



# КОМПОНЕНТИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ.

## ЧАСТИНА 4. ЯКІСТЬ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

### Робоча програма навчальної дисципліни (Силабус)

#### Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалавр)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>121 Інженерія програмного забезпечення</i>
Освітня програма	<i>Інженерія програмного забезпечення мультимедійних та інформаційно-пошукових систем</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>Очна (денна)</i>
Рік підготовки, семестр	<i>3 рік підготовки, 5 семестр</i>
Обсяг дисципліни	<i>Лекції: 36 год., комп'ютерний практикум: 18 год., самостійна робота: 66 год.</i>
Семестровий контроль/ контрольні заходи	<i>Екзамен, календарний контроль</i>
Розклад занять	<i>Згідно розкладу на осінній семестр поточного навчального року (<a href="http://roz.kpi.ua/">http://roz.kpi.ua/</a>)</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: к.т.н., асистент, Погорелов Володимир Володимирович, volodymyr.pogorelov@gmail.com Комп'ютерний практикум: к.т.н., асистент, Погорелов В.В.</i>
Розміщення курсу	<i>Google classroom. Доступ надається зареєстрованим студентам.</i>

#### Програма навчальної дисципліни

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

*Вивчення дисципліни «Компоненти програмної інженерії. Частина 4. Якість та тестування програмного забезпечення» дозволяє сформувати у здобувачів освіти компетенції, необхідні для розв'язання практичних задач професійної діяльності, пов'язаної із оцінкою якості та тестуванням програмного забезпечення.*

***Метою** вивчення дисципліни «Компоненти програмної інженерії. Частина 4. Якість та тестування програмного забезпечення» є формування у здобувачів освіти здатностей самостійно оцінювати якість та проводити тестування розробленого програмного забезпечення.*

***Предметом** дисципліни «Компоненти програмної інженерії. Частина 4. Якість та тестування програмного забезпечення» є процеси, принципи і техніки, що використовуються для аналізу якості та тестування програмного забезпечення.*

Вивчення дисципліни «Компоненти програмної інженерії. Частина 4. Якість та тестування програмного забезпечення» формує у здобувачів освіти **фахові компетентності (ФК)**, необхідні для розв'язання практичних задач професійної діяльності, пов'язаної з оцінкою якості, тестуванням розробленого програмного забезпечення:

**ФК01** Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.

**ФК02** Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

**ФК03** Здатність розробляти архітектуру, модулі та компоненти програмних систем.

**ФК04** Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

**ФК05** Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.

**ФК07** Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

**ФК08** Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.

**ФК10** Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

**ФК11** Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розроблення програмного забезпечення.

**ФК12** Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.

**ФК13** Здатність обґрунтовано обирати та освоювати інструментарій з розроблення та супроводження програмного забезпечення.

**ФК21** Здатність визначати, аналізувати та документувати вимоги до програмного забезпечення мультимедійних та інформаційно-пошукових систем.

**ФК22** Здатність створювати інноваційні стартап-проекти, обчислювати основні техніко-економічні показники та розробляти бізнес-моделі інноваційних стартап-проектів програмного забезпечення мультимедійних та інформаційно-пошукових систем, які мають комерційний потенціал для інвестування.

**Програмні результати навчання (ПРН)** дисципліни «Компоненти програмної інженерії. Частина 4. Якість та тестування програмного забезпечення» за освітньою програмою:

**ПРН01** Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

**ПРН02** Знати кодекс професійної етики, розуміти соціальну значимість та культурні аспекти інженерії програмного забезпечення і дотримуватись їх в професійній діяльності.

**ПРН03** Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

**ПРН04** Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.

**ПРН06** Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

**ПРН07** Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

**ПРН08** Знати та вміти розробляти людино-машинний інтерфейс.

**ПРН09** Вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.

**ПРН10** Проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування.

**ПРН11** Вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання.

**ПРН13** Знати і застосовувати методи розроблення алгоритмів, конструювання програмного забезпечення та структур даних і знань.

**ПРН14** Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

**ПРН15** Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

**ПРН16** Мати навички програмного розроблення, погодження оформлення і випуску всіх видів програмної документації.

**ПРН17** Вміти застосовувати методи компонентної розробки програмного забезпечення.

**ПРН18** Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.

**ПРН19** Знати та вміти застосовувати методи верифікації та валідації програмного забезпечення.

**ПРН20** Знати підходи щодо оцінювання та забезпечення якості програмного забезпечення.

**ПРН23** Вміти документувати та презентувати результати розроблення програмного забезпечення.

**ПРН31** Вміти визначати, аналізувати та документувати вимоги до програмного забезпечення мультимедійних та інформаційно-пошукових систем.

**ПРН32** Вміти розробляти та аналізувати моделі повного циклу створення програмного забезпечення мультимедійних та інформаційно-пошукових систем.

**ПРН33** Вміти організовувати повний цикл управління програмним продуктом.

**ПРН34** Вміти створювати інноваційні стартап проекти проектування програмного забезпечення мультимедійних та інформаційно-пошукових систем, які мають комерційний потенціал для інвестування.

**ПРН35** Вміти розробляти та аналізувати бізнес-моделі інноваційних стартап проектів розроблення програмного забезпечення мультимедійних та інформаційно-пошукових систем, які мають комерційний потенціал для інвестування.

**ПРН36** Вміти управляти проектами створення та впровадження програмного забезпечення згідно міжнародних стандартів.

**ПРН38** Вміти застосовувати технології програмування для розроблення програмного забезпечення мультимедійних та інформаційно-пошукових систем.

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

Успішному вивченню дисципліни «Компоненти програмної інженерії. Частина 4. Якість та тестування програмного забезпечення» передують вивчення дисциплін «Основи програмування», «Компоненти програмної інженерії. Частина 1. Вступ до програмної інженерії», «Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення», «Компоненти програмної інженерії. Частина 3. Архітектура програмного забезпечення» навчального плану підготовки бакалаврів за спеціальністю 121 Інженерія програмного забезпечення.

Отримані при засвоєнні дисципліни «Компоненти програмної інженерії. Частина 4. Якість та тестування програмного забезпечення» теоретичні знання та практичні уміння забезпечують

успішне проходження переддипломної практики, виконання курсових проєктів та дипломних проєктів за спеціальністю 121 Інженерія програмного забезпечення.

### **3. Зміст навчальної дисципліни**

Дисципліна «Компоненти програмної інженерії. Частина 4. Якість та тестування програмного забезпечення» передбачає вивчення таких тем:

Тема 1. Основи тестування програмного забезпечення

Тема 2. Типи тестування програмного забезпечення

Тема 3. Організація процесу тестування програмного забезпечення

Тема 4. Оцінка якості програмного забезпечення

Тема 5. Засоби оптимізації процесу тестування і контролю якості

Екзамен

### **4. Навчальні матеріали та ресурси**

#### **Базова література:**

1. Авраменко А.С., Авраменко В.С., Косенюк Г.В. Тестування програмного забезпечення. Навчальний посібник. Черкаси: ЧНУ імені Богдана Хмельницького, 2017. 284 с.
2. Скорін Ю. І. Методика кількісної оцінки ефективності роботи інженера із забезпечення якості програмного забезпечення / Ю. І. Скорін, О. В. Щербаков, Є. С. Луценко // Системи обробки інформації : збірник наукових праць. – У 2-х т. Т1. Інформаційні технології та захист інформації. – Харків : ХУПС ім. І. Кожедуба. – 2012. – Вип. 4 (102). – С. 81–84.
3. Системи управління якістю. Основні положення та словник термінів. (ISO 9000:2015, IDT): ДСТУ ISO 9000:2015 – [Чинний від 2016-07-01]. Київ: ДП «УкрНДНЦ», 2016. 49 с. (Національний стандарт України).
4. Системи управління якістю. Вимоги. (ISO 9001:2015, IDT): ДСТУ ISO 9001:2015 – [Чинний від 2016-07-01]. Київ: ДП «УкрНДНЦ», 2016. 30 с. (Національний стандарт України).
5. Табунщик Г.В, Кудерметов Р.К., Брагіна Т.І. Інженерія якості програмного забезпечення [Електронний ресурс]: навчальний посібник. Запоріжжя: ЗНТУ, 2013. 180 с.
6. Якість програмного забезпечення та тестування: базовий курс. Навчальний посібник / За ред. Крепич С.Я., Співак І.Я. / для бакалаврів галузі знань 12 «Інформаційні технології» спеціальності 121 «Інженерія програмного забезпечення». Тернопіль: ФОП Паляниця В.А., 2020. 478 с.

#### **Додаткова література:**

7. *Software Testing and Analysis: Process, Principles and Techniques*, by Mauro Pezze and Michal Young, John Wiley & Sons
8. *The Art of Software Testing, Second Edition* by Glenford J. Myers et. al. Digital copy available in DePaul library.
9. *Software Engineering: A Practitioner's Approach*, Roger S Pressman, McGraw-Hill. Chapters 13 and 14. Використати для опанування практичних умінь дисципліни. Матеріали знаходяться у вільному доступі в Інтернеті.

## Навчальний контент

### 5. Методика опанування навчальної дисципліни (освітнього компонента)

№ з/п	Тип навчального заняття	Опис навчального заняття
<i>Тема 1. Вступ в пошукові системи та сервіси</i>		
1	<i>Лекція 1. Введення в тестування програмного забезпечення</i>	<p><i>Визначення тестування програмного забезпечення. Призначення та цілі тестування ПЗ. Види тестування ПЗ. Життєвий цикл розробки програмного забезпечення та роль тестування на кожному етапі. Важливість тестування програмного забезпечення. Переваги тестування програмного забезпечення.</i></p> <p><i>Завдання на СРС №1.</i></p>
2	<i>Лекція 2. Інспекція в тестуванні. Основні поняття, характеристики інспекції.</i>	<p><i>Визначення Інспекції програмного забезпечення. Переваги Інспекції програмного забезпечення. Процес і методологія Інспекції. Ролі та обов'язки в Інспекції програмного забезпечення. Види Інспекції програмного забезпечення. Інструменти та методи, що використовуються для Інспекції програмного забезпечення</i></p> <p><i>Завдання на СРС №2.</i></p>
3	<i>Лекція 3. Статичний аналіз. Основні поняття та характеристики, засоби виконання статичного аналізу.</i>	<p><i>Визначення статичного аналізу. Переваги статичного аналізу. Типи статичного аналізу (аналіз вихідного коду, аналіз потоку даних тощо). Інструменти та методи, що використовуються в статичному аналізі. Інтеграція статичного аналізу в життєвий цикл розробки програмного забезпечення. Обмеження статичного аналізу.</i></p> <p><i>Завдання на СРС №3.</i></p>
<i>Тема 2. Типи тестування програмного забезпечення</i>		
4	<i>Лекція 4. Модульне тестування. Завдання, вхідні дані, основні кроки тестування, умови завершення модульного тестування.</i>	<p><i>Визначення модульного тестування. Переваги модульного тестування. Процес і методологія модульного тестування. Написання ефективних модульних тестів. Розробка, керована тестуванням (TDD). Інструменти та методи, що використовуються в модульному тестуванні.</i></p> <p><i>Завдання на СРС №4.</i></p>
5	<i>Лекція 5. Інтеграційне тестування та тестування системи.</i>	<p><i>Визначення інтеграційного та системного тестування. Переваги інтеграційного та системного тестування. Процес і методологія інтеграційного тестування. Процес і методологія системного тестування. Типи інтеграційного та системного тестування.</i></p>



		<p>Інструменти та методи, що використовуються в інтеграційному та системному тестуванні.</p> <p>Завдання на СРС №5.</p>
6	<p>Лекція 6. Регресійне тестування, основні поняття та характеристики, ключові етапи.</p>	<p>Визначення регресійного тестування. Переваги регресійного тестування. Процес і методологія регресійного тестування. Типи регресійного тестування (ручне та автоматичне регресійне тестування). Стратегії ефективного регресійного тестування. Інструменти та методика, що використовуються в регресійному тестуванні.</p> <p>Завдання на СРС №6.</p>
7	<p>Лекція 7. Функціональне тестування.</p>	<p>Визначення функціонального тестування. Переваги функціонального тестування. Процес і методологія функціонального тестування. Типи функціонального тестування (тестування чорного ящика, тестування білого ящика тощо). Написання ефективних функціональних тестів. Інструменти та методика, що використовуються у функціональному тестуванні.</p> <p>Завдання на СРС №7.</p>
8	<p>Лекція 8. Структурне тестування</p>	<p>Визначення структурного тестування. Переваги структурного тестування. Процес і методологія структурного тестування. Види структурних випробувань (тестування гілок, тестування стану тощо). Написання ефективних структурних тестів. Інструменти та методи, що використовуються в структурному тестуванні.</p> <p>Завдання на СРС №8.</p>
9	<p>Лекція 9. Тестування об'єктно-орієнтованого програмного забезпечення</p>	<p>Визначення об'єктно-орієнтованого програмного забезпечення. Стратегії тестування об'єктно-орієнтованого програмного забезпечення (наслідування, поліморфізм, інкапсуляція). Написання ефективних тестів для об'єктно-орієнтованого програмного забезпечення. Інструменти та методи, що використовуються при тестуванні об'єктно-орієнтованого програмного забезпечення.</p> <p>Завдання на СРС №9.</p>
10	<p>Комп'ютерний практикум 1. Тестування об'єктно-орієнтованого програмного забезпечення</p>	<p>Завдання: За допомогою програмних засобів реалізувати модуль для реалізації тестувань, типи яких розглянуті в Темі 2.</p> <p>Завдання на СРС №10.</p>
<p>Тема 3. Організація процесу тестування програмного забезпечення</p>		

11	<i>Лекція 10. Вибір тестового випадку</i>	<i>Визначення вибору тестового випадку. Процес і методологія відбору тестових випадків. Фактори, які слід враховувати під час вибору тестових випадків (покриття, ризик тощо). Типи методів вибору тестових випадків (розподіл еквівалентності, аналіз граничних значень тощо). Інструменти та методи, що використовуються для вибору тестових випадків. Завдання на СРС №11.</i>
12	<i>Лекція 11. Тестування продуктивності</i>	<i>Визначення тестування продуктивності. Переваги тестування продуктивності. Процес і методологія тестування продуктивності. Типи тестування продуктивності (тестування навантаження, стрес-тестування тощо). Критерії тестування продуктивності (час відгуку, пропускна здатність тощо). Інструменти та методи, що використовуються в тестуванні продуктивності Завдання на СРС №12.</i>
13	<i>Лекція 12. Тестування безпеки</i>	<i>Визначення тестування безпеки. Переваги тестування безпеки. Процес і методологія тестування безпеки. Види тестування безпеки (тестування на проникнення, сканування вразливостей тощо). Критерії перевірки безпеки (конфіденційність, цілісність, доступність). Інструменти та методи, що використовуються в тестуванні безпеки. Завдання на СРС №13.</i>
14	<i>Лекція 13. Тестування веб-додатків</i>	<i>Визначення тестування веб-додатків. Переваги тестування веб-додатків. Процес і методологія тестування веб-додатків. Написання ефективних тестів для веб-додатків. Інструменти та методи, що використовуються для тестування веб-додатків. Завдання на СРС №14.</i>
15	<i>Комп'ютерний практикум 2. Організація процесу тестування програмного забезпечення</i>	<i>Завдання: За допомогою програмних засобів реалізувати модуль для реалізації тестувань, типи яких розглянуті в Темі 3. Завдання на СРС №15.</i>
<i>Тема 4. Оцінка якості програмного забезпечення</i>		
16	<i>Лекція 14. Тестування графічного інтерфейсу користувача (GUI)</i>	<i>Визначення GUI тестування. Процес і методологія тестування GUI. Критерії тестування GUI (зручність використання, доступність, послідовність тощо). Інструменти та методи,</i>

		що використовуються для тестування GUI. Завдання на СРС №16.
17	Лекція 15. Тестування зручності використання	Визначення, переваги, процес і методологія тестування зручності використання. Види (експертна рецензія, евристична оцінка тощо). Критерії тестування зручності використання. Інструменти та методи, що використовуються для тестування зручності використання.  Завдання на СРС №17.
18	Лекція 16. Тестування на основі помилок	Визначення тестування на основі помилок. Переваги тестування на основі помилок. Процес і методологія тестування на основі помилок. Типи тестування на основі помилок (ін'єкція помилок, перевірка мутацій тощо). Критерії тестування на основі несправностей (швидкість виявлення, частота помилкових тривог тощо). Інструменти та методи, що використовуються в тестуванні на основі помилок.  Завдання на СРС №18.
<i>Тема 5. Засоби оптимізації процесу тестування і контролю якості</i>		
19	Лекція 17. Автоматизація та інструменти тестування	Визначення автоматизації тестування та інструментів. Переваги автоматизації тестування. Процес і методологія автоматизації тестування. Види автоматизації тестування. Критерії вибору засобів автоматизації тестування (охоплення, надійність тощо). Інструменти та методи, що використовуються для автоматизації тестування.  Завдання на СРС №19.
20	Комп'ютерний практикум 3. Використання JUnit для тестування ПЗ.	Завдання: За допомогою програмних засобів реалізувати модуль, що виконує тестування ПЗ.  Завдання на СРС №20.
21	Лекція 18. Планування та моніторинг процесу якості програмного забезпечення	Визначення процесу планування та моніторингу якості програмного забезпечення. Процес і методологія планування та моніторингу. Типи показників якості ПЗ (надійність, ефективність тощо). Методи покращення процесу якості програмного забезпечення (Six Sigma, Total Quality Management тощо). Інструменти та методи, що використовуються для планування та моніторингу процесу якості програмного забезпечення.  Завдання на СРС № 21.



Модульна контрольна робота

**6. Самостійна робота студента/аспіранта**

Дисципліна «Інформаційно-пошукові системи та сервіси» ґрунтується на самостійних підготовках до аудиторних занять на теоретичні та практичні теми.

№ з/п	Назва теми, що виноситься на самостійне опрацювання	Кількість годин	Література
1	Підготовка до лекції 1	1	1, 2
2	Підготовка до лекції 2	1	1, 8
3	Підготовка до лекції 3	1	1, 7
4	Підготовка до лекції 4	1	1, 8
5	Підготовка до лекції 5	1	1, 7, 8
6	Підготовка до лекції 6	1	1, 2
7	Підготовка до комп'ютерного практикуму 1	3	1, 8
8	Підготовка до лекції 7	1	1, 8
9	Підготовка до лекції 8	1	1, 7
10	Підготовка до лекції 9	1	1, 8, 9
11	Підготовка до комп'ютерного практикуму 2	3	1, 2, 7
12	Підготовка до лекції 10	1	1, 9
13	Підготовка до лекції 11	1	1, 8
14	Підготовка до лекції 12	1	1, 7, 8
15	Підготовка до лекції 13	1	1, 2
16	Підготовка до лекції 14	1	1, 3, 4
17	Підготовка до лекції 15	1	1, 2, 4
18	Підготовка до лекції 16	1	1, 3, 4
19	Підготовка до лекції 17	1	1, 8
20	Підготовка до комп'ютерного практикуму 3	1	1, 8, 9
21	Підготовка до лекції 18		
22	Підготовка до модульної контрольної роботи	9	1, 2, 7
23	Підготовка до екзамену	36	1, 2

**Політика та контроль**

**7. Політика навчальної дисципліни (освітнього компонента)**

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

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

Протягом семестру студенти виконують 3 комп'ютерних практикуми. **Максимальна кількість балів** за кожний комп'ютерний практикум: 10 балів.

Бали нараховуються за:

- якість виконання комп'ютерного практикуму: 0-6 бали;
- відповідь під час захисту комп'ютерного практикуму: 0-2 бали;
- своєчасне представлення роботи до захисту: 0-2 бали.

Критерії оцінювання якості виконання:

- 6 балів – робота виконана якісно, в повному обсязі;
- 3-5 бали – робота виконана якісно, в повному обсязі, але має недоліки;
- 1-2 балів – робота виконана в повному обсязі, але містить суттєві помилки;
- 0 балів – робота виконана не в повному обсязі.

Критерії оцінювання відповіді:

- 2 бали – відповідь повна, добре аргументована;
- 1 бали – у відповіді є суттєві помилки;
- 0 балів – немає відповіді або відповідь невірна.

Критерії оцінювання своєчасності представлення роботи до захисту:

- 2 бали – робота представлена до захисту не пізніше вказаного терміну;
- 0 балів – робота представлена до захисту пізніше вказаного терміну.

**Максимальна кількість балів за виконання та захист комп'ютерних практикумів:**

10 балів × 3 комп. практ. = 30 балів.

Протягом семестру студенти виконують контрольну роботу. Завдання складається з 3 теоретичних та 2 практичних запитань. Відповідь на кожне запитання оцінюється 4 балами.

Критерії оцінювання кожного запитання контрольної роботи:

- 4 бали – відповідь вірна, повна, добре аргументована;
- 2-3 балів – в цілому відповідь вірна, але має недоліки;
- 1 бали – у відповіді є суттєві помилки;
- 0 балів – немає відповіді або відповідь невірна.

**Максимальна кількість балів за модульну контрольну роботу:**

4 бали × 5 запитань = 20 балів.

**Рейтингова шкала з дисципліни дорівнює:**

$R = R_C = R_{\text{ком.практ}} + R_{\text{МКР}} + R_{\text{екзамен}} = 30 \text{ балів} + 20 \text{ балів} + 50 \text{ балів} = 100 \text{ балів.}$

Календарний контроль: проводиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.

На першій атестації (8-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 10 балів (50 % від максимальної кількості балів, яку може отримати студент до першої атестації).

На другій атестації (14-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 20 балів (50 % від максимальної кількості балів, яку може отримати студент до другої атестації).

Семестровий контроль: екзамен

Умови допуску до семестрового контролю:

При семестровому рейтингу ( $R_C$ ) не менше 30 балів та зарахуванні усіх робіт комп'ютерного практикуму, студент має допуск до екзамену. Після складання екзамену виставляється оцінка відповідно до таблиці (Таблиця відповідності рейтингових балів оцінкам за університетською шкалою).

*Необхідною умовою допуску до екзамену є виконання та захист комп'ютерного практикуму.*

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

<i>Кількість балів</i>	<i>Оцінка</i>
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

#### **9. Додаткова інформація з дисципліни (освітнього компонента)**

*Перелік питань, які виносяться на семестровий контроль, наведено у Додатку 1.*

#### **Робочу програму навчальної дисципліни (силабус):**

**Складено** к.т.н., асистент, Погорелов В.В.; аспірант, Ільїн М.О.

**Ухвалено** кафедрою ПЗКС (протокол № 12 від 26.04.2023 р.)

**Погоджено** Методичною комісією факультету прикладної математики (протокол № 10 від 26.05.2023 р.)

*Додаток 1. Перелік питань, які виносяться на семестровий контроль*

- 1. Що таке тестування програмного забезпечення і чому це важливий етап розробки ПЗ?*
- 2. У чому різниця між тестуванням Інспекцією та статичним тестуванням?*
- 3. Що таке модульне тестування і чим воно відрізняється від інтеграційного та системного тестування?*
- 4. Що таке регресійне тестування і навіщо воно проводиться?*
- 5. Що таке функціональне тестування і як воно виконується?*
- 6. Що таке структурне тестування і які методи використовуються?*
- 7. Вибір тестових випадків для програмної системи.*
- 8. Чим тестування об'єктно-орієнтованого програмного забезпечення відрізняється від програмного забезпечення інших типів?*
- 9. Що таке тестування продуктивності і як воно виконується?*
- 10. Що таке тестування безпеки та які методи використовуються?*
- 11. Що таке тестування веб-додатків і які методи використовуються?*
- 12. Що таке тестування графічного інтерфейсу користувача (GUI) і які методи використовуються?*
- 13. Що таке юзабіліті-тестування і які методи використовуються?*
- 14. Що таке тестування на основі помилок і які методи використовуються?*
- 15. Що таке автоматизація тестування та які інструменти використовуються?*
- 16. Яка роль планування та моніторингу в процесі якості програмного забезпечення?*
- 17. Показники якості програмного забезпечення.*
- 18. Яка різниця між модульним тестуванням та інтеграційним тестуванням?*
- 19. Яка мета регресійного тестування і як воно виконується?*
- 20. Яка різниця між тестуванням безпеки та тестуванням продуктивності?*