

Федеральное государственное образовательное учреждение
высшего образования
**«КАЛИНИНГРАДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»**

А. М. Лутовинова

**ТЕСТИРОВАНИЕ И АНАЛИЗ КАЧЕСТВА ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ**

Учебно-методическое пособие по изучению дисциплины
для студентов бакалавриата по направлению подготовки
09.03.03 Прикладная информатика

Калининград
Издательство ФГБОУ ВО «КГТУ»
2025

УДК 004.9(075)

Рецензент

кандидат педагогических наук,

доцент кафедры прикладной информатики института цифровых технологий
ФГБОУ ВО «Калининградский государственный технический университет»

Е. Ю. Заболотнова

Лутовинова, А.М

Тестирование и анализ качества программного обеспечения: учеб.-метод. пособие по изучению дисциплины для студентов бакалавриата по направлению подготовки 09.03.03 Прикладная информатика / А. М. Лутовинова. – Калининград: Изд-во ФГБОУ ВО «КГТУ», 2024. – 24 с.

В учебно-методическом пособии приведен тематический план по дисциплине и даны методические указания по её самостоятельному изучению, подготовке к лекционным и лабораторным занятиям, подготовке и сдаче экзамена, выполнению самостоятельной работы.

Пособие подготовлено в соответствии с требованиями утвержденной рабочей программы дисциплины направления подготовки 09.03.03 Прикладная информатика.

Табл. 5, список лит. – 14 наименований

Учебно-методическое пособие рассмотрено и одобрено в качестве локального электронного методического материала кафедрой прикладной информатики 16 сентября 2025 г., протокол № 2

Учебно-методическое пособие по изучению дисциплины рекомендовано к использованию в качестве локального электронного методического материала в учебном процессе методической комиссией института цифровых технологий ФГБОУ ВО «Калининградский государственный технический университет» 21 октября 2025 г., протокол № 7

УДК 004.9(075)

© Федеральное государственное
бюджетное образовательное учреждение
высшего образования
«Калининградский государственный
технический университет», 2025 г.
© Лутовинова А. М., 2025 г.

ОГЛАВЛЕНИЕ

1 ВВЕДЕНИЕ.....	4
2 ТЕМАТИЧЕСКИЙ ПЛАН.....	6
3 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ	7
4 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРОВЕДЕНИЮ ЛАБОРАТОРНЫХ ЗАНЯТИЙ.....	11
5 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ РАСЧЕТНО- ГРАФИЧЕСКОЙ РАБОТЫ.....	12
5.1 Цель и задачи РГР	13
5.2 Тематика РГР	13
5.3 Структура РГР	13
5.4 Порядок выполнения и защиты РГР.....	14
6 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ	15
7 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРОВЕДЕНИЮ ЗАНЯТИЙ И ОСВОЕНИЮ ДИСЦИПЛИНЫ	16
8 ТРЕБОВАНИЯ К АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ	17
8.1 Текущая аттестация.....	17
8.2 Промежуточная аттестация по дисциплине	18
9 ЗАКЛЮЧЕНИЕ	20
10 БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	21

1 ВВЕДЕНИЕ

Данное учебно-методическое пособие предназначено для студентов направления подготовки 09.03.03 Прикладная информатика, изучающих дисциплину «Тестирование и анализ качества ПО».

Целью освоения дисциплины является формирование знаний о способах, подходах, рисках и оценке экономического эффекта от внедрения информационных систем (ИС) на предприятиях.

Задача изучения дисциплины – приобретение студентом способности проводить тестирование и анализ качества разработанного программного обеспечения и документировать его результаты.

В результате изучения дисциплины студент должен:

знать:

- этапы разработки программного обеспечения;
- принципы технологий структурного и объектно-ориентированного программирования;
- способы оптимизации и приемы рефакторинга;
- принципы отладки и тестирования программных продуктов;
- основные цели и задачи управления качеством продукции;
- общие принципы построения систем управления качеством;
- критерии эффективности систем управления качеством;

уметь:

- осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
- создавать программу по разработанному алгоритму как отдельный модуль;
- выполнять отладку и тестирование программы на уровне модуля;
- анализировать состав подсистем управления качеством и осуществлять их формирование;
- применять модели обеспечения качества;
- анализировать общие вопросы оценки эффективности систем управления качеством;

владеть:

- навыками в разработке кода программного продукта на основе готовой спецификации на уровне модуля;

- навыками использования инструментальных средств на этапе отладки программного продукта;
- навыками проведения тестирования программного модуля по определенному сценарию;
- навыками использования основных инструментов качества;
- навыками применения статистических методов контроля качества;
- навыками использования общих подходов к описанию моделей обеспечения качества.

Дисциплина «Тестирование и анализ качества ПО» входит в состав профессионального модуля части, формируемой участниками образовательных отношений, образовательной программы бакалавриата по направлению подготовки 09.03.03 Прикладная информатика.

Для успешного освоения дисциплины, в соответствии с учебным планом, ей предшествуют такие дисциплины, как: «Программирование на языках высокого уровня», «Технологии программирования», «Базы данных», «Системный анализ».

Результаты освоения дисциплины могут быть использованы при выполнении выпускной квалификационной работы, а также в дальнейшей профессиональной деятельности.

Далее в пособии представлен тематический план, содержащий перечень изучаемых тем и лабораторных занятий, мероприятий текущей аттестации и отводимое на них аудиторное время (занятия в соответствии с расписанием) и самостоятельную работу. При формировании личного образовательного плана на семестр студентам следует оценивать рекомендуемое время на изучение дисциплины, возможно, им потребует больше времени на выполнение отдельных заданий или проработку отдельных тем.

В разделе «Содержание дисциплины» приведены подробные сведения об изучаемых вопросах, по которым вы можете ориентироваться в случае пропуска каких-то занятий, а также методические рекомендации преподавателя для самостоятельной подготовки, каждая тема имеет ссылки на литературу (или иные информационные ресурсы), а также контрольные вопросы для самопроверки.

Раздел «Требования к аттестации по дисциплине» содержит описание обязательных мероприятий контроля самостоятельной работы и усвоения разделов или отдельных тем дисциплины. Далее изложены требования к завершающей аттестации – экзамену.

Помимо данного пособия, студентам следует использовать материалы, размещенные в соответствующем данной дисциплине разделе ЭИОС, в которые более оперативно вносятся изменения для адаптации дисциплины под конкретную группу.

2 ТЕМАТИЧЕСКИЙ ПЛАН

Общая трудоемкость дисциплины составляет 6 зачетных единиц (ЗЕТ), т. е. 216 академических часов контактной (лекционных и лабораторных занятий) и самостоятельной работы студента, в том числе связанной с текущей и промежуточной (заключительной) аттестацией по дисциплине.

Формы аттестации по дисциплине:

- очная форма, восьмой семестр – экзамен + расчетно-графическая работа (РГР);
- заочная форма, восьмой семестр – экзамен + расчетно-графическая работа (РГР).

Тематический план лекционных занятий для очной и заочной формы обучения приведен в таблице 1.

Таблица 1 – Тематический план лекционных занятий

Номер микромодуля	Тема лекционного занятия	Объем учебной работы	
		очная форма, ч	заочная форма, ч
Микромодуль 1	Введение в тестирование. Оценка оттестированности проекта	4	1
	Принципы тестирования ПО	4	1
Микромодуль 2	Проектирование тестов	6	2
	Критерии «черного» ящика	6	1
	Критерии «белого» ящика	6	1
	Модульное и интеграционное тестирование	8	1
	Рандомизированное тестирование	4	1
	Регрессионное тестирование	4	1
Микромодуль 3	Автоматизация построения тестов	6	1
Итого:		48	10

3 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Содержательно структура дисциплины представлена тремя тематическими блоками (микромодулями).

Микромодуль 1. Введение в тестирование ПО

Тема 1.1.1 Введение в тестирование. Оценка оттестированности проекта

Перечень изучаемых вопросов:

Цели и задачи дисциплины. Место дисциплины в структуре образовательной программы. Планируемые результаты освоения дисциплины.

Основные понятия и определения в области тестирования программного обеспечения.

Различие между тестированием, отладкой и верификацией. Жизненный цикл разработки программного обеспечения и роль тестирования в нем. Принципы тестирования: исчерпывающее тестирование невозможно, раннее тестирование, кластеризация дефектов, парадокс пестицида, тестирование зависит от контекста, заблуждение об отсутствии ошибок.

Оценка оттестированности проекта: метрики покрытия кода, функциональное покрытие, критерии завершения тестирования. Количественные и качественные показатели тестирования.

Рекомендуемая литература: [1, гл. 1]; [2, гл. 1, 2]; [3, гл. 1].

Контрольные вопросы

1. Что такое тестирование программного обеспечения и каковы его основные цели?
2. Чем отличается тестирование от отладки программного обеспечения?
3. Назовите основные принципы тестирования и объясните их значение.
4. Какие существуют метрики для оценки качества тестирования?
5. Как определить критерии завершения тестирования проекта?

Тема 1.1.2. Принципы тестирования ПО

Перечень изучаемых вопросов

Семь принципов тестирования по ISTQB. Психология тестирования. Роли и ответственность в команде тестирования. Процесс тестирования: планирование, анализ, проектирование, реализация, выполнение, завершение.

Документирование процесса тестирования. Стратегии тестирования. Риск-ориентированное тестирование. Управление дефектами: обнаружение, документирование, отслеживание, закрытие.

Рекомендуемая литература: [1, гл. 2]; [2, гл. 3]; [4, гл. 1].

Контрольные вопросы

1. Перечислите и объясните семь принципов тестирования по ISTQB.

2. Какие роли существуют в команде тестирования?
3. Опишите основные этапы процесса тестирования.
4. Что такое риск-ориентированное тестирование?
5. Какие состояния может иметь дефект в процессе его жизненного цикла?

Микромодуль 2. Методы и техники тестирования

Тема 2.1.1 Проектирование тестов

Перечень изучаемых вопросов

Процесс проектирования тестов. Анализ требований к программному обеспечению.

Создание тест-планов и тест-дизайна. Структура тест-кейса: предусловия, шаги выполнения, ожидаемый результат. Трассируемость требований и тестов.

Техники тест-дизайна: статические и динамические методы. Принципы создания эффективных тестовых сценариев. Приоритизация тестов. Управление тестовыми данными. Оптимизация тестового покрытия.

Рекомендуемая литература: [1, гл. 3]; [2, гл. 4]; [3, гл. 3].

Контрольные вопросы

1. Что включает в себя процесс проектирования тестов?
2. Какова структура типичного тест-кейса?
3. Что такое трассируемость требований и почему она важна?
4. Какие принципы следует соблюдать при создании тестовых сценариев?
5. Как осуществляется приоритизация тестов?

Тема 2.1.2. Критерии «черного» ящика: эквивалентное разбиение, граничные значения, функциональные диаграммы

Перечень изучаемых вопросов

Тестирование методом «черного ящика». Техника эквивалентного разбиения: принципы выделения классов эквивалентности, выбор представителей классов. Анализ граничных значений: внутренние и внешние границы, типичные ошибки в граничных условиях.

Функциональные диаграммы и их использование для создания тестов. Предположение об ошибке (error guessing): использование опыта для выявления потенциальных проблем. Исследовательское тестирование. Тестирование таблиц решений. Тестирование переходов состояний.

Рекомендуемая литература: [1, гл. 4]; [2, гл. 5]; [4, гл. 2].

Контрольные вопросы

1. В чем заключается принцип эквивалентного разбиения?
2. Как правильно выбирать граничные значения для тестирования?
3. Что такое предположение об ошибке и когда его следует применять?
4. Какие преимущества и недостатки имеет тестирование «черного ящика»?
5. Как применять таблицы решений в тестировании?

Тема 2.1.3. Критерии «белого» ящика: критерии потока управления

Перечень изучаемых вопросов

Тестирование методом «белого ящика». Структурное тестирование и его виды.

Критерии покрытия потока управления: покрытие операторов (statement coverage), покрытие ветвей (branch coverage), покрытие условий (condition coverage), покрытие путей (path coverage).

Цикломатическая сложность и её использование для определения количества тестов.

Инструменты для измерения покрытия кода. Практические аспекты структурного тестирования. Тестирование потоков данных.

Рекомендуемая литература: [1, гл. 5]; [2, гл. 6]; [4, гл. 3].

Контрольные вопросы

1. Какие виды покрытия кода существуют при структурном тестировании?
2. Что такое цикломатическая сложность и как она связана с тестированием?
3. В чем разница между покрытием ветвей и покрытием условий?
4. Какие инструменты можно использовать для измерения покрытия кода?
5. Что такое тестирование потоков данных?

Тема 2.2.1. Модульное и интеграционное тестирование. Интеграционное тестирование для ООП

Перечень изучаемых вопросов:

Модульное (unit) тестирование: цели, принципы, инструменты. Создание модульных тестов. Заглушки (stubs) и драйверы (drivers). Фреймворки для модульного тестирования. Изоляция модулей при тестировании.

Интеграционное тестирование: стратегии интеграции (восходящая, нисходящая, сэндвич). Особенности интеграционного тестирования объектно-ориентированных систем: тестирование классов, наследования, полиморфизма. Тестирование взаимодействия компонентов. Системное тестирование.

Рекомендуемая литература: [1, гл. 6]; [2, гл. 7]; [5, гл. 4].

Контрольные вопросы

1. В чем различие между модульным и интеграционным тестированием?
2. Какие стратегии интеграционного тестирования существуют?
3. Какие особенности имеет тестирование объектно-ориентированных систем?
4. Что такое заглушки и драйверы в контексте тестирования?
5. Как тестировать наследование и полиморфизм в ООП?

Тема 2.2.2. Рандомизированное тестирование

Перечень изучаемых вопросов

Принципы рандомизированного тестирования. Генерация случайных тестовых данных. Статистические методы в тестировании. Фаззинг (fuzzing) как метод тестирования безопасности и надежности.

Преимущества и ограничения рандомизированного тестирования. Комбинирование с детерминированными методами. Инструменты для автоматической генерации тестовых данных. Monkey testing и его применение.

Рекомендуемая литература: [2, гл. 8]; [4, гл. 5]; [6, гл. 3].

Контрольные вопросы

1. В каких случаях эффективно применение рандомизированного тестирования?
2. Что такое фаззинг и для чего он используется?
3. Какие преимущества и недостатки имеет генерация случайных тестовых данных?
4. Как можно комбинировать рандомизированное и детерминированное тестирование?
5. Что такое Monkey testing?

Тема 2.2.3. Регрессионное тестирование

Перечень изучаемых вопросов:

Понятие регрессионного тестирования. Причины проведения регрессионных тестов: изменения в коде, исправление дефектов, добавление новой функциональности. Стратегии регрессионного тестирования: повторное выполнение всех тестов, селективное повторное тестирование, приоритизация тестов.

Инструменты автоматизации регрессионного тестирования. Управление тест-сьютами. Анализ влияния изменений на систему тестирования. Дымовое тестирование (smoke testing).

Рекомендуемая литература: [1, гл. 7]; [2, гл. 9]; [5, гл. 6].

Контрольные вопросы

1. Что такое регрессионное тестирование и когда оно проводится?

2. Какие стратегии регрессионного тестирования существуют?
3. Как определить, какие тесты следует включить в регрессионный набор?
4. Какую роль играет автоматизация в регрессионном тестировании?
5. Что такое дымовое тестирование?

Микромодуль 3. Автоматизация и управление качеством

Тема 3.1.1 Автоматизация построения тестов

Перечень изучаемых вопросов

Принципы автоматизации тестирования. Когда следует автоматизировать тестирование, а когда нет. Пирамида тестирования: модульные, интеграционные и UI тесты. ROI автоматизации тестирования.

Инструменты автоматизации тестирования: Selenium WebDriver, Appium, TestComplete, Jest, JUnit. Создание автоматизированных тестов: запись и воспроизведение, программирование тестов. Паттерны проектирования в автоматизации тестирования: Page Object Model, Page Factory.

Continuous Integration и Continuous Testing. Интеграция автоматизированных тестов в процесс разработки. DevOps и тестирование. Управление качеством в Agile-разработке.

Рекомендуемая литература: [3, гл. 8]; [5, гл. 7]; [7, гл. 4].

Контрольные вопросы

1. Какие принципы следует соблюдать при автоматизации тестирования?
2. Что такое пирамида тестирования и как она помогает в планировании автоматизации?
3. Какие паттерны проектирования применяются в автоматизированном тестировании?
4. Как интегрировать автоматизированные тесты в процесс непрерывной интеграции?
5. Как рассчитать ROI автоматизации тестирования?

4 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРОВЕДЕНИЮ ЛАБОРАТОРНЫХ ЗАНЯТИЙ

Особое место в структуре дисциплины занимает практикум, включающий в себя восемь лабораторных работ (таблица 2).

Таблица 2– Объем (трудоёмкость освоения) и структура ЛЗ

Номер темы	Содержание лабораторного занятия	Очная форма, ч	Заочная форма, ч
1.1	Оценка оттестированности проекта	4	1
1.1	Применение принципов тестирования	4	1
2.1	Разработка тестовых сценариев и тест-кейсов	6	1
2.1	Тестирование методом «черного ящика»	8	1
2.1	Тестирование методом «белого ящика»	8	1
2.2	Модульное и интеграционное тестирование	6	2
2.2	Рандомизированное и регрессионное тестирование	8	2
3.1	Разработка автоматизированных тестов	4	1
ИТОГО:		48	10

Практические занятия по дисциплине проводятся в компьютерных классах 303Г, 306Г, 311Г, 261/6, 261/8, 261/17 (главный учебный корпус) кафедры прикладной информатики, оснащенных персональными компьютерами с программным обеспечением и проектором.

Студент в ходе практического занятия реализует технологическое решение заданной задачи. Защита практического задания проводится на основании выполненного задания, оформленного отчета, а также ответа на контрольные вопросы.

При выполнении лабораторных работ студенты используют различные инструменты и среды разработки: IDE (Visual Studio, IntelliJ IDEA), системы контроля версий (Git), инструменты автоматизации тестирования (Selenium, JUnit, TestNG), системы управления тестированием (TestRail, Jira).

5 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ РАСЧЕТНО-ГРАФИЧЕСКОЙ РАБОТЫ

Расчетно-графическая работа (РГР) является обязательным элементом изучения дисциплины и направлена на закрепление теоретических знаний и приобретение практических навыков анализа качества программного обеспечения.

5.1 Цель и задачи РГР

Цель выполнения РГР – углубление и практическое применение теоретических знаний по тестированию и анализу качества ПО, развитие навыков самостоятельной работы с научно-технической литературой и специализированными инструментами.

Задачи РГР:

- изучение методов оценки качества программного обеспечения;
- освоение инструментов анализа качества кода;
- проведение практического анализа качества выбранного ПО;
- разработка рекомендаций по улучшению качества;
- оформление результатов исследования в виде отчета с графическими материалами.

5.2 Тематика РГР

Тематика расчетно-графических работ включает следующие направления:

Анализ качества веб-приложения – оценка качества кода, производительности, безопасности веб-приложения с использованием специализированных инструментов.

Исследование эффективности методов тестирования – сравнительный анализ различных подходов к тестированию на конкретном программном продукте.

Разработка системы метрик качества – создание системы показателей для оценки качества ПО определенного класса.

Автоматизация процесса тестирования – разработка автоматизированного тестового решения для выбранного приложения.

Анализ покрытия кода тестами – исследование различных метрик покрытия и их влияния на качество тестирования.

5.3 Структура РГР

Расчетно-графическая работа должна содержать:

1. Титульный лист – оформляется согласно стандартам вуза.
2. Содержание – с указанием страниц разделов.
3. Введение (1–2 стр.) – обоснование актуальности темы, цель и задачи работы, объект исследования.
4. Теоретическая часть (3–5 стр.) – обзор методов и подходов, относящихся к выбранной теме.
5. Практическая часть (5–8 стр.) – описание проведенных исследований, расчетов, полученных результатов с графиками и диаграммами.

6. Анализ результатов (2–3 стр.) – интерпретация полученных данных, выводы, рекомендации.
7. Заключение (1 стр.) – основные результаты и выводы работы.
8. Список литературы – не менее 8 источников.
9. Приложения – исходный код, скриншоты, дополнительные графики (при необходимости).

5.4 Порядок выполнения и защиты РГР

Выбор темы – согласование с преподавателем до третьей недели семестра.

Выполнение работы – самостоятельная работа студента в течение семестра.

Сдача работы – не позднее, чем за две недели до экзамена.

Защита РГР – публичная защита работы с презентацией (10–15 минут) и ответами на вопросы.

С целью контроля качества работы студентов запланировано выполнение и защита расчетно-графической работы. Система оценивания и критерии оценки приведены в таблице 3.

Таблица 3 – Система оценивания и критерии оценки расчетно-графической работы

Критерий	Система оценок			
	2	3	4	5
	«неудовлетворительно»	«удовлетворительно»	«хорошо»	«отлично»
	«зачтено»			
1 Работа с информацией	Не в состоянии находить необходимую информацию, либо в состоянии находить отдельные фрагменты информации в рамках поставленной задачи	Может найти необходимую информацию в рамках поставленной задачи	Может найти, интерпретировать и систематизировать необходимую информацию в рамках поставленной задачи	Может найти, систематизировать необходимую информацию, а также выявить новые, дополнительные источники информации в рамках поставленной задачи
2 Научное осмысление изучаемого	Не может делать научно-корректных	В состоянии осуществлять научно-	В состоянии осуществлять систематический	В состоянии осуществлять систематичес-

Критерий	Система оценок				
	2	3	4	5	
	«неудовлетворительно»	«удовлетворительно»	«хорошо»	«отлично»	
	«зачтено»				
явления, процесса, объекта	выводов из имеющихся у него сведений, в состоянии проанализировать только некоторые из имеющихся у него сведений	корректный анализ предоставленной информации	и научно-корректный анализ предоставленной информации, вовлекает в исследование новые релевантные задачи	данные	кий и научно-корректный анализ предоставленной информации, вовлекает в исследование новые релевантные поставленные задачи, предлагает новые ракурсы поставленной задачи
3 Освоение стандартных алгоритмов решения профессиональных задач	В состоянии решать только фрагменты поставленной задачи в соответствии с заданным алгоритмом, не освоил предложенный алгоритм, допускает ошибки	В состоянии решать поставленные задачи в соответствии с заданным алгоритмом	В состоянии решать поставленные задачи в соответствии с заданным алгоритмом, понимает основы предложенного алгоритма		Не только владеет алгоритмом и понимает его основы, но и предлагает новые решения в рамках поставленной задачи

Защита РГР является обязательным условием допуска к экзамену.

6 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Самостоятельная работа студентов по дисциплине включает 73 ч для очной формы обучения и проводится в компьютерных классах кафедры прикладной информатики, оснащенных персональными компьютерами с выходом в сеть Интернет, а также в ЭИОС университета.

Самостоятельная работа студента включает в себя:

- освоение теоретического учебного материала;
- подготовку к лабораторным занятиям;
- оформление лабораторных работ;
- подготовку к защите лабораторных работ;
- выполнение расчетно-графической работы;
- подготовку к экзамену.

Наряду с проработкой лекционного материала и подготовкой к лабораторным занятиям, студент заочной формы обучения обязан выполнить индивидуальную контрольную работу.

В качестве задания для контрольной работы студентов заочной формы обучения выбираются (по указанию преподавателя) два вопроса из примерного перечня экзаменационных вопросов по дисциплине.

7 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРОВЕДЕНИЮ ЗАНЯТИЙ И ОСВОЕНИЮ ДИСЦИПЛИНЫ

Для успешного освоения дисциплины необходимо ознакомиться с основными методами и принципами тестирования программного обеспечения.

При разработке образовательной технологии организации учебного процесса основной упор сделан на соединение активной и интерактивной форм обучения. Интерактивная форма позволяет студентам проявить самостоятельность в освоении теоретического материала и овладении практическими навыками, формирует интерес и позитивную мотивацию к учебе.

При изучении дисциплины внимание студентов постоянно акцентируется не только на теоретическом аспекте методов тестирования, но и их практическом применении при обеспечении качества программного обеспечения в различных областях разработки.

Для планирования работы студента в начале семестра производится выдача тем для самостоятельного изучения и тем расчетно-графических работ, определяются источники информации и график проведения текущего контроля. В качестве источников информации рекомендуется наряду с учебными пособиями использовать профессиональные интернет-ресурсы и документацию инструментов тестирования.

В ходе лекционных занятий студенту следует вести конспектирование учебного материала. При самостоятельном изучении заданных преподавателем тем рекомендуется

вносить основные материалы по ним в тот же конспект лекций в соответствии с рекомендованным порядком следования учебного материала.

При проведении занятий в интерактивной форме важно участвовать в процессе обсуждения и решения поставленных задач тестирования и анализа качества ПО, задавать преподавателю вопросы с целью уяснения теоретических положений, области их применения, разрешения спорных ситуаций.

Активность студентов и проявленные знания при обсуждении материала учитываются при текущей и промежуточной (заключительной) аттестации по дисциплине.

8 ТРЕБОВАНИЯ К АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ

8.1 Текущая аттестация

Текущая аттестация (текущий контроль) проводится с целью оценки освоения теоретического учебного материала, в том числе в рамках самостоятельной работы студента.

Контроль на лекциях и лабораторных занятиях производится в виде тестирования или устного опроса.

Типовые контрольные вопросы для устного опроса по темам приведены в п. 3 настоящего пособия. Тестовые задания представлены в фонде оценочных средств (приложение к рабочей программе дисциплины).

Положительная оценка («зачтено») по результатам каждого контроля (опроса) выставляется в соответствии с универсальной системой оценивания, приведенной в таблице 4. В случае получения оценки «не зачтено» студент должен пройти повторный контроль по данной теме в ходе последующих консультаций.

Таблица 4— Система оценок и критерии выставления оценки при прохождении контроля (опроса)

Критерий	«Не зачтено»	«Зачтено»
Системность и полнота знаний в отношении изучаемых объектов	Обладает частичными и разрозненными знаниями, которые не может научно-корректно связывать между собой	Обладает минимальным набором знаний, необходимым для системного взгляда на изучаемый объект

Текущий контроль в виде защиты лабораторных работ проводится на лабораторном практикуме, целью которого является формирование умений и навыков по тестированию и анализу качества программного обеспечения. Оценка результатов выполнения задания по каждой лабораторной работе производится при представлении студентом отчета по лабораторной работе и на основании ответов студента на вопросы по тематике лабораторной работы. Студент, самостоятельно выполнивший задание, продемонстрировавший знание использованных им методов и инструментов тестирования, получает по лабораторной работе оценку «зачтено».

Текущий контроль в виде защиты расчетно-графической работы проводится в соответствии с требованиями, изложенными в п. 5 данного пособия.

Текущий контроль в виде защиты лабораторных работ проводится на лабораторном практикуме, целью которого является формирование умений и навыков по программированию и настройке КИС. Оценка результатов выполнения задания по каждой лабораторной работе производится при представлении студентом отчета по лабораторной работе и на основании ответов студента на вопросы по тематике лабораторной работы. Студент, самостоятельно выполнивший задание, продемонстрировавший знание использованных им технических средств, алгоритмов и языков программирования задачи, получает по лабораторной работе оценку «зачтено».

Положительная оценка («зачтено») по каждому заданию выставляется, если задание выполнено без ошибок. Если студентом получены неправильные результаты или действия выполнены неправильно, то задание получает оценку «не зачтено» и отправляется на доработку.

8.2 Промежуточная аттестация по дисциплине

Промежуточная (заключительная) аттестация по дисциплине предусматривает проведение экзамена (экзаменационного тестирования).

К экзамену допускаются студенты:

- выполнившие и защитившие все предусмотренные лабораторные работы (получившие положительную оценку по результатам лабораторного практикума);
- выполнившие и защитившие расчетно-графическую работу (получившие оценку «зачтено» по РГР).

Экзамен может проводиться как в традиционной форме, так и в виде тестирования. Тестовые задания приведены в ФОС по дисциплине.

Экзаменационный билет содержит два экзаменационных вопроса.

Примерный перечень экзаменационных вопросов по дисциплине

1. Основные понятия и принципы тестирования программного обеспечения.
2. Жизненный цикл разработки ПО и роль тестирования.
3. Семь принципов тестирования по ISTQB.
4. Процесс проектирования тестов и создание тест-планов.
5. Техника эквивалентного разбиения и её применение.
6. Анализ граничных значений в тестировании.
7. Предположение об ошибке и исследовательское тестирование.
8. Структурное тестирование: покрытие операторов и ветвей.
9. Цикломатическая сложность и её использование в тестировании.
10. Модульное тестирование: принципы и инструменты.
11. Интеграционное тестирование: стратегии и подходы.
12. Особенности тестирования объектно-ориентированных систем.
13. Рандомизированное тестирование и фаззинг.
14. Регрессионное тестирование: цели и стратегии.
15. Автоматизация тестирования: принципы и инструменты.
16. Пирамида тестирования и её значение.
17. Паттерны проектирования в автоматизированном тестировании.
18. Continuous Integration и Continuous Testing.
19. Метрики качества программного обеспечения.
20. Управление дефектами и отслеживание ошибок.
21. Оценка оттестированности проекта.
22. Риск-ориентированное тестирование.
23. Тестирование безопасности и производительности.
24. Инструменты автоматизации тестирования (Selenium, JUnit).
25. ROI автоматизации тестирования и критерии выбора тестов для автоматизации.
26. Тестовые задания для проведения экзаменационного тестирования приведены в фонде оценочных средств по дисциплине.

Система оценивания и критерии выставления оценок по экзамену (экзаменационному тестированию) соответствуют универсальной системе оценивания результатов обучения (таблица 5).

Таблица 5– Система оценок и критерии выставления оценки на экзамене

Оценка	Критерии
«Отлично»	Студент демонстрирует глубокие системные знания по всем разделам дисциплины, умеет применять теоретические знания на практике, способен решать сложные задачи тестирования
«Хорошо»	Студент показывает хорошие знания по основным разделам дисциплины с незначительными пробелами, умеет применять знания для решения стандартных задач тестирования
«Удовлетворительно»	Студент демонстрирует базовые знания основных понятий и методов тестирования, может решать простые практические задачи под руководством преподавателя
«Неудовлетворительно»	Студент не владеет основными понятиями дисциплины, не может применить знания для решения практических задач

9 ЗАКЛЮЧЕНИЕ

Тестирование программного обеспечения стало неотъемлемой частью современного процесса разработки, играя ключевую роль в обеспечении качества программных продуктов. В условиях возрастающей сложности программных систем и требований к их надежности, знания и навыки в области тестирования становятся критически важными для специалистов в области информационных технологий.

Освоение дисциплины «Тестирование и анализ качества ПО» с объемом 6 зачетных единиц (216 ч) является одним из важнейших компонентов подготовки будущего специалиста в области прикладной информатики. Комплексный подход к изучению дисциплины, включающий лекционные и лабораторные занятия, выполнение расчетно-графической работы и сдачу экзамена, обеспечивает формирование как теоретических знаний, так и практических навыков.

Приобретенные в ходе изучения дисциплины знания, умения и навыки будут углубляться и совершенствоваться в процессе дальнейшего обучения и могут быть применены при выполнении выпускной квалификационной работы, а также в профессиональной деятельности в области разработки и сопровождения программного обеспечения.

10 БИБЛИОГРАФИЧЕСКИЙ СПИСОК

Основная литература

1. Чернов, Е. А. Тестирование и верификация ПО: учеб. пособие / Е. А. Чернов, М. А. Овчинникова, Д. Е. Новичков. – Москва: РТУ МИРЭА, 2024. – 107 с. – Режим доступа: для авториз. пользователей. – Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/432665> (дата обращения: 25.11.2024). – ISBN 978-5-7339- 2255-3. – Текст : электронный.
2. Проскуряков, А. В. Качество и тестирование программного обеспечения. Метрология программного обеспечения: учеб. пособие / А. В. Проскуряков; Южный федеральный университет, Инженерно-технологическая академия. – Ростов-на-Дону; Таганрог : Южный федеральный университет, 2022. – 199 с. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=69874> 2 (дата обращения: 30.09.2024). – ISBN 978-5-9275- 4044-0. – Текст : электронный.
3. Попова, Ю. Б. Тестирование и отладка программного обеспечения: учеб. пособие / Ю. Б. Попова. – Минск: БНТУ, 2020. – 66 с. – Режим доступа: для авториз. пользователей. – Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/248642> (дата обращения: 26.11.2024). – ISBN 978-985-583-056-7. – Текст : электронный.
4. Мякишев, Д. В. Принципы и методы создания надежного программного обеспечения АСУТП: учеб. пособие / Д. В. Мякишев. – 2-е изд. – Москва Вологда: Инфра-Инженерия, 2021. – 116 с. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=61722> 5 (дата обращения: 26.11.2024). – ISBN 978-5-9729- 0674-1. – Текст : электронный.
5. Сорока, Е. Г. Управление качеством программного продукта: учеб. пособие для вузов / Е. Г. Сорока. – Санкт-Петербург: Лань, 2021. – 100 с. – Режим доступа: для авториз. пользователей. – Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/176878> (дата обращения: 25.11.2024). – ISBN 978-5-8114-7519-3. – Текст : электронный.

Дополнительная литература

1. Программирование, тестирование, проектирование, нейросети, технологии аппаратно-программных средств (практические задания и способы их решения): учебник / С. В. Веретехина, К. С. Кармицкий, Д. Д. Лукашин [и др.]. – Москва: Директ-Медиа, 2022. – 144 с. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=694782> (дата обращения: 30.09.2024). – ISBN 978-5-4499-3321-8. – DOI 10.23681/694782. – Текст : электронный.

2. Семахин, А. М. Методы верификации и оценки качества программного обеспечения: учеб. пособие / А. М. Семахин. – Курган: КГУ, 2018. – 150 с. – Режим доступа: для авториз. пользователей. – Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/177908> (дата обращения: 01.10.2024). – ISBN 978-5-4217-0461-4. – Текст : электронный.

3. Миронов, А. И. Тестирование и верификация программного обеспечения. Практикум: учеб. пособие / А. И. Миронов, С. М. Трушин, А. А. Петренко. – Москва: РТУ МИРЭА, 2022. – 65 с. – Режим доступа: для авториз. пользователей. – Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/240095> (дата обращения: 26.11.2024). – Текст : электронный.

4. Мякишев, Д. В. Принципы и методы создания надежного программного обеспечения АСУТП: практическое пособие / Д. В. Мякишев. – Москва; Вологда: Инфра-Инженерия, 2017. – 115 с. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=466489> (дата обращения: 26.11.2024). – ISBN 978-5-9729-0179-1. – Текст : электронный.

5. Игнатьев, А. В. Тестирование программного обеспечения / А. В. Игнатьев. – 3-е изд., стер. – Санкт-Петербург: Лань, 2023. – 56 с. – Режим доступа: для авториз. пользователей. – Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/269873> (дата обращения: 26.11.2024). – ISBN 978-5-507-45425-9. – Текст : электронный.

6. Баланов, А. Н. Комплексное руководство по разработке: от мобильных приложений до веб-технологий: учеб. пособие для вузов / А. Н. Баланов. – Санкт-Петербург: Лань, 2024. – 412 с. – Режим доступа: для авториз. пользователей. – Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/394577> (дата обращения: 01.10.2024). – ISBN 978-5-507-48841-4. – Текст : электронный.

Учебно-методические пособия, нормативная литература

1. Алпатов, А. Н. Тестирование и отладка программного обеспечения: методы. указания / А. Н. Алпатов. – Москва: РТУ МИРЭА, 2020. – 40 с. – Режим доступа: для авториз. пользователей. – Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/167578> (дата обращения: 27.11.2024). – Текст : электронный.

2. Говоров, П. М. Расчет показателей надежности при оценке качества программного обеспечения: учеб.-метод. пособие / П. М. Говоров; сост. П. М. Говоров. – Москва: МГУПП, 2022. – 20 с. – Режим доступа: для авториз. пользователей. – Лань : электронно-библиотечная

система. – URL: <https://e.lanbook.com/book/277127> (дата обращения: 01.10.2024). – ISBN 978-5-9920-0393-2. – Текст : электронный.

3. Бергер, Е. Г. Единая система программной документации: учеб.-метод. пособие / Е. Г. Бергер. – Москва: РТУ МИРЭА, 2020. – 109 с. – Режим доступа: для авториз. пользователей. – Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/163817> (дата обращения: 27.11.2024). – Текст : электронный.

Локальный электронный методический материал

Анастасия Михайловна Лутовинова

ТЕСТИРОВАНИЕ И АНАЛИЗ КАЧЕСТВА ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ

Редактор С. Кондрашова
Корректор Т. Звада

Уч.-изд. л. 1,7. Печ. л. 1,5.

Издательство федерального государственного бюджетного
образовательного учреждения высшего образования
«Калининградский государственный технический университет».
236022, Калининград, Советский проспект, 1