'ютерний клас чи персональні комп'ютери з доступом до Інтернету та можливістю локального запуску програмного прототипу. Навчальні матеріали, завдання і зворотний зв'язок - у Google Classroom.

Інструменти обираються відповідно до задачі: Python або інша погоджена з викладачем мова програмування; Git в якості системи керування версіями; PlantUML для UML-моделей (<https://plantuml.com/>); Alloy для перевірки формальних моделей (<https://alloytools.org/documentation.html>).

Для перевірки та відтворення: pytest (<https://docs.pytest.org/en/stable/>); Docker (<https://docs.docker.com/>); GitHub Actions або локальний конвеєр CI (<https://docs.github.com/en/actions>); Apache JMeter для навантажувальних експериментів (<https://jmeter.apache.org/usermanual/index.html>). Допускаються функціонально еквівалентні засоби.

ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ ДИСЦИПЛІНИ

Основні джерела

1. Washizaki H. (ed.). Guide to the Software Engineering Body of Knowledge (SWEBOK Guide). Version 4.0a. IEEE Computer Society, 2025. Оновлений електронний випуск: серпень 2026. <https://ieeecs-media.computer.org/media/education/swebok/swebok-v4.pdf>
2. Цибульник С. О., Барандич К. С. Технології розроблення програмного забезпечення. Частина 1. Життєвий цикл програмного забезпечення: підручник. Київ: КПІ ім. Ігоря Сікорського, 2022. <https://ela.kpi.ua/server/api/core/bitstreams/9521e5f9-421a-4874-a17d-f0853f942856/content>
3. Жебка В. В., Герцюк М. М., Аронов А. О. Методологія та організація проведення наукових досліджень: навч. посіб. Київ: ДУІКТ, 2023. https://duikt.edu.ua/uploads/1_2386_87494381.pdf
4. Wilson G. Software Design by Example: A Tool-Based Introduction with Python. CRC Press, 2024. <https://third-bit.com/sdxdpy/>
5. Felfernig A., Falkner A., Benavides D. Feature Models: AI-Driven Design, Analysis and Applications. Cham: Springer, 2024. <https://link.springer.com/content/pdf/10.1007/978-3-031-61874-1.pdf>
6. Тихоход В. О., Смаковський Д. С., Беспала О. М. Технології розробки програмного забезпечення. Практикум: навч. посіб. Київ: КПІ ім. Ігоря Сікорського, 2024. <https://ela.kpi.ua/server/api/core/bitstreams/280a5ad9-3a93-4412-a4f0-20b682c76049/content>

Додаткові джерела та інформаційні ресурси

7. Компоненти програмної інженерії. Архітектура програмного забезпечення. Лабораторний практикум: навч. посіб. / уклад. Б. Я. Корнієнко. Київ: КПІ ім. Ігоря Сікорського, 2023. <https://ela.kpi.ua/server/api/core/bitstreams/9cfb0e19-fb14-484a-ab13-3063cb375001/content>
8. Cunha A., Macedo N., Brunel J., Chemouil D. Practical Alloy: A Hands-on Guide to Formal Software Design. Електронний навчальний посібник, попередня редакція, оновлена 2025 року. <https://practicalalloy.github.io/>

ПОЛІТИКА КУРСУ («ПРАВИЛА ГРИ»)

Відвідування

Студенти зобов'язані відвідувати заняття згідно з розкладом та вчасно інформувати викладача про неможливість із поважних причин відвідувати заняття, скласти (перескласти) екзамен.

Відпрацювання

Якщо студент із поважних причин був відсутній на практичному занятті, він має право його відпрацювати. Відпрацювання полягає у виконанні студентом в позанавчальний час завдання за темою заняття та захисті роботи викладачеві на наступному практичному занятті (при наявності вільного часу у викладача), або під час додаткової консультації з викладачем, час якої узгоджується окремо. Для виконання завдань курсу може використовуватись обладнання лабораторій кафедр Інформаційних систем та технологій, Інженерії програмного забезпечення або власна комп'ютерна техніка студента.

Академічна доброчесність

Під час вивчення дисципліни всі учасники навчального процесу мають дотримуватися вимог, установлених Положенням про академічну доброчесність здобувачів освіти у ДУІКТ, своєю діяльністю утверджувати академічну доброчесність як засадничу цінність університетської спільноти. Студенти не повинні отримувати чи надавати недозволеної допомоги, вдаватися до жодних інших нечесних дій під час виконання навчальних завдань. Порушення студентом вимог академічної доброчесності під час опанування дисципліни тягне виставлення оцінки 0 балів (неприйнятно).

Поведінка

Спілкуючись з учасниками навчального процесу, студенти мають дотримуватися етичних норм, утримуватися від образливих проявів емоцій, бути політично та гендерно коректними і толерантними, поважати звичаї і традиції різних етнічних, культурних, соціальних груп і релігійних конфесій. Під час лекційних і практичних занять студентам забороняється приймати їжу та користуватися мобільним телефоном. Використання мобільних телефонів допускається при виконанні практичних занять для доступу до навчальних та довідкових матеріалів.

Студенти з особливими потребами

ДУІКТ всіляко сприяє вирішенню індивідуальних проблем студентів з особливими потребами, сприймаючи їх як рівних в інтегрованому студентському колективі. Для покращення сприйняття навчального матеріалу особами з особливими потребами можуть бути використані додаткові програмні засоби, що розширюють можливості інклюзивного доступу (наприклад, програми озвучування тексту). Для осіб з порушеннями зору завдання, що передбачають використання графічних методів роботи з ризиками, можуть бути замінені на завдання з використанням табличних підходів (наприклад, представлення графових структур табличними еквівалентами).

Важливі дати

Подія (чи вид роботи)	Дата
Подання викладачам, що проводять практичні заняття, всіх відпрацьованих завдань	не пізніше, ніж за 10 днів до призначеного екзамену
Подання лектору даних для нарахування додаткових балів (сертифікати про проходження курсів, проекти, наукові публікації тощо).	не пізніше, ніж за 5 днів до призначеного екзамену
Повідомлення про неможливість із поважних причин скласти (перескласти) екзамен	не пізніше, ніж за годину до проведення екзамену

КРИТЕРІЇ ТА МЕТОДИ ОЦІНЮВАННЯ

Підсумковий контроль - екзамен у другому семестрі. Оцінювання за 100-бальною шкалою: практичні роботи - до 45 балів; самостійна робота - до 15 балів; екзамен - до 40 балів. Підсумковий бал = бали практичних робіт + бали самостійної роботи + бали екзамену.

Поточний контроль: перевірка моделей, коду, тестів і звітів; усний захист, демонстрація прототипу, обговорення та відтворення експериментів. Для допуску необхідно виконати передбачені силабусом роботи; допуск оформлюється згідно з Положенням про організацію освітнього процесу. Позитивна підсумкова оцінка - від 60 балів за результатами поточного та екзаменаційного контролю.

Практичні роботи - 45 балів

У кожній із 9 тем перше практичне завдання оцінюється до 2 балів, друге - до 3 балів. Разом за тему - до 5 балів.

Робота на 2 бали: 1 бал - коректність і повнота моделі, специфікації або плану; 1 бал - обґрунтування рішення, пояснення та відповіді на запитання. Робота на 3 бали: 1 бал - коректний програмний або модельний результат; 1 бал - перевірка, аналіз даних і висновки; 1 бал - відтворюваність матеріалів та індивідуальний захист. Кожен критерій: 1 - виконано, 0,5 - виконано частково, 0 - не виконано.

Самостійна робота - 15 балів

Завдання 1, 4, 6 - до 1 бала: по 0,5 бала за змістовну відповідь і належно оформлений підтверджувальний матеріал. Завдання 2, 3, 5, 7, 8, 9 - до 2 балів: 1 бал за повноту та коректність результату, 1 бал за аргументований аналіз, посилання і висновки. За часткове виконання кожного критерію на 1 бал нараховується 0,5 бала. Повторно бали за той самий результат не нараховуються.

Екзамен - 40 балів

Екзаменаційне завдання: два теоретико-аналітичні питання по 10 балів та одне прикладне дослідницьке завдання на 20 балів. Теоретичне питання: коректність понять - 4; аргументація і порівняння підходів - 4; доречний приклад - 2 бали.

Прикладне завдання: постановка задачі та формалізація - 5; архітектурне або програмне рішення - 5; план перевірки та метрики - 5; обґрунтування висновків і обмежень - 5 балів. За кожним критерієм виставляється від 0 до встановленого максимуму відповідно до повноти виконання. Питання охоплюють теми 1-9, насамперед моделі, сервісну взаємодію, реінжиніринг та експериментальну перевірку.

Форма контролю	Вид навчальної роботи	Оцінювання
Поточний контроль	18 практичних робіт	45 балів
	9 завдань самостійної роботи	15 балів
Підсумковий контроль - екзамен	2 теоретико-аналітичні питання + 1 прикладне завдання	40 балів

ПІДСУМКОВА ОЦІНКА ЗА ДИСЦИПЛІНУ

бали	Критерії оцінювання	Рівень компетентності	Оцінка /затис в екзаменаційній відомості
90-100	Студент демонструє повні і міцні знання навчального матеріалу в обсязі, що відповідає робочій програмі дисципліни, правильно і обґрунтовано приймає необхідні рішення в різних нестандартних ситуаціях. Вміє реалізувати теоретичні положення дисципліни в практичних розрахунках, аналізувати та співставляти дані об'єктів діяльності фахівця на основі набутих з даної та суміжних дисциплін знань та умінь. Знає сучасні технології та методи розрахунків з даної дисципліни. За час навчання при проведенні практичних занять, при виконанні індивідуальних / контрольних завдань проявив вміння самостійно вирішувати поставлені завдання, активно включатись в дискусії, може відстоювати власну позицію в питаннях та рішеннях, що розглядаються. Зменшення 100-бальної оцінки може бути пов'язане з недостатнім розкриттям питань, що стосується дисципліни, яка вивчається, але виходить за рамки об'єму матеріалу, передбаченого робочою програмою, або студент проявляє невпевненість в тлумаченні теоретичних положень чи складних практичних завдань.	Високий Повністю забезпечує вимоги до знань, умінь і навичок, що викладені в робочій програмі дисципліни. Власні пропозиції студента в оцінках і вирішенні практичних задач підвищує його вміння використовувати знання, які він отримав при вивченні інших дисциплін, а також знання, набуті при самостійному поглибленому вивченні питань, що відносяться до дисципліни, яка вивчається.	Відмінно / Зараховано (А)
82-89	Студент демонструє гарні знання, добре володіє матеріалом, що відповідає робочій програмі дисципліни, робить на їх основі аналіз можливих ситуацій та вміє застосовувати теоретичні положення при вирішенні практичних задач, але допускає окремі неточності. Вміє самостійно виправляти допущені помилки, кількість яких є незначною. Знає сучасні технології та методи розрахунків з даної дисципліни. За час навчання при проведенні практичних занять, при виконанні індивідуальних / контрольних завдань та поясненні прийнятих рішень, дає вичерпні пояснення.	Достатній Забезпечує студенту самостійне вирішення основних практичних задач в умовах, коли вихідні дані в них змінюються порівняно з прикладами, що розглянуті при вивченні дисципліни	Добре / Зараховано (В)
75-81	Студент в загальному добре володіє матеріалом, знає основні положення матеріалу, що відповідає робочій програмі дисципліни, робить на їх основі аналіз можливих ситуацій та вміє застосовувати при вирішенні типових практичних завдань, але допускає окремі неточності. Вміє пояснити основні положення виконаних завдань та дати правильні відповіді при зміні результату при заданій зміні вихідних параметрів. Помилки у відповідях/ рішеннях/ розрахунках не є системними. Знає характеристики основних положень, що мають визначальне значення при проведенні практичних занять, при виконанні індивідуальних / контрольних завдань та поясненні прийнятих рішень, в межах дисципліни, що вивчається.	Достатній Конкретний рівень, за вивченим матеріалом робочої програми дисципліни. Додаткові питання про можливість використання теоретичних положень для практичного використання викликають утруднення.	Добре / Зараховано (С)
67-74	Студент засвоїв основний теоретичний матеріал, передбачений робочою програмою дисципліни, та розуміє постанову стандартних практичних завдань, має пропозиції щодо напрямку їх вирішень. Розуміє основні положення, що є визначальними в курсі, може вирішувати подібні завдання тим, що розглядалися з викладачем, але допускає значну кількість неточностей і грубих помилок, які може усувати за допомогою викладача.	Середній Забезпечує достатньо надійний рівень відтворення основних положень дисципліни	Задовільно / Зараховано (D)

60-66	Студент має певні знання, передбачені в робочій програмі дисципліни, володіє основними положеннями, що вивчаються на рівні, який визначається як мінімально допустимий. З використанням основних теоретичних положень, студент з труднощами пояснює правила вирішення практичних/розрахункових завдань дисципліни. Виконання практичних / індивідуальних / контрольних завдань значно формалізовано: є відповідність алгоритму, але відсутнє глибоке розуміння роботи та взаємозв'язків з іншими дисциплінами.	Середній Є мінімально допустимим у всіх складових навчальної програми з дисципліни	Задовільно / Зараховано (Е)
35-59	Студент може відтворити окремі фрагменти з курсу. Незважаючи на те, що програму навчальної дисципліни студент виконав, працював він пасивно, його відповіді під час практичних робіт в більшості є невірними, необґрунтованими. Цілісність розуміння матеріалу з дисципліни у студента відсутні.	Низький Не забезпечує практичної реалізації задач, що формуються при вивченні дисципліни	Незадовільно з можливістю повторного складання) / Не зараховано (FX) В екзаменув книжку не представляється
0-34	Студент повністю не виконав вимог робочої програми навчальної дисципліни. Його знання на підсумкових етапах навчання є фрагментарними. Студент не допущений до здачі екзамену/екзамену.	Незадовільний Студент не підготовлений до самостійного вирішення задач, які окреслює мета та завдання дисципліни	Незадовільно з обов'язковим повторним вивченням / Не допущений (F) В екзаменув книжку не представляється