

УТВЕРЖДАЮ
 Декан факультета

 (подпись) Матвеев П.В.
 ФИО
 «___» _____ 20__

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ ТЕСТИРОВАНИЕ И ВЕРИФИКАЦИЯ ПРОГРАММНЫХ ПРОДУКТОВ

Направление/специальность подготовки	09.03.04 Программная инженерия
Специализация/профиль/программа подготовки	Разработка программно-информационных систем
Уровень высшего образования	Бакалавриат
Форма обучения	Очная
Факультет	О Естественнонаучный
Выпускающая кафедра	О7 Информационные системы и программная инженерия
Кафедра-разработчик рабочей программы	О7 Информационные системы и программная инженерия

КУРС	СЕМЕСТР	ОБЩАЯ ТРУДОЁМКОСТЬ (ЗАЧЕТНЫХ ЕДИНИЦ)	ЧАСЫ (по наличию видов занятий)									ВИД ПРОМЕЖУТОЧНОГО КОНТРОЛЯ
			ОБЩАЯ ТРУДОЁМКОСТЬ	АУДИТОРНЫЕ ЗАНЯТИЯ				САМОСТОЯТЕЛЬНАЯ РАБОТА				
				ВСЕГО	ЛЕКЦИИ	ЛАБОРАТОРНЫЙ ПРАКТИКУМ	ПРАКТИЧЕСКИЕ ЗАНЯТИЯ	ВСЕГО	КУРСОВОЙ ПРОЕКТ	КУРСОВАЯ РАБОТА	ДРУГИЕ ВИДЫ САМОСТ. РАБОТЫ	
4	7	5	180	68	0	0	68	112	0	0	112	диф. зач.

ЛИСТ СОГЛАСОВАНИЯ

РАБОЧАЯ ПРОГРАММА СОСТАВЛЕНА В СООТВЕТСТВИИ С ТРЕБОВАНИЯМИ ФЕДЕРАЛЬНОГО
ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО СТАНДАРТА ВЫСШЕГО ОБРАЗОВАНИЯ (ФГОС ВО)

09.03.04 Программная инженерия

год набора группы: 2024

Программу составил:

Кафедра О7 Информационные системы и программная инженерия
Смирнова Мария Сергеевна, д.т.н., доцент, профессор

Программа рассмотрена
на заседании кафедры-разработчика
рабочей программы **О7 Информационные системы и программная инженерия**

Заведующий кафедрой Семенова Е.Г., д.т.н., проф.

Программа рассмотрена
на заседании выпускающей кафедры

О7 Информационные системы и программная инженерия

Заведующий кафедрой Семенова Е.Г., д.т.н., проф.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ТЕСТИРОВАНИЕ И ВЕРИФИКАЦИЯ ПРОГРАММНЫХ ПРОДУКТОВ

Разделы рабочей программы

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО
3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ
4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ
5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ
6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Приложения к рабочей программе дисциплины

- Приложение 1. Аннотация рабочей программы
- Приложение 2. Технологии и формы обучения
- Приложение 3. Фонды оценочных средств

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-6 — способность разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов
--

ПСК-1/23.1 — способность использовать методы и инструментальные средства исследования объектов профессиональной деятельности на всех этапах жизненного цикла программных средств
--

Формированию компетенций служит достижение следующих результатов образования:

ОПК-6

знания:

На уровне представлений: основной математический аппарат, применяемый при тестировании, верификации и валидации программных продуктов.

На уровне воспроизведения: понятие сильной и слабой бисимуляции агентов.;

умения:

формализовывать спецификацию процессов в программных продуктах автоматной моделью.;

навыки:

эмуляции взаимодействия с пользовательским интерфейсом программных продуктов..

ПСК-1/23.1

знания:

основные этапы жизненного цикла разработки программного обеспечения; основного математического аппарата, применяемого при тестировании, верификации и валидации программных продуктов.;

умения:

проводить анализ требований к программному обеспечению; формализовывать спецификацию процессов в программных продуктах автоматной моделью;

навыки:

разработки тестов и осуществления процесса тестирования..

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО

Дисциплина **ТЕСТИРОВАНИЕ И ВЕРИФИКАЦИЯ ПРОГРАММНЫХ ПРОДУКТОВ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *09.03.04 Программная инженерия*.

Содержание дисциплины является логическим продолжением дисциплин: **ВВЕДЕНИЕ В СПЕЦИАЛЬНОСТЬ, ВВЕДЕНИЕ В ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ**.

Содержание дисциплины является основой для освоения дисциплин: **ВЫПОЛНЕНИЕ И ЗАЩИТА ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ**.

Предварительные компетенции, сформированные у обучающегося до начала изучения дисциплины:

- ОПК-2 — Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности
- ОПК-3 — Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности
- ОПК-7 — Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой
- ПК-94 — способен к управлению информацией и данными, поиску источников информации и данных, восприятию, анализу, запоминанию и передаче информации с использованием цифровых средств, а также с помощью алгоритмов при работе с полученными из различных источников данными с целью эффективного использования полученной информации для решения задач
- УК-6 — Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 5 з.е., 180 ч.

3.1. Содержание (дидактика) дисциплины

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме		Самостоятельная работа студентов	Формируемая компетенция, %	
				ВСЕГО	Практические занятия		ОПК-6	ПСК-1/23.1
4	7	Раздел 1. Задачи тестирования, верификации и валидации. 1.1 Понятия верификации, валидации и тестирования программных продуктов. 1.2 Цели и задачи процессов верификации и валидации. 1.3 Тестирование как динамический метод верификации программных систем. Связь тестирования с валидацией и отладкой программных систем. 1.4 Методы анализа и построения моделей программных систем. 1.5 Тестирование и верификация в процессе проектирования.	11	4	4	7	20	20
4	7	Раздел 2. Основные виды программных ошибок. Виды и методы тестирования. 2.1 Классификация программных ошибок и способов их обнаружения. 2.2 Классификация видов и методов тестирования. Аксиомы тестирования по Г. Майерсу. 2.3 Структурное тестирование (тестирование «белого ящика») и функциональное тестирование (тестирование черного ящика) ящика. 2.4 Интеграционное тестирование. Системное тестирование. Нагрузочное тестирование. 2.5 Тест-требования. Классы входных и выходных данных. 2.6 Тестовое окружение. Технология документирования и отслеживания программных ошибок. Журнал ошибок. 2.7 Отладка как научный эксперимент. Анализ покрытия программного кода по ветвям и условиям.	28	14	14	14	20	20
4	7	Раздел 3. Методы автоматизации тестирования. 3.1 Ручное и автоматизированное тестирование. Регрессионное тестирование. 3.2 Методы формирования тестового окружения для автоматизации модульного, интеграционного и системного тестирования. 3.3 Автоматизация тестирования трансформационных и реагирующих программных систем.	35	14	14	21	20	20
4	7	Раздел 4. Технологии и инструментальные средства автоматизации тестирования. 4.1 Технологии автоматизации тестирования. Виды инструментальных средств автоматизации тестирования. 4.2 Инструментальные средства автоматизированного модульного тестирования. Инструментальные средства автоматизированного функционального тестирования трансформационных программ. 4.3 Системное тестирование. Запись и воспроизведение взаимодействия пользователя с реагирующей программной системой.	64	20	20	44	20	20
4	7	Раздел 5. Верификация и методы доказательства правильности программ. 5.1 Формализация требований к системе. Методы доказательства правильности программы. 5.2. Алгебры процессов. Исчисление взаимодействующих систем Р.Милнера. 5.3 Процессный граф. Эквивалентность поведения программных агентов. Свойство бисимуляции программных агентов. 5.4 Тестирование на моделях и темпоральные логики. Автоматизация проверки исполнимых моделей.	42	16	16	26	20	20
Всего за 7 семестр			180	68	68	112	100	100
Всего по дисциплине			180	68	68	112	100	100

3.2. Аудиторный практикум

№ п/п	Номер и наименование раздела дисциплины	Тема практического занятия	Объем, ауд. часов
1	Раздел 1. Задачи тестирования, верификации и валидации.	Понятия верификации, валидации и тестирования программных продуктов. Цели и задачи процессов верификации и валидации. Тестирование как динамический метод верификации программных систем. Связь тестирования с валидацией и отладкой программных систем.	2
2		Методы анализа и построения моделей программных систем. Тестирование и верификация в процессе проектирования.	2
3	Раздел 2. Основные виды программных ошибок. Виды и методы тестирования.	Классификация программных ошибок и способов их обнаружения. Классификация видов и методов тестирования. Аксиомы тестирования по Г. Майерсу. Структурное тестирование (тестирование «белого ящика») и функциональное тестирование (тестирование черного ящика) ящика.	2
4		Интеграционное тестирование. Системное тестирование. Нагрузочное тестирование. Тест-требования. Классы входных и выходных данных. Тестовое окружение. Технология документирования и отслеживания программных ошибок. Журнал ошибок.	2
5		Отладка как научный эксперимент. Анализ покрытия программного	2

		кода по ветвям и условиям.	
6		Выполнение индивидуальной практической работы 1 (ИПР-1): Освоение техники тестирования на примере программы средней сложности. Составление тестов, ручное тестирование, составление отчетов о тестировании.	8
7	Раздел 3. Методы автоматизации тестирования.	Ручное и автоматизированное тестирование. Регрессионное тестирование.	2
8		Выполнение индивидуальной практической работы 2 (ИПР-2): Разработка тестового окружения для модульного тестирования функций и классов.	4
9		Автоматизация тестирования трансформационных и реагирующих программных систем.	2
10		Выполнение индивидуальной практической работы 3 (ИПР-3): Автоматизация тестирования трансформационных и реагирующих программ на примере консольных программ.	6
11	Раздел 4. Технологии и инструментальные средства автоматизации тестирования.	Технологии автоматизации тестирования. Виды инструментальных средств автоматизации тестирования. Инструментальные средства автоматизированного модульного тестирования.	2
12		Выполнение индивидуальной практической работы 4 (ИПР-4): Освоение базовых функций инструментальных средств автоматизации модульного тестирования.	2
13		Инструментальные средства автоматизированного функционального тестирования трансформационных программ.	2
14		Выполнение индивидуальной практической работы 5 (ИПР-5): Освоение базовых возможностей инструментальных средств автоматизации функционального тестирования трансформационных программ.	4
15		Системное тестирование. Запись и воспроизведение взаимодействия пользователя с реагирующей программной системой.	2
16		Выполнение индивидуальной практической работы 6 (ИПР-6): Автоматизация тестирования программных продуктов с графическим пользовательским интерфейсом путем записи и последующего воспроизведения взаимодействия пользователя с программным продуктом.	8
17	Раздел 5. Верификация и методы доказательства правильности программ.	Формализация требований к системе. Методы доказательства правильности программы. Алгебры процессов. Исчисление взаимодействующих систем Р. Милнера.	2
18		Процессный граф. Эквивалентность поведения программных агентов. Свойство бисимуляции программных агентов.	2
19		Выполнение индивидуальной практической работы 7 (ИПР-7): Вычисление бисимуляции заданных программных агентов	4
20		Тестирование на моделях и темпоральные логики.	4
21		Автоматизация проверки исполнимых моделей.	4
Всего за 7 семестр			68

3.3. Самостоятельная работа студента (СРС)

№ п/п	Номер и наименование раздела дисциплины	Содержание учебного задания	Объем, часов
1	Раздел 1. Задачи тестирования, верификации и валидации.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	7
2	Раздел 2. Основные виды программных ошибок. Виды и методы тестирования.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	8
3		Подготовка к практическим занятиям	3
4		Оформление отчета по ПР-1	3
5	Раздел 3. Методы автоматизации тестирования.	Изучение предусмотренных программой дидактических единиц по рекомендуемой	7

		литературе	
6		Подготовка к практическим занятиям	8
7		Оформление отчета по ПР-2	3
8		Оформление отчета по ПР-3	3
9	Раздел 4. Технологии и инструментальные средства автоматизации тестирования.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	10
10		Подготовка к практическим занятиям	26
11		Оформление отчета по ПР-5	2
12		Оформление отчета по ПР-5	2
13		Оформление отчета по ПР-6	4
14	Раздел 5. Верификация и методы доказательства правильности программ.	Подготовка к практическим занятиям	3
15		Оформление отчета по ПР-7	3
16		Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе.	20
Всего за 7 семестр			112

4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

СЕМЕСТР	НЕДЕЛИ СЕМЕСТРА																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
7					Отч. по ПЗ	ДР	Отч. по ПЗ		Отч. по ПЗ	ДР	Отч. по ПЗ			Отч. по ПЗ		ДР	Тест, диф. зач.

Условные обозначения:

- ДР – диагностическая работа;
- Отч. по ПЗ – отчет по практическому заданию;
- Тест – тест;
- диф. зач. – дифференцированный зачет.

Текущий контроль успеваемости студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- отчет по практическому заданию;
- тест.

Промежуточная аттестация проводится в формах:

- дифференцированный зачет.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Основная литература по дисциплине:

1. . Управление программными проектами. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2013, эл. рес.
2. В. Н. Каминский. . Веб-программирование. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014, эл. рес.
3. Г. С. Иванова, Т. Н. Ничушкина, Е. К. Пугачёв. . Методы обработки данных и оценки программ. М.: Изд-во МГТУ им. Н. Э. Баумана, 2020, эл. рес.
4. Е. М. Лаврищева. . Программная инженерия и технологии программирования сложных систем. Москва: Юрайт, 2022, эл. рес.
5. И. С. Петухов. . Разработка программного обеспечения. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007, эл. рес.
6. Л. Кристин, Дж. Грегори. . Гибкое тестирование. М.: Вильямс, 2010, 5 экз.
7. Н. Н. Смирнова. . Верификация и тестирование программных систем. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014, эл. рес.
8. Н. Н. Смирнова, С. Д. Тарасов. . Основы построения компиляторов. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007, эл. рес.
9. С. В. Сеницын, Н. Ю. Налютин. . Верификация программного обеспечения. М.: Интернет-Университет Информационных Технологий, 2007, эл. рес.
10. С. Л. Романов. . Утилиты обработки текста в операционной системе Linux. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2013, эл. рес.
11. С. М. Старолетов. . Основы тестирования и верификации программного обеспечения. Санкт-Петербург: Лань, 2020, эл. рес.
12. Ю. Г. Карпов. . Model Checking. Верификация параллельных и распределённых программных систем. СПб.: БХВ-Петербург, 2010, 7 экз.

5.2. Дополнительная литература по дисциплине:

не требуется.

5.3. Периодические издания:

1. Прикладная информатика.

5.4. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины, электронные библиотечные системы:

1. <https://e.lanbook.com/> — ЭБС Лань;
2. <https://ibooks.ru/> — ЭБС Айбукс.ру - это большой выбор актуальной литературы для вашей библиотеки в электронном виде;
3. <https://urait.ru/> — Образовательная платформа «Юрайт». Для вузов и ссузов.;
4. http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=474 — Электронные ресурсы — Фундаментальная библиотека БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова;
5. <https://techbase.kde.org/Development/Tutorials/> – Руководства по разработке приложений для KDE.;
6. <https://robotframework.org/> – Generic open source automation framework for acceptance testing, acceptance test-driven development (ATTD) and robotic process automation (RPA).;
7. <http://www.emis.de/ELibM.html> – Electronic Library of Mathematics.;
8. <https://frama-c.com/> – Extensible and collaborative Platform dedicated to source code analysis of C software.;
9. <https://github.com/srki/SpinRoot/> – Система верификации Spin;
10. <https://www.unitesk.ru/> – Унифицированное решение для промышленного тестирования и обеспечения качества UniTESK.

Современные профессиональные базы данных:

1. <https://rusneb.ru> – Национальная электронная библиотека (НЭБ);
2. <https://cyberleninka.ru/> - Научная электронная библиотека «Киберленинка»;
<http://www.rfbr.ru/rffi/ru/library> - Полнотекстовая электронная библиотека Российского фонда фундаментальных исследований.

Информационные справочные системы:

1. Техэксперт – Информационный портал технического регулирования: Нормы, правила, стандарты РФ;
2. http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=457 - БД ГОСТов собственной генерации БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова;
3. <http://www.consultant.ru/> - КонсультантПлюс- информационный портал правовой информации.

5.5. Программное обеспечение:

1. Code::Blocks;
2. Kubuntu 18.04 LTS;
3. Офисный пакет Libre Office;
4. Интегрированная среда разработки Eclipse IDE;
5. Набор средств трансляции, компоновки, отладки и выполнения Python 3.x с интегрированной средой разработки IDLE;
6. Интернет-браузер Chromium;
7. Графическая консоль Konsole;
8. Набор средств верификации Robot Framework;
9. Набор средств верификации Spin.

5.6. Информационные технологии:

взаимодействие с обучающимися посредством ЭИОС Moodle БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Практические занятия:

1. Code::Blocks;
2. Kubuntu 18.04 LTS;
3. Офисный пакет Libre Office;
4. Интегрированная среда разработки Eclipse IDE;
5. Набор средств трансляции, компоновки, отладки и выполнения Python 3.x с интегрированной средой разработки IDLE;
6. Интернет-браузер Chromium;
7. Графическая консоль Konsole;
8. Набор средств верификации Robot Framework;
9. Набор средств верификации Spin.

6.2. Прочее:

1. рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
2. рабочие места студентов, оснащенные компьютерами с доступом в Интернет, предназначенные для работы в электронной образовательной среде.

Аннотация рабочей программы

Дисциплина **ТЕСТИРОВАНИЕ И ВЕРИФИКАЦИЯ ПРОГРАММНЫХ ПРОДУКТОВ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *09.03.04 Программная инженерия*. Дисциплина реализуется на факультете О Естественнотехнический БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова кафедрой О7 Информационные системы и программная инженерия.

Дисциплина нацелена на формирование *компетенций*:

ОПК-6 способность разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов;

ПСК-1/23.1 способность использовать методы и инструментальные средства исследования объектов профессиональной деятельности на всех этапах жизненного цикла программных средств.

Содержание дисциплины охватывает круг вопросов, связанных с тестированием программных продуктов, технологиями проведения тестирования, организацией процесса тестирования, автоматизацией тестирования, а также с формальной верификацией как с технологией доказательства правильности работы программных продуктов. Рассматриваются математические основы формальной верификации, в частности исчисление взаимодействующих процессов, понятия эквивалентности поведения процессов, свойства бисимуляции и применение этих формализмов на конкретных примерах.

Программой дисциплины предусмотрены следующие **виды контроля**:

Текущий контроль успеваемости студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- отчет по практическому заданию;
- тест.

Промежуточная аттестация проводится в формах:

- дифференцированный зачет.

Общая трудоемкость освоения дисциплины составляет 5 з.е., **180 ч**. Программой дисциплины предусмотрены практические занятия (**68 ч**), самостоятельная работа студента (**112 ч**).

ТЕХНОЛОГИИ И ФОРМЫ ОБУЧЕНИЯ

Рекомендации по освоению дисциплины для студента

Трудоемкость освоения дисциплины составляет 180 ч., из них 68 ч. аудиторных занятий, и 112 ч., отведенных на самостоятельную работу студента.

Рекомендации по распределению учебного времени по видам самостоятельной работы и разделам дисциплины приведены в таблице.

Контроль освоения дисциплины производится в соответствии с Положением о текущем, рубежном контроле успеваемости и промежуточной аттестации обучающихся.

Формы контроля и критерии оценивания приведены в приложении 3 к Рабочей программе.

Наименование работы	Рекомендуемая литература	Трудоемкость, час.
Раздел 1. Задачи тестирования, верификации и валидации.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Н. Н. Смирнова. . Верификация и тестирование программных систем: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1.1, 2.1, 2.2) С. М. Старолетов. . Основы тестирования и верификации программного обеспечения: Санкт-Петербург: Лань, 2020 (1.1) Е. М. Лаврищева. . Программная инженерия и технологии программирования сложных систем: Москва: Юрайт, 2022 (2, 10) И. С. Петухов. . Разработка программного обеспечения: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007 (1.2) С. В. Сеницын, Н. Ю. Налютин. . Верификация программного обеспечения: М.: Интернет-Университет Информационных Технологий, 2007 (1.1-1.3, 1.7, 1.9)	7
Итого по разделу 1		7
Раздел 2. Основные виды программных ошибок. Виды и методы тестирования.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Г. С. Иванова, Т. Н. Ничушкина, Е. К. Пугачёв. . Методы обработки данных и оценки программ: М.: Изд-во МГТУ им. Н. Э. Баумана, 2020 (Тестирование программного обеспечения, Ручное тестирование программных продуктов, Тестирование по принципу «белого ящика», Тестирование по принципу «черного ящика», Общая стратегия тестирования) С. М. Старолетов. . Основы тестирования и верификации программного обеспечения: Санкт-Петербург: Лань, 2020 (1.3, 1.4) Е. М. Лаврищева. . Программная инженерия и технологии программирования сложных систем: Москва: Юрайт, 2022 (10)	8
Подготовка к практическим занятиям	Н. Н. Смирнова. . Верификация и тестирование программных систем: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (2.1-2.6) С. В. Сеницын, Н. Ю. Налютин. . Верификация программного обеспечения: М.: Интернет-Университет Информационных Технологий, 2007 (1.9, 2.1, 2.2, 3.1, 4.1, 6.3, 7.1, 7.2)	3
Оформление отчета по ПР-1		3
Итого по разделу 2		14
Раздел 3. Методы автоматизации тестирования.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Н. Н. Смирнова, С. Д. Тарасов. . Основы построения компиляторов: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2007 (2, 3) Н. Н. Смирнова. . Верификация и тестирование программных систем: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (2.4, 2.5, 2.7) . Управление программными проектами: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2013 (Материалы лабораторной работы №4) В. Н. Каминский. . Веб-программирование: СПб.БГТУ	7
Подготовка к практическим занятиям		8

занятиям	"ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (регулярные выражения)	
Оформление отчета по ПР-2	С. Л. Романов. . Утилиты обработки текста в операционной системе Linux: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2013 (2, 3, 5)	3
Оформление отчета по ПР-3	Л. Криспин, Дж. Грегори. . Гибкое тестирование: М.: Вильямс, 2010 (7, 8) С. В. Сеницын, Н. Ю. Налютин. . Верификация программного обеспечения: М.: Интернет-Университет Информационных Технологий, 2007 (2.3, 5.1-5.3, 7.2)	3
Итого по разделу 3		21
Раздел 4. Технологии и инструментальные средства автоматизации тестирования.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	С. В. Сеницын, Н. Ю. Налютин. . Верификация программного обеспечения: М.: Интернет-Университет Информационных Технологий, 2007 (2.3, 5.1-5.3, 7.2, 12.1, 12.2, 13.1, 13.2) Л. Криспин, Дж. Грегори. . Гибкое тестирование: М.: Вильямс, 2010 (7-9, 13, 14) Н. Н. Смирнова. . Верификация и тестирование программных систем: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (2.4, 2.5, 2.7)	10
Подготовка к практическим занятиям		26
Оформление отчета по ПР-5		2
Оформление отчета по ПР-5		2
Оформление отчета по ПР-6		4
Итого по разделу 4		44
Раздел 5. Верификация и методы доказательства правильности программ.		
Подготовка к практическим занятиям	Е. М. Лаврищева. . Программная инженерия и технологии программирования сложных систем: Москва: Юрайт, 2022 (6) Н. Н. Смирнова. . Верификация и тестирование программных систем: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2014 (1.1-1.10) Ю. Г. Карпов. . Model Checking. Верификация параллельных и распределённых программных систем: СПб.: БХВ-Петербург, 2010 (1-6, 8-12) С. М. Старолетов. . Основы тестирования и верификации программного обеспечения: Санкт-Петербург: Лань, 2020 (2.1)	3
Оформление отчета по ПР-7		3
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе.		20
Итого по разделу 5		26

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств, позволяющие оценить результаты обучения по данной дисциплине, включают в себя:

- диагностическая работа
- тест;
- отчет по практическому заданию;
- дифференцированный зачет.

Критерии оценивания

Диагностическая работа

Диагностическая работа проводится в форме теста в ЭИОС Moodle:

- при правильном ответе менее чем на 60% вопросов - не аттестация;
- при правильном ответе на 60% вопросов и более - аттестация.

Тест

Итоговый тест из 10 вопросов, включающий вопросы по всем разделам дисциплины проводится на последней неделе семестра. Итоговый тест считается успешно пройденным при выполнении с рейтингом не менее 70% (получении не менее 70 баллов из 100).

Тестовые вопросы размещены в ЭИОС Moodle.

Отчет по практическому заданию

Перечень индивидуальных практических работ:

1. Освоение техники тестирования на примере программы средней сложности. Составление тестов, ручное тестирование, составление отчетов о тестировании.
2. Разработка тестового окружения для модульного тестирования функций и классов.
3. Автоматизация тестирования трансформационных и реагирующих программ на примере консольных программ с разработкой средств автоматизации процесса тестирования и анализа результатов тестирования.
4. Освоение базовых функций инструментальных средств автоматизации модульного тестирования.
5. Освоение базовых возможностей инструментальных средств автоматизации функционального тестирования трансформационных программ.
6. Автоматизация тестирования программных продуктов с графическим пользовательским интерфейсом путем записи и последующего воспроизведения взаимодействия пользователя с программным продуктом, используя специализированные инструментальные средства.
7. Вычисление бисимуляции заданных программных агентов.

Требования к выполнению ИПР:

По всем ИПР необходимо успешное выполнение требования общей и вариативной части задания, включая предъявление в работе самостоятельно написанных соответствующих программ, если это предусмотрено заданием.

Отчет по ИПР представляется в электронной форме (PDF), а при разработке программ в рамках конкретной работы — они представляются в виде исходных текстов, непосредственно пригодных к трансляции и последующему выполнению.

Приеме отчета предусматривает обсуждение порядка решения предусмотренных тематикой работы задач, включая проверку усвоения студентом соответствующих сведений из теории и степени самостоятельности при выполнении работы.

Варианты индивидуальных практических работ включены в состав УМК дисциплины.

Дифференцированный зачет

Дифференцированный зачет оформляется на 17-й неделе семестра по результатам успешной сдачи отчетов по ИПР-1, ИПР-2, ИПР-3, ИПР-4, ИПР-5, ИПР-6, ИПР-7 и прохождения итогового тестирования. Для получения оценки «зачтено-удовлетворительно» необходимо пройти итоговое тестирование с рейтингом не менее 70%, но менее 80%. Для получения оценки «зачтено-хорошо» необходимо пройти итоговое тестирование с рейтингом 80% и выше, но менее 90%. Для получения оценки «зачтено-отлично» необходимо пройти итоговое тестирование с рейтингом 90% и выше.

Паспорт фонда оценочных средств

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме		Самостоятельная работа студентов	Формируемая компетенция, %		НАИМЕНОВАНИЕ ОЦЕНОЧНОГО СРЕДСТВА
				ВСЕГО	Практические занятия		ОПК-6	ПСК-1/23.1	
4	7	Раздел 1. Задачи тестирования, верификации и валидации.	11	4	4	7	20	20	Тест
4	7	Раздел 2. Основные виды программных ошибок. Виды и методы тестирования.	28	14	14	14	20	20	Тест, Отчет по практическому заданию
4	7	Раздел 3. Методы автоматизации тестирования.	35	14	14	21	20	20	Тест, Отчет по практическому заданию
4	7	Раздел 4. Технологии и инструментальные средства автоматизации тестирования.	64	20	20	44	20	20	Тест, Отчет по практическому заданию
4	7	Раздел 5. Верификация и методы доказательства правильности программ.	42	16	16	26	20	20	Тест, Отчет по практическому заданию
Всего за 7 семестр			180	68	68	112	100	100	
Всего по дисциплине			180	68	68	112	100	100	

Критерии оценивания

ОПК-6

Вопросы открытого типа:

- № 1 Проверка корректности работы отдельных компонентов системы, выполнения ими своих функций и предполагаемых проектом характеристик – это _____ или компонентное тестирование
- № 2 Проверка корректности взаимодействий внутри отдельных групп компонентов - это _____ тестирование
- № 3 Проверка работы системы в целом, выполнения ею своих основных функций, с использованием определенных ресурсов, в окружении с заданными характеристиками – это _____ тестирование
- № 4 Тестирование отдельно взятых модулей, функций или методов класса в тестовой среде, включающей заглушки и драйверы для всех интерфейсов тестируемого модуля – это _____ тестирование
- № 5 Принцип тестирования, при котором известны функции программы, исследуется работа каждой функции на области определения, внутренняя логическая структура программы игнорируется, исследуется только внешнее ее поведение – тестирование _____ ящика
- № 6 Частный случай пользовательского тестирования, который выполняется самими разработчиками, но в среде, максимально приближенной к рабочему окружению системы и на наиболее вероятных сценариях ее реального использования
- № 7 Частный случай пользовательского тестирования, который выполняется пользователями, желающими познакомиться с возможностями системы до ее официального выпуска и передачи в эксплуатацию
- № 8 Вид динамических методов верификации, при котором проверяемое ПО выполняется в рамках заранее подготовленных сценариев
- № 9 При тестировании _____ ящика исходный код доступен для тестирования частично, либо известно что-то про внутреннюю систему
- № 10 При тестировании _____ ящика исходный код системы недоступен, производится проверка соответствия системы формальной спецификации или ожидаемому поведению

Вопросы закрытого типа:

- № 1 Непредвиденные конечные состояния, возникающие вследствие некорректности или неполноты логического описания протокольных объектов, как правило носят названия:

 тупиков (deadlocks)

 активных тупиков (livelocks)

 гонок

 терминалов
- № 2 Если все циклы активного ожидания протокола зациклились навсегда, то о таком процессе говорят, что он:

 демонизировался

 вошел в активный тупик (livelocks)

 попал в тупик (deadlocks)
- № 3 Может ли тестирование как динамический метод верификации доказать наличие у системы какого-либо свойства:

 Да

 Нет

 Неизвестно

 Иногда

№ 4	<p>Основным достоинством экспертных методов верификации является:</p> <p>линейная зависимость убывания трудоемкости при увеличении числа исполнителей на проект фиксированного объема</p> <p>достоверность и надежность</p> <p>возможность автоматического выполнения</p>
№ 5	<p>применимость к любым свойствам ПО и любым артефактам жизненного цикла на любом этапе проекта (с использованием соответствующих видов экспертизы)</p> <p>Основным недостатком экспертных методов верификации является:</p> <p>необходимость использования самих артефактов жизненного цикла ПО, а не их моделей или результатов работ</p> <p>невозможность автоматизации и необходимость активного участия людей в проведении экспертизы</p> <p>отсутствие методов экспертизы, предполагающих участие единственного исполнителя</p> <p>падение производительности сотрудников из-за отвлечения на участие в проведении экспертиз</p>
№ 6	<p>Тестирование на основе моделей является методом верификации следующего вида:</p> <p>Синтетическим</p> <p>Формальным</p> <p>Динамическим</p> <p>Статическим</p>
№ 7	<p>Цель формальных методов верификации —</p> <p>установить правила описания спецификации для работы экспертов</p> <p>показать некорректность системы</p> <p>дать формальное математическое описание системы</p>
№ 8	<p>доказать корректность системы математически строго</p> <p>Желательное свойство теста «один тест должен выявлять как можно больше ошибок» называется:</p> <p>Детективность</p> <p>Воспроизводимость</p> <p>Покрывающая способность</p>
№ 9	<p>Устойчивость</p> <p>Идея назначения классов эквивалентности входных/выходных данных для функционального тестирования основана на предположении, что:</p> <p>любая программа на всем классе ведет себя также, как и на одном его представителе</p> <p>программа на всем классе ведет себя также, как и на одном его представителе</p> <p>при функциональном тестировании известен только интерфейс тестируемой программы</p>

любую программу можно рассматривать с точки зрения объектно-ориентированной парадигмы программирования

Спецификация протокола включает в себя:

сервисы, определяемые протоколом; допущения о среде, в которой функционирует протокол; словарь сообщений, используемых для реализации протокола; формат сообщений словаря; процедурные правила, задающие ту или иную последовательность обмена сообщениями

сервисы, определяемые протоколом; словарь сообщений, используемых для реализации протокола; формат сообщений словаря; процедурные правила, задающие ту или иную последовательность обмена сообщениями

сервисы, определяемые протоколом; допущения о среде, в которой функционирует протокол; словарь сообщений, используемых для реализации протокола; формат сообщений словаря; процедурные правила, задающие ту или иную последовательность обмена сообщениями; приложения, обслуживаемые протоколом

сервисы, определяемые протоколом; допущения о среде, в которой функционирует протокол; словарь сообщений, используемых для реализации протокола; процедурные правила, задающие ту или иную последовательность обмена сообщениями

ПСК-1/23.1

Вопросы открытого типа:

- № 1 В алгебре процессов отношение эквивалентности, удовлетворяющее условию: системы, связанные отношением эквивалентности, могут быть заменены в составе объемлющей системы без изменения поведения последней, называется _____
- № 2 Поиск типичных ошибок и дефектов в артефактах на основе некоторых шаблонов относится к _____ методам верификации
- № 3 В основе методов автоматической статической верификации артефактов, представленных в текстовой форме, лежит использование _____ грамматик
- № 4 Для возможность однозначной проверки артефактов, построенных с использованием графических языков проектирования, их внутреннее представление должно предоставлять соответствие элементарного графического символа символу алфавита некоторого _____ языка
- № 5 В случае, если необходимо учесть риски (дефицит специалистов, непрекращающийся поток изменений, нехватка информации о внешних компонентах, определяющих окружение системы или вовлеченных в интеграцию и т.п.), выбирают _____ модель разработки программного обеспечения
- № 6 Каждая стадия _____ модели заканчивается получением некоторых результатов, которые служат в качестве исходных данных для следующей стадии
- № 7 Системы FoxPro и MS Access характеризуются низкими требованиями к вычислительным ресурсам сервера и относятся к системам с _____ архитектурой
- № 8 Процесс доказательства корректности специально созданной модели программы называется _____
- № 9 Часть процесса разработки, в котором проверяется соответствие программы ее спецификации называется _____
- № 10 Процессный _____ – это именованная система переходов с выделенным начальным состоянием

Вопросы закрытого типа:

- № 1 Вид динамических методов верификации, в рамках которого идет только наблюдение, запись и оценка результатов работы ПО при его обычном использовании:

Наблюдение

Тестирование

Мониторинг

№ 2	<p>Отладка</p> <p>Группа методов мониторинга и профилирования, при которых не производится какого-либо вмешательства в саму верифицируемую систему, все измерения и оценки каких-либо параметров производятся только стандартным и внешними средствами целевой внешней среды, например, оценкой со стороны операционной системы выделенных ресурсов, времени исполнения, частоты и длительности интервалов задействия вычислительных блоков, частоты обращения к тем или иным функциям:</p> <p>мониторинг и профилирование средствами самой верифицируемой системы</p> <p>мониторинг и профилирование путем инструментирования</p> <p>мониторинг и профилирование со стороны внешней среды</p>
№ 3	<p>симуляторный мониторинг и профилирование</p> <p>Группа методов мониторинга и профилирования, при которых тем или иным способом изменяется сама верифицируемая система для возможности взаимодействия верифицируемой программной системы с системой мониторинга:</p> <p>мониторинг и профилирование путем инструментирования</p> <p>мониторинг и профилирование со стороны внешней среды</p> <p>симуляторный мониторинг и профилирование</p>
№ 4	<p>мониторинг и профилирование средствами самой верифицируемой системы</p> <p>Группа методов мониторинга и профилирования, при которых используется объемлющая система, отличающаяся от целевой объемлющей системы для возможности взаимодействия верифицируемой программной системы с системой мониторинга:</p> <p>мониторинг и профилирование со стороны внешней среды</p> <p>мониторинг и профилирование средствами самой верифицируемой системы</p> <p>симуляторный мониторинг и профилирование</p>
№ 5	<p>мониторинг и профилирование путем инструментирования</p> <p>Группа методов мониторинга и профилирования, при которых используются соответствующие средства, являющиеся частью самой верифицируемой системы при её нормальном (обычном) функционировании, например, учет объемов переданных и принятых данных, задействованных ресурсов памяти, скорости обработки информации:</p> <p>мониторинг и профилирование средствами самой верифицируемой системы</p> <p>мониторинг и профилирование со стороны внешней среды</p> <p>мониторинг и профилирование путем инструментирования</p>
№ 6	<p>симуляторный мониторинг и профилирование</p> <p>Основные классы алгоритмов для вычисления сильной бисимуляции двух агентов:</p> <p>класс глобальных алгоритмов и класс алгоритмов «на лету»</p> <p>класс локальных алгоритмов и класс алгоритмов «на лету»</p> <p>класс глобальных алгоритмов и класс локальных алгоритмов</p>
№ 7	<p>класс локальных алгоритмов и класс алгоритмов последовательных приближений</p> <p>Может ли эквивалентность по путям всегда служить эквивалентностью поведения при описании систем в CCS?</p>

- Нет, по причине возможного недетерминизма при совпадении множеств строгих путей начальных состояний
- Нет, по причине отсутствия формальной связи между этими понятиями
- Нет, согласно аксиомам
- № 8 Да
Что должно достигаться для обеспечения полного функционального покрытия пользовательского интерфейса?
- все требования к пользовательскому интерфейсу должны быть покрыты
- каждый интерфейсный элемент должен быть использован хотя бы один раз
- каждый интерфейсный элемент должен побывать во всех состояниях
- каждый интерфейсный элемент должен побывать во всех состояниях в различных режимах работы системы
- № 9 Контроль того, что все формализованные правила корректности построения артефактов выполнены относится к следующим методам верификации:
- Статическим
- Статистическим
- Экзистенциальным
- Динамическим
- № 10 В случае, когда необходимо определить соответствие программного обеспечения предназначению, осуществляют:
- Верификацию
- Апробацию
- Оценку качества
- Оценку эффективности