

Белорусский государственный университет

УТВЕРЖДАЮ

Проректор по учебной работе и
образовательным инновациям

О. Г. Прохоренко

5 декабря 2023 г.

Регистрационный № 12679/уч.

ТЕСТИРОВАНИЕ И ОЦЕНКА КАЧЕСТВА РАБОТЫ СИСТЕМ

**Учебная программа учреждения высшего образования
по учебной дисциплине для специальностей:**

1-31 03 07 Прикладная информатика (по направлениям)

направление специальности:

1-31 03 07-01 Прикладная информатика (программное
обеспечение компьютерных систем)

2023 г.

Учебная программа составлена на основе образовательного стандарта высшего образования ОСВО 1-31 03 07-2021, типового учебного плана №G 31-1-030/пр-тип от 01.07.2021, учебных планов №G 31-1-034/уч. от 23.07.2021, №G 31-1-023/уч. ин от 09.08.2021, №G 31-1-216/уч. от 22.03.2022, №G 31-1-224/уч. ин. от 27.05.2022 г.

СОСТАВИТЕЛИ:

С.Е. Гутников – старший преподаватель кафедры информационных систем управления Белорусского государственного университета.

РЕЦЕНЗЕНТ:

В.В. Лепин – ведущий научный сотрудник института математики НАН Беларуси, кандидат физико-математических наук, доцент.

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой информационных систем управления БГУ
(протокол № 3 от 19.10.2023 г.).

Научно-методическим Советом БГУ
(протокол № 3 от 30.11.2023 г.)

Заведующий кафедрой



В.В. Краснопрошин

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Учебная дисциплина «Тестирование и оценка качества работы систем» знакомит студентов с задачами, принципами и способами тестирования программного обеспечения (ПО). Особое внимание в учебной дисциплине уделяется методам тестирования, проектированию тестов и оценке результатов тестирования.

Рассматриваются вопросы организации процесса тестирования программного продукта включая интеграцию и системное тестирование. В учебной дисциплине также рассматриваются характеристики качества программ, влияние тестирования на качество продукта и методы оценки качества ПО.

Цели и задачи учебной дисциплины

Цель преподавания учебной дисциплины «Тестирование и оценка качества работы систем» – формирование у студентов фундаментальных знаний и приобретение студентами практических навыков в области современной теории и практики тестирования ПО.

При изложении материала учебной дисциплины важно показать возможности использования методов тестирования при решении прикладных задач, возникающих в различных областях науки, техники, экономики и др. Целесообразно выделить моменты анализа математических моделей процессов и задач с целью их последующего тестирования.

Задачи учебной дисциплины

В рамках поставленной цели основные задачи учебной дисциплины «Тестирование и оценка качества работы систем» состоят в следующем:

1. исследование предметной области под названием: «Тестирование и оценка качества работы систем»;
2. формирование у студентов основных понятий: подход, стратегия, критерий, драйвер, заглушка тестирования;
3. изучение техники и технологии автоматизированного тестирования с целью использования при решении практических задач;
4. использование современных инструментов для модульного и функционального тестирования при решении прикладных задач.

Место учебной дисциплины в системе подготовки специалиста с высшим образованием.

Учебная дисциплина относится к **модулю «Информационные системы»** компонента учреждения образования.

Программа составлена с учётом межпредметных **связей** с учебными дисциплинами. Основой для обучения являются дисциплины модуля «Программирование», изучаемые в 1-м, 2-м и 3-м семестрах: «Основы и методологии программирования», «Разработка кросс-платформенных приложений», «Машинно-ориентированное программирование», «Промышленное программирование».

Знания, полученные в учебной дисциплине, используются при изучении всех дисциплин специализации, при выполнении курсовых и дипломных работ.

Требования к компетенциям

Освоение учебной дисциплины «Тестирование и оценка качества работы систем» должно обеспечить формирование следующей **специализированной компетенции:**

СК-6. Создавать модели данных и проектировать базы данных для разработки систем разного типа, тестировать и оценивать качество и безопасность информационных систем.

В результате освоения учебной дисциплины студент должен:

знать:

- принципы, способы, методы и методику тестирования;
- особенности тестирования объектно-ориентированных программ;
- основные способы и методы оценки качества продукта;
- стандарты процесса разработки программного обеспечения;
- основные способы и методы оценки качества продукта;
- принципы работы с различными инструментами для тестирования;

уметь:

- строить системы тестов для различных критериев тестирования;
- выполнять ручное тестирование;
- осуществлять поддержку тестирования;
- определять основные качественные характеристики программ;
- получать качественные метрики программного продукта;
- пользоваться современными CASE-средствами проектирования и тестирования ПО;

владеть:

- основными методами тестирования ПО;
- методами оценки качества и надежности ПО;
- навыками самообразования и способами использования приемов и методов тестирования в процессе разработки программных продуктов.

Структура учебной дисциплины

Дисциплина изучается в 6-м семестре. Всего на изучение учебной дисциплины «Тестирование и оценка качества работы систем» отведено:

– для очной формы получения высшего образования отведено 90 часов, в том числе 34 аудиторных часа, из них: лекции – 18 часов, лабораторные занятия – 14 часов, управляемая самостоятельная работа – 2 часа.

Трудоёмкость учебной дисциплины составляет 3 зачётных единицы.

Форма промежуточной аттестации – **зачёт**.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Введение

Тема 1.1. Тестирование и качество систем в программной инженерии

Основные определения. Жизненный цикл и требования к программному обеспечению. Атрибуты качества. Цели и задачи тестирования и оценки качества программных систем. Место и роль тестирования и оценки качества в различных моделях процесса разработки.

Тема 1.2. Качество исходного кода и парадигмы программирования

Ограничения, связанные с парадигмой программирования, качество исходного кода. Принципы дизайна SOLID.

Раздел 2. Тестирование и отладка

Тема 2.1. Модульное тестирование

Модульное тестирование и его задачи. Обзоры программного кода. Тестирования структуры модулей. Тестирование взаимодействия модулей. Стратегии пошагового тестирования. Объектно-ориентированное тестирование. Автоматизация модульного тестирования. Семейство xUnit. Универсальные средства автоматизации тестирования.

Тема 2.2. Планирование и реализация функционального тестирования

Планирование функционального тестирования. Тестовый план. Разработка тестовых случаев. Выбор тестовых данных. Ошибка, свойства ошибки. Составления отчетов об ошибках. Системы документирования/отслеживания ошибок. Жизненный цикл ошибки. Реализация тестов: приемочный тест, позитивный тест, негативный тест. Особенности функционального тестирования.

Тема 2.3. Автоматизация тестирования и отладка ПО

Плюсы/минусы автоматизации тестирования. Требования к автоматизированным тестам. Методы автоматизации функционального тестирования. Метод Play&Record. Метод функциональной декомпозиции. Метод Data-driven. Метод Keyword-driven. Семейство Selenium и его возможности. Проблемы внедрения автоматизации тестирования. Методика отладки. Методы отладки. Средства отладки.

Тема 2.4. Методики разработки через тестирование

Agile-методологии разработки. Модель MSF for Agile Software Development. Модель Rational Unified Process (RUP). Модель Extreme Programming (XP). Разработка через тестирование. TDD - Test Driven Development. ATDD - Acceptance TDD. BDD - Behaviour Driven Development. Сравнение TDD, BDD и ATDD.

Раздел 3. Вопросы оценки качества программных систем

Тема 3.1. Основы создания качественных программных систем

Стандартизация качества ПО. Роль стандартизации в управлении качеством. Оценка качества ПО в соответствии со стандартом. Стандарт ISO 9126. Сертификация программного обеспечения. Основные понятия сертификации. Требования к качеству функционирования программного продукта. Методики оценки качества при сертификации. Критерии оценки качества ПО.

Тема 3.2. Модели оценки сложности программных модулей

Модели оценки сложности ПО. Структурные и статистические характеристики сложности программных модулей. Влияние сложности ПО на его надежность.

Тема 3.3. Оценка качества проекта сложной программной системы

Метрики лексического анализа программ. Метрики структурной сложности программ. Процедурно-ориентированные метрики. Объектно-ориентированные метрики.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Дневная форма получения образования с применением электронных средств обучения (ДОТ)

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Форма контроля знаний
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное		
1	2	3	4	5	6	7	8	9
	Тестирование и оценка качества работы систем	18			14		2	
1.	Введение	4					2	
1.1.	Тестирование и качество систем в программной инженерии	2						Устный опрос.
1.2.	Качество исходного кода и парадигмы программирования	2					2	УСР. Лабораторная работа №1, отчет по лабораторному заданию с устной защитой.
2.	Тестирование и отладка	8			8			
2.1.	Модульное тестирование	2			2			Лабораторная работа №2, отчет по лабораторному заданию с устной защитой.
2.2.	Планирование и реализация функционального тестирования	2			2			Лабораторная работа №3, отчет по лабораторному заданию с устной защитой.
2.3.	Автоматизация тестирования и отладка ПО	2			2			Лабораторная работа №4, отчет по лабораторному заданию с устной защитой.
2.4.	Методики разработки через тестирование	2			2			Расчетно-графическое задание №1, отчет по расчетно-графическому заданию с устной защитой.
3.	Вопросы оценки качества программных систем	6			6			
3.1.	Основы создания качественных	2			2			Расчетно-

	программных систем						графическое задание №2, отчет по расчетно-графическому заданию с устной защитой.
3.2.	Модели оценки сложности программных модулей	2			2		Расчетно-графическое задание №3, отчет по расчетно-графическому заданию с устной защитой.
3.3.	Оценка качества проекта сложной программной системы	2			2		Расчетно-графическое задание №4, отчет по расчетно-графическому заданию с устной защитой. Коллоквиум по разделам 1,2,3.

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Перечень основной литературы

1. Попова, Ю. Б. Тестирование и отладка программного обеспечения: пособие / Ю. Б. Попова. – Минск : БНТУ, 2020. – 66 с.
2. Маран, М. М. Программная инженерия: учебное пособие / М. М. Маран. - Изд. 3-е, стер. - Санкт-Петербург; Москва ; Краснодар : Лань, 2022. - 194 с. - URL: <https://reader.lanbook.com/book/189470#2>.
3. Хориков, В. Принципы юнит-тестирования / Хориков В. - СПб.: Питер. Для профессионалов, 2021. - 320 с. - URL: <https://ibooks.ru/bookshelf/373514>.
4. Мартин Р. Чистая архитектура. Искусство разработки программного обеспечения. - СПб.: Питер, 2018. - 352 с. - URL: <https://ibooks.ru/bookshelf/361841>.

Перечень дополнительной литературы

1. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем: учебник для вузов / Е. М. Лаврищева. - Москва: Издательство Юрайт, 2022. - 432 с.
2. Оукс Скотт. Эффективный Java. Тюнинг кода на Java 8, 11 и дальше. — СПб.: Питер, 2021. — 496 с.
3. Маттиас Нобак. Принципы разработки программных пакетов: Проектирование повторно используемых компонентов / пер. с англ. Д. А. Беликова. – М.: ДМК Пресс, 2020. – 274 с.
4. Вигерс К., Битти Д. Разработка требований к программному обеспечению, издание третье. – М: Русская редакция, 2018. – 716 с.
5. Макконнелл С. Совершенный код. Мастер класс / Пер. с англ. — М.: Издательство «Русская редакция», СПб. : Питер, 2016. — 896 с.
6. Бахтизин В.В., Глухова Л.А. Метрология, стандартизация и сертификация в информационных технологиях. Учебное пособие. — В 2-х частях. — Минск: БГУИР, 2016.
7. Буч Г., Рамбо Д., Джекобсон А. Язык UML. Руководство пользователя. Пер. с англ. – М.: ДМК Пресс, 2014. – 496 с.
8. Ошероув Р. Искусство автономного тестирования с примерами на C#. 2-е издание / пер. с англ. Слинкин А. А. – М.: ДМК Пресс, 2014. – 360 с.
9. Мартин Р. Чистый код: создание, анализ и рефакторинг. Библиотека программиста. — СПб.: Питер, 2013. — 464 с.
10. Software Engineering компьютерных систем. Парадигмы, технологии и CASE-средства программирования/Е. М. Лаврищева. –К.:Наук. думка, 2013. – 283 с.

11. Липаев В.В. Тестирование компонентов и комплексов программ. Учебник. – М.: СИНТЕГ, 2010. – 400 с.

Электронные ресурсы

1. Образовательный портал БГУ [Электронный ресурс]. – Режим доступа: <https://edufpmi.bsu.by/course/view.php?id=873> – Дата доступа: 30.11.2023.

Перечень рекомендуемых средств диагностики и методика формирования итоговой отметки

Для диагностики компетенции в рамках учебной дисциплины рекомендуется использовать следующие формы:

1. Устная форма: устный опрос.
2. Письменная форма: лабораторные и расчетно-графические задания, коллоквиум.
3. Устно-письменная форма: отчеты по лабораторным и расчетно-графическим заданиям с их устной защитой, оценивание на основе проектного метода.

В качестве рекомендуемых технических средств используется обучение, организованное на платформе Moodle (<https://edufpmi.bsu.by>).

Формой промежуточной аттестации по дисциплине учебным планом предусмотрен **зачёт**.

При формировании итоговой отметки используется рейтинговая система оценки знания студента, дающая возможность проследить и оценить динамику процесса достижения целей обучения. Рейтинговая система предусматривает использование весовых коэффициентов для текущего контроля знаний и текущей аттестации студентов по дисциплине.

Примерные весовые коэффициенты, определяющие вклад текущего контроля знаний в итоговую отметку:

- отчеты по лабораторным заданиям с их устной защитой – 40 %;
- отчеты по расчетно-графическим заданиям с их устной защитой – 40 %;
- коллоквиум – 20%;

Отметка «зачтено» выставляется студенту, имеющему отметку за текущую успеваемость не ниже 4 («четырёх») баллов.

Точки контроля по текущей успеваемости формируются из расчета общего количества часов (зачетных единиц), выделенных на изучение дисциплины.

Примерный перечень заданий для управляемой самостоятельной работы студентов

В качестве заданий для управляемой самостоятельной работы могут быть выданы задания для самостоятельного решения задач по следующим темам:

1. **Тема 1.1.** Тестирование и качество систем в программной инженерии (2 ч.)

Форма контроля – Лабораторная работа №1, отчет по лабораторному заданию с устной защитой.

Примерная тематика лабораторных занятий

Занятие №1. Тема 2.1. Модульное тестирование. Лабораторная работа №2.

Занятие №2. Тема 2.2. Планирование и реализация функционального тестирования. Лабораторная работа №3.

Занятие №3. Тема 2.3. Автоматизация тестирования и отладка ПО. Лабораторная работа №4.

Занятие №4. Тема 2.4. Методики разработки через тестирование. Расчетно-графическое задание №1.

Занятие №5. Тема 3.1. Основы создания качественных программных систем. Расчетно-графическое задание №2.

Занятие №6. Тема 3.2. Модели оценки сложности программных модулей. Расчетно-графическое задание №3.

Занятие №7. Тема 3.3. Оценка качества проекта сложной программной системы. Расчетно-графическое задание №4.

Описание инновационных подходов и методов к преподаванию учебной дисциплины

При организации образовательного процесса используются следующие методы:

- **метод учебной дискуссии**, который предполагает участие студентов в целенаправленном обмене мнениями, идеями для предъявления и/или согласования существующих позиций по определенной проблеме. Использование метода обеспечивает появление нового уровня понимания изучаемой темы, применение знаний (теорий, концепций) при решении проблем, определение способов их решения.

- **метод группового обучения**, который представляет собой форму организации учебно-познавательной деятельности обучающихся, предполагающую функционирование разных типов малых групп, работающих как над общими, так и специфическими учебными заданиями.

В качестве технических средств организации работы в рамках учебной дисциплины рекомендуется использовать Образовательный портал БГУ (<https://edufpmi.bsu.by>) – инструмент с эффективной функциональностью контроля, тренинга и самостоятельной работы.

- **практико-ориентированный подход**, который предполагает освоение содержания образования через решения практических задач; приобретение навыков эффективного выполнения разных видов профессиональной деятельности; ориентацию на генерирование идей, реализацию групповых

студенческих проектов; использование процедур, способов оценивания, фиксирующих профессиональные компетенции.

Методические рекомендации по организации самостоятельной работы обучающихся

Для организации самостоятельной работы студентов по учебной дисциплине следует использовать современные информационные ресурсы: разместить на образовательном портале комплекс учебных и учебно-методических материалов (учебно-программные материалы, учебное издание для теоретического изучения дисциплины, методические указания к лабораторным занятиям, материалы текущего контроля и текущей аттестации, позволяющие определить соответствие учебной деятельности обучающихся требованиям образовательных стандартов высшего образования и учебно-программной документации, в т.ч. вопросы для подготовки к зачету, задания, тесты, вопросы для самоконтроля и др., список рекомендуемой литературы, информационных ресурсов и др.).

При составлении заданий УСР по учебной дисциплине необходимо предусмотреть следующие материалы для выдачи студентам:

1. Порядок работы – даётся пошаговая инструкция по выполнению задания;
2. Теоретический материал – темы для изучения перед выполнением задания;
3. Примеры решения типовых задач по изучаемому теоретическому материалу;
4. Условие и варианты задания – подробное описание условия задания и, опционально, варианты заданий.

С примерными вариантами методических материалов к индивидуальным заданиям и заданиям для УСР можно ознакомиться на образовательном портале БГУ (<https://edufpmi.bsu.by/course/view.php?id=873>)

Примерный перечень вопросов к зачёту

1. Определения требований, атрибуты качества.
2. Парадигмы программирования, ограничения, связанные с парадигмой программирования.
3. Принципы дизайна SOLID, принцип единственной ответственности.
4. Принципы дизайна SOLID, принцип открытости / закрытости.
5. Принципы дизайна SOLID, принцип подстановки Барбары Лисков.
6. Принципы дизайна SOLID, принцип разделения интерфейсов.
7. Принципы дизайна SOLID, принцип инверсии зависимости.
8. Модели промышленных технологий создания ПП, Agile-методологии разработки.
9. Модели промышленных технологий создания ПП, модель MSF for Agile Software Development.

10. Модели промышленных технологий создания ПП, модель Rational Unified Process (RUP).
11. Модели промышленных технологий создания ПП, модель Extreme Programming (XP).
12. Виды тестирования ПО. Экономическая сторона тестирования.
13. Модульное тестирование, модульное тестирование и его задачи.
14. Модульное тестирование, обзоры программного кода.
15. Модульное тестирование, тестирования структуры модулей.
16. Модульное тестирование, тестирование взаимодействия модулей.
17. Модульное тестирование, стратегии пошагового тестирования.
18. Модульное тестирование, объектно-ориентированное тестирование.
19. Модульное тестирование, автоматизация модульного тестирования.
20. Модульное тестирование, семейство xUnit.
21. Модульное тестирование, универсальные средства автоматизации тестирования.
22. Функциональное тестирование, планирование функционального тестирования.
23. Функциональное тестирование, тестовый план, разработка тестовых случаев, выбор тестовых данных.
24. Функциональное тестирование, ошибка, свойства ошибки, составления отчетов об ошибках.
25. Функциональное тестирование, системы документирования/отслеживания ошибок, жизненный цикл ошибки.
26. Функциональное тестирование, реализация тестов: приемочный тест, позитивный тест, негативный тест.
27. Функциональное тестирование, особенности функционального тестирования.
28. Автоматизация функционального тестирования, плюсы/минусы автоматизации тестирования.
29. Автоматизация функционального тестирования, требования к автоматизированным тестам.
30. Автоматизация функционального тестирования, методы автоматизации функционального тестирования.
31. Автоматизация функционального тестирования, метод Play&Record.
32. Автоматизация функционального тестирования, метод функциональной декомпозиции.
33. Автоматизация функционального тестирования, метод Data-driven.
34. Автоматизация функционального тестирования, метод Keyword-driven.
35. Автоматизация функционального тестирования, семейство Selenium и его возможности.
36. Автоматизация функционального тестирования, проблемы внедрения автоматизации тестирования.
37. Отладка ПО, методика отладки.
38. Отладка ПО, методы отладки.

39. Отладка ПО, средства отладки.
40. Сравнение TDD, BDD и ATDD.
41. Стандартизация качества ПО.
42. Роль стандартизации в управлении качеством.
43. Оценка качества ПО в соответствии со стандартом.
44. Стандарт ISO 9126.
45. Сертификация ПО, основные понятия сертификации.
46. Сертификация ПО, требования к качеству функционирования ПП.
47. Сертификация ПО, методики оценки качества ПС при сертификации.
48. Метрики Холстеда.
49. Метрики Джилба.
50. Метрики Чепина.
51. Метрики структурной сложности программ.
52. Метрика Маккейба.
53. Функциональные указатели и метрики свойств.
54. Связность программных модулей.
55. Сцепление программных модулей.
56. Метрики Мартина.
57. Метрики Чидамбера и Кемерера.
58. Метрики Лоренца и Кидда.
59. Метрики Абреу.

**ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ
ПО ИЗУЧАЕМОЙ УЧЕБНОЙ ДИСЦИПЛИНЕ
С ДРУГИМИ ДИСЦИПЛИНАМИ СПЕЦИАЛЬНОСТИ**

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
Согласований не требуется			

**ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ
ПО ИЗУЧАЕМОЙ УЧЕБНОЙ ДИСЦИПЛИНЕ
на ____ / ____ учебный год**

№№ ПП	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры информационных систем управления (протокол № ____ от _____ 200_ г.)

Заведующий кафедрой

_____ (степень, звание)

_____ (подпись)

_____ (И.О.Фамилия)

УТВЕРЖДАЮ
Декан факультета

_____ (степень, звание)

_____ (подпись)

_____ (И.О.Фамилия)