

Направление подготовки 09.03.01 «Информатика и вычислительная техника»
Профиль «Программное обеспечение средств вычислительной техники и автоматизированных систем»
РПД Б1.В.14 «Тестирование программного обеспечения»



**Филиал федерального государственного бюджетного образовательного учреждения
высшего образования
«Национальный исследовательский университет «МЭИ»
в г. Смоленске**

УТВЕРЖДАЮ

Директор филиала

ФГБОУ ВО «НИУ «МЭИ» в г. Смоленске

д-р техн. наук, профессор



А.С. Федулов

2023 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ТЕСТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

(наименование дисциплины)

Направление подготовки: **09.03.01 «Информатика и вычислительная техника»**

Профиль **«Программное обеспечение средств вычислительной техники и автоматизированных систем»**

Уровень высшего образования: **бакалавриат**

Нормативный срок обучения: **4 года**

Форма обучения: **очная**

Год набора: **2023**


Смоленск

Направление подготовки 09.03.01 «Информатика и вычислительная техника»
Профиль «Программное обеспечение средств вычислительной техники и автоматизированных систем»
РПД Б1.В.14 «Тестирование программного обеспечения»



Программа составлена с учетом ФГОС ВО – бакалавриат по направлению подготовки 09.03.01 «**Информатика и вычислительная техника**», утвержденного приказом Минобрнауки России от «19» сентября 2017 г. № 929.

Программу составил:

Ст. преподаватель  _____ А.И. Гаврилов _____ ФИО
подпись

«05» июня 2023 г.

Программа обсуждена и одобрена на заседании кафедры «Вычислительная техника» «07» июня 2023 г., протокол № 8

**Заведующий кафедрой вычислительной техники
д.т.н., профессор**

А.С. Федулов

«07» июня 2023 г.

РПД адаптирована для лиц с ограниченными возможностями здоровья и инвалидов

**Ответственный в филиале по работе
с ЛОВЗ и инвалидами**

Е.В. Зуева

«07» июня 2023 г.

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель освоения дисциплины (модуля): подготовка обучающихся по направлению подготовки 09.03.01 "Информатика и вычислительная техника" посредством обеспечения этапов формирования компетенций, предусмотренных ФГОС, в части представленных ниже знаний, умений и навыков.

Задачами дисциплины являются:

- ознакомление студентов с основными методиками тестирования, отладки и оценки качества разрабатываемого программного обеспечения;
- изучение студентами методик планирования тестов;
- обучение студентов методам организации контроля над процессом тестирования;
- ознакомление студентов с программными средствами автоматизации процесса тестирования;
- изучение методов документирования процесса тестирования.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО

Дисциплина «Тестирование программного обеспечения» относится к *части программы, формируемой участниками образовательных отношений*

Для изучения данной дисциплины необходимы знания, умения и навыки, формируемые предшествующими дисциплинами:

- Б1.В.15 Объектно-ориентированные технологии
- Б1.В.ДВ.05.01 Трансляторы программных языков
- Б1.В.ДВ.05.02 Теория формальных грамматик
- Б2.В.02(П) Проектно-технологическая практика

Перечень последующих дисциплин, для которых необходимы знания, умения и навыки, формируемые данной дисциплиной:

- Б1.В.20 Основные парадигмы программирования
- Б2.В.04(Пд) Преддипломная практика
- Б3.01 Подготовка к защите и защита выпускной квалификационной работы

3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Освоение дисциплины направлено на формирование элементов следующих компетенций в соответствии с ФГОС ВО и ОП ВО по данному направлению подготовки:

Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций

Компетенция	Индикаторы достижения компетенций	Результаты обучения
ПК-8 Способен выполнять работы по проекти-	ПК-8.1 Выполняет работы по проектированию компонентов системного программного обеспечения	Знает: - методы тестирования программных продуктов, в т.ч. системного программного обеспечения;

<p>рованию, интеграции и тестированию компонентов системного программного обеспечения</p>		<ul style="list-style-type: none"> - методы разработки эффективных наборов тестов для простых и крупных информационных систем; - методы проектирования автоматизированных программных тестов. <p>Умеет:</p> <ul style="list-style-type: none"> - устанавливать, тестировать, интегрировать, испытывать и использовать программные средства; - проектировать программные автоматизированные тесты для программного обеспечения. <p>Владеет:</p> <ul style="list-style-type: none"> - навыками использования различных методов тестирования программного обеспечения в приложении к компонентам системного и прикладного программного обеспечения.
	<p>ПК-8.2 Выполняет работы по интеграции компонентов системного программного обеспечения</p>	<p>Знает:</p> <ul style="list-style-type: none"> - методы интеграционного тестирования программного обеспечения. <p>Умеет:</p> <ul style="list-style-type: none"> - строить наборы функциональных тестов для интеграционного тестирования программного обеспечения; - строить наборы тестов для тестирования качества (в т.ч. надежности) программного обеспечения. <p>Владеет:</p> <ul style="list-style-type: none"> - навыками интеграционного и системного тестирования программного обеспечения.
	<p>ПК-8.3 Выполняет работы по тестированию компонентов системного программного обеспечения</p>	<p>Знает: методы анализа и тестирования требований, теорию тестирования (модели тестирования, планирование тестирования, тест-дизайн, проектирование тестов).</p> <p>Умеет:</p> <ul style="list-style-type: none"> - определять наиболее затратные (трудоемкие) места в процессе тестирования, определять конечные данные для эксплуатации на основе разрабатываемых требований. - оценивать важность (приоритет выполнения) различных тестов (на основе приоритетов пользователя, проектных задач и рисков возникновения ошибки); - выполнять ручное и автоматизированное тестирование программного обеспечения. <p>Владеет:</p> <ul style="list-style-type: none"> - навыками тестирования исходной документации (поиск нестыковок, выяснение

		<p>недостающей информации по продукту);</p> <ul style="list-style-type: none">- навыками разработки требования к тестированию на основе требований к системе (бизнес-требований, функциональных требований, требований к производительности и др.);- навыками выбора видов тестирования и их применения по отношению к объекту тестирования;- навыками выполнения задач тестирования и анализа результатов.
--	--	---



4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Структура дисциплины:

№	Индекс	Наименование	Семестр 7											Семестр 8											Итого за курс											Каф.	Семестры				
			Контроль	Академических часов									з.е.	Неделя	Контроль	Академических часов									з.е.	Неделя															
				Всего	Контакт.	Лек	Лаб	Пр	КРП	СР	Контроль	Всего				Контакт.	Лек	Лаб	Пр	КРП	СР	Контроль	з.е.	Неделя																	
5	Б1.В.14	Тестирование программного обеспечения	Экз	180	52	18	34				92	36	5															Экз	180	52	18	34			92	36	5			15	7

ОБОЗНАЧЕНИЯ:

Виды промежуточной аттестации (виды контроля):

Экз - экзамен;

ЗаО - зачет с оценкой;

За – зачет;

Виды работ:

Контакт. – контактная работа обучающихся с преподавателем;

Лек. – лекционные занятия;

Лаб.– лабораторные работы;

Пр. – практические занятия;

КРП – курсовая работа (курсовой проект);

РГР – расчетно-графическая работа (реферат);

СР – самостоятельная работа студентов;

з.е.– объем дисциплины в зачетных единицах.

Содержание дисциплины:

№	Наименование видов занятий и тематик, содержание
1	<p>лекционные занятия 9 шт. по 2 часа:</p> <p>Тема 1. Основные понятия и подходы к тестированию программного обеспечения. Классификации. Процессы тестирования. Методы тестирования.</p> <p>Лекция 1 (2 часа). Тестирование. Основные понятия и подходы Понятия и определения. Классификация видов тестирования. История тестирования (этапы развития подходов). Процессы тестирования в жизненном цикле (ЖЦ) ПО. Тестирование дефектов. Расширенный взгляд на терминологию, описывающую проблемы. Черный и белый ящик. Нисходящее и восходящее тестирование. Функциональное тестирование методами черного ящика.</p> <p>Лекция 2 (2 часа). Структурное тестирование программного обеспечения. Место структурного тестирования в классификации видов тестирования. Тестирование «белого ящика. Способы тестирования белого ящика. Способ тестирования базового пути. Способы тестирования условий. Тестирование ветвей и операторов отношений. Способ тестирования потоков данных. Тестирование циклов. Модульное (unit) тестирование. Инструменты модульного тестирования (встроенные в среду программирования Visual Studio, D-Unit, N-Unit, прочие). Интеграционное тестирование (Integration Testing) .</p> <p>Лекция 3 (2 часа). Тестирование объектно-ориентированных систем. Тестирование классов объектов. Интеграция объектов. Особенности тестирования классов (тестирование наследования и проч.). Варианты использования, диаграммы Use Case. Применение Use Cases в процессе тестирования. Создание тестового набора на основе вариантов использования. Пример тестирования класса.</p> <p>Системное Тестирование (System Testing). Задачи и цели системного тестирования. Виды системного тестирования. Приемочное Тестирование (Acceptance Testing). Прочее.</p> <p>Тема 2. Специальные виды тестирования программного обеспечения.</p> <p>Лекция 4 (2 часа). Нагрузочное тестирование. Терминология. Основные виды тестирования производительности. Основные принципы нагрузочного тестирования. Цели нагрузочного тестирования. Этапы проведения нагрузочного тестирования. Инструменты нагрузочного тестирования (Apache Jmeter и прочие). Стресс-тестирование.</p> <p>Лекция 5 (2 часа). Тестирование пользовательского интерфейса Задачи и цели тестирования пользовательского интерфейса . Функциональное тестирование пользовательских интерфейсов. Проверка требований к пользовательскому интерфейсу. Методы проведения тестирования пользовательского интерфейса, повторяемость тестирования пользовательского интерфейса. Тестирование удобства использования пользовательских интерфейсов. Тестирование пользовательского интерфейса на основе состояний и диаграммы переходов состояний. Методика тестирования пользовательского интерфейса.</p> <p>Лекция 6 (2 часа). Тестирование Веб-приложений и мобильных приложений. Особенности веб-приложений как объекта тестирования. Подходы к функциональному тестированию Веб-приложений. Нагрузочное и стресс-тестирование веб-приложений. Инструменты тестирования веб-приложений. Тестирование мобильных приложений</p> <p>Лекция 7 (2 часа). Тестирование баз данных. Тестирование модели, целостности,</p>

	<p>связности, непротиворечивости, тестирование атрибутов, сценариев, проч. Функциональное тестирование. Нагрузочное и стресс-тестирование баз данных.</p> <p>Тема 4. Тестирование качества и управление качеством программного обеспечения</p> <p>Лекция 8 (2 часа). Тестирование надежности программного обеспечения и методы повышения надежности.</p> <p>Основные показатели надежности программного обеспечения.</p> <p>Математические модели оценки надежности ПО. Основные показатели надежности программного обеспечения. Основные понятия в проблематике надежности ПС. Классификация моделей надежности. Марковские и пуассоновские модели надежности. Модель Джелинского-Моранды. Модель Шика-Уолвертона.</p> <p>Лекция 9 (2 часа). Тестирование качества и управление качеством программного обеспечения.</p> <p>Качество программного обеспечения (Software Quality). Требования к качеству программного обеспечения. Модели и характеристики качества. Метрики, способы оценки качества. Процессы управления качеством программного обеспечения.</p>
2	<p>лабораторные работы 8 шт. по 4 часа, дополнительное аудиторное время на защиту – 2 часа:</p> <p>Лабораторная работа № 1 (4 часа). Структурное тестирование программного обеспечения. Функциональное тестирование программного обеспечения. Цель работы: Изучение методов тестирования программного обеспечения. Задачи: Выполнение ручного тестирования программ. Выполнение структурного тестирования программ следующими методами: тестирование базового пути; тестирование условий; тестирование циклов; тестирование потоков данных. Выполнение функционального тестирования программ следующими методами: разбиение на классы эквивалентности; анализ граничных значений; анализ причинно-следственных связей; предположение об ошибке.</p> <p>Лабораторная работа № 2 (4 часа). Изучение инструментов автоматизированного модульного (Unit) тестирования. Цель: получение начальных навыков применения инструментария автоматизированного тестирования программного кода методом белого ящика. Задачи: Изучить средства автоматизированного тестирования, встроенные в MS Visual Studio, отдельные кроссплатформенные инструменты. Подготовить план тестирования методами белого ящика (по вариантам) и создать тестовый проект для заданной программы. Получить и проанализировать результаты тестирования.</p> <p>Лабораторная работа №3 (4 часа). Тестирование объектно-ориентированного программного обеспечения. Цель: изучение методов тестирования объектно-ориентированного программного обеспечения, получение навыков применения инструментов автоматизированного тестирования средствами MS Visual Studio (программирование тестового проекта). Задачи: Разработать объектно-ориентированное приложение. Предусмотреть создание интерфейсов и не менее трех уровней иерархии классов. Выполнить объектно-ориентированное тестирование разработанного приложения: тестирование методов классов (Unit –тестирование методом белого ящика), тестирование наследования, составление тест-кейсов для функционального тестирования методом черного ящика на основе диаграмм UseCases.</p> <p>Лабораторная работа № 4 (4 часа)</p> <p>Разработка и тестирование web-сайта. Функциональное тестирование и отладка разработанного приложения с использованием стратегий тестирования интернет приложе-</p>

	<p>ний. Цель: получить навык функционального тестирования веб-приложений. Задачи: согласовать техническое задание на разработку сайта, выполнить реализацию сайта, установить на локальный веб-сервер APACHE. Составить план тестирования, выполнить функциональное тестирование методами черного ящика.</p> <p>Лабораторная работа №5 (4 часа). Тестирование производительности web – сайтов. Цель: получить навык тестирования производительности веб-сайтов с помощью инструментов автоматизированного тестирования (APACHE JMETER, онлайн-сервисы). Задачи: Нагрузочное тестирование (load testing), сбор показателей и определение производительности и времени отклика веб-сайта в ответ на генерируемые внешние запросы (нагрузку); исследование времени отклика системы на высоких или пиковых нагрузках (стресс-тестирование).</p> <p>Лабораторная работа №6 (4 часа). Разработка и тестирование базы данных. Цель: получить навык тестирования баз данных. Задачи: Согласовать техническое задание на разработку базы данных, реализовать БД в СУБД mysql. Выполнить тестирование БД на основе методов, изложенных в лекции, за исключением нагрузочного тестирования.</p> <p>Лабораторная работа №7 (4 часа). Тестирование производительности базы данных с помощью APACHE JMETER. Цель: получить навык тестирования производительности баз данных с помощью инструментов автоматизированного тестирования. Задачи: сделать замеры времени выполнения операций при определенных, при максимальных, минимальных и средних нагрузках; определить количество пользователей, которые одновременно могут работать с продуктом; определить границы приемлемой производительности при увеличении нагрузки; исследовать производительность при больших, максимальных и стрессовых нагрузках.</p> <p>Лабораторная работа № 8 (4 часа). Тестирование документации. Цель: получить навык разработки и тестирования документации в процессах тестирования программного обеспечения. Задачи: подготовить пакет документации для разработанных ранее веб-сайта и базы данных, включая: технические задания, технические описания, программу и методику испытаний, протокол испытаний и отчет об испытаниях. Разработать программу (план) тестирования документации на соответствие комплектности, содержанию и прочим критериям, рассмотренным в лекции. Дополнить программу и методику испытаний соответствующим разделом; результаты работы внести в отчет и протокол.</p> <p>Дополнительное аудиторное время для защит лабораторных работ – 2 часа.</p>
3	
4	<p>Самостоятельная работа студентов:</p> <ol style="list-style-type: none"> 1. Самостоятельное изучение материала: <ul style="list-style-type: none"> • К теме 1: Подробное рассмотрение классификаций видов тестирования: Классификация по запуску кода на исполнение. Классификация по доступу к коду и архитектуре приложения. Классификация по степени автоматизации. Классификация по уровню детализации приложения. Классификация по принципам работы с приложением. Классификация по природе приложения. Классификация по фокусировке на уровне архитектуры приложения. Классификация по привлечению конечных пользователей. Классификация по степени формализации . Классификация по целям и задачам. Классификация по техникам и подходам. Классификация по моменту выполнения (хронологии).

<ul style="list-style-type: none">• К теме 1: Регрессионное тестирование программного обеспечения. Цели и задачи регрессионного тестирования. Виды регрессионного тестирования. Управляемое регрессионное тестирование. Обоснование корректности метода отбора тестов. Классификация тестов при отборе. Возможности повторного использования тестов. Классификация выборочных методов.• К теме 1: Основные понятия отладки программного обеспечения. Принципы отладки ПО. Методы отладки программного обеспечения. Виды отладки программного обеспечения. Автономная отладка программного обеспечения. Комплексная отладка программного обеспечения. Методы отладки параллельных программ. Анализ инструментов отладки параллельных программ.• К теме 4: Модели тестирования надежности ПО, основанные на методе "посева" и разметки ошибок, и модели на основе учета структуры входных данных. Модель Нельсона. Применение последовательного анализа Вальда для снижения количества прогонов программы.• К теме 4: Методы повышения надежности программ за счет резервирования и оценка эффективности их применения. Влияние избыточности на повышение надежности программ. Эффективность применения избыточности для повышения надежности комплексов программ. Влияние оперативного контроля и восстановления на производительность. Методы программного восстановления. Методы обеспечения надежности комплексов программ при сопровождении. <ol style="list-style-type: none">2. Подготовка к выполнению и защите лабораторных работ (5 работ).3. Подготовка к экзамену по дисциплине с использованием оценочных материалов, приведенных в разделе 6 настоящей РПД
--

Текущий контроль:

- проверка дополнительных теоретических материалов;
- проверка отчетов по лабораторным работам и защита лабораторных работ.

Результаты текущего контроля фиксируются с использованием трехбалльной системы (0, 1, 2) при проведении контрольных недель по графику филиала в течение семестра, а также учитываются преподавателем при осуществлении промежуточной аттестации по настоящей дисциплине.

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Таблица - Образовательные технологии, используемые при реализации различных видов учебной занятий по дисциплине

№ п/п	Виды учебных занятий	Образовательные технологии
1	Лекции	Классическая (традиционная, информационная) лекция
2	Лабораторные работы	Технология выполнения лабораторных заданий: индивидуально. Технология проблемного обучения на основе анализа результатов лабораторной работы: индивидуальный опрос, представление студентом результатов лабораторной работы в форме отчета.
3	Самостоятельная работа студентов (внеаудиторная)	Информационно-коммуникационные технологии (доступ к ЭИОС филиала, к ЭБС филиала, доступ к информационно-методическим материалам по дисциплине).
4	Контроль (промежуточная аттестация: экзамен)	Технология устного опроса.

6. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ – ДЛЯ ОЦЕНКИ КАЧЕСТВА ОСВОЕНИЯ ДИСЦИПЛИНЫ

К промежуточной аттестации студентов по дисциплине могут привлекаться представители работодателей, преподаватели последующих дисциплин, заведующие кафедрами.

Оценка качества освоения дисциплины включает как текущий контроль успеваемости, так и промежуточную аттестацию.

Оценочные средства текущего контроля успеваемости:

Вопросы к экзамену:

1. Понятия и определения тестирования ПО.
2. Классификация видов тестирования ПО (общая). Черный и белый ящик, восходящее и нисходящее, тестирование функциональное и тестирование качества.
3. Классификация видов тестирования по запуску кода на исполнение. Классификация по доступу к коду и архитектуре приложения. Классификация по степени автоматизации.
4. Классификация видов тестирования по уровню детализации приложения. Классификация по принципам работы с приложением. Классификация по природе приложения.
5. Классификация видов тестирования по фокусировке на уровне архитектуры приложения. Классификация по привлечению конечных пользователей. Классификация по степени формализации.
6. Классификация видов тестирования по целям и задачам. Классификация по техникам и подходам. Классификация по моменту выполнения (хронологии). Тестирование дефектов. Виды дефектов. Подходы к тестированию.
7. История тестирования (этапы развития подходов). Процессы тестирования в жизненном цикле (ЖЦ) ПО.
8. Тестирование «белого ящика» Способы тестирования белого ящика. Способ тестирования базового пути.
9. Способы тестирования условий.
10. Тестирование ветвей и операторов отношений.
11. Способ тестирования потоков данных.
12. Тестирование циклов.
13. Тестирование объектно-ориентированных систем. Тестирование классов объектов.
14. Интеграция объектов. Варианты использования (Use Case). Применение Use Cases в процессе тестирования ПО.
15. Создание тестового набора на основе вариантов использования. Пример тестирования классов.
16. Проектирование объектно-ориентированных тестовых вариантов.
17. Тестирование с использованием прочих диаграмм UML. Тестирование объектного ПО с использованием диаграмм состояний.
18. Системное тестирование. Задачи и цели системного тестирования. Виды системного тестирования.
19. Тестирование качества ПО. Виды тестирования качества.
20. Основные принципы нагрузочного тестирования. Цели нагрузочного тестирования. Этапы проведения нагрузочного тестирования. Стресс-тестирование.
21. Инструменты нагрузочного тестирования. Обзор инструментов. Jmeter, особенности применения.
22. Тестирование пользовательского интерфейса.
23. Задачи и цели тестирования пользовательского интерфейса. Функциональное тестирование пользовательских интерфейсов. Проверка требований к пользовательскому интерфейсу.
24. Методы проведения тестирования пользовательского интерфейса, повторяемость тестирования пользовательского интерфейса. Тестирование удобства использования пользовательских интерфейсов.

25. Тестирование пользовательского интерфейса на основе состояний и диаграммы переходов состояний. Методика тестирования пользовательского интерфейса.
26. Тестирование Веб-приложений. Общие требования.
27. Подходы к функциональному тестированию Веб-приложений.
28. Инструменты автоматизации тестирования Веб-приложений.
29. Основные показатели надежности программного обеспечения. Математические модели оценки надежности ПО.
30. Обеспечение надежности ПО. Обеспечение надежности за счет разных видов резервирования.
31. Тестирование баз данных. Общий подход.
32. Тестирование баз данных. Нагрузочное тестирование. Подходы, инструменты.
33. Отладка программного обеспечения. Понятия, общие подходы.
34. Прикладные методы отладки ПО.
35. Отладка параллельных программ.
36. Отладка мобильных приложений. Особенности тестирования и отладки мобильных приложений.
37. Инструменты отладки мобильных приложений.
38. Тестирование качества программного обеспечения. Показатели, метрики.
39. Регрессионное тестирование программного обеспечения.
40. Подготовка документации в процессе тестирования

Примеры экзаменационных задач

Задача 1. Составить потоковый граф и наборы тестовых данных для задачи нахождения корней квадратного уравнения:

Квадратное уравнение — это уравнение вида $ax^2 + bx + c = 0$, где a не равно 0. Для решения квадратного уравнения можно использовать формулы:

$$x_+ = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \text{и} \quad x_- = \frac{-b - \sqrt{b^2 - 4ac}}{2a},$$

где $D = b^2 - 4ac$ — дискриминант многочлена $ax^2 + bx + c$.

Если $D > 0$, то уравнение имеет два различных вещественных корня.

Если $D = 0$, то оба корня вещественны и равны.

Если $D < 0$, то оба корня являются комплексными числами.

Задача 2. Есть программа, которая интерпретирует три целых числа, вводимых с клавиатуры, как длины сторон треугольника и выводит сообщение, о том, какой это треугольник: равносторонний, равнобедренный или неравносторонний. Составьте потоковый граф. Напишите на листе бумаги тесты (наборы входных данных и ожидаемые результаты), которые, как вам кажется, будут адекватно проверять эту программу. Рекомендуется запись тестов в виде таблицы:

№ теста	Исходные данные			Ожидаемый результат
	a	b	c	

Задача 3. В компьютерной обучающей системе тестовые задания для контроля знаний берутся из файла типа .txt, где каждое задание занимает одну строку. Для формирования теста указывается имя файла, количество заданий в тесте и количество вариантов теста (не более 10). Варианты должны различаться не менее чем тремя заданиями. Если заданий в файле недостаточно для реализации этого требования, выдается сообщение «Недостаточно заданий», если файл не найден - «файл отсутствует».

Составьте тестовые наборы для проверки перечисленных функций. Рекомендуется оформить их в виде таблицы:

№ теста	Исходные данные			Ожидаемый результат
	a	b	c	

Задача 4. Составьте потоковый граф и тестовые наборы данных для проверки перечисленных ниже функций программы:

Программа определяет тип треугольника. Возможные результаты: прямоугольный, остроугольный, тупоугольный, равнобедренный, правильный (равносторонний)

Задача 5. Составьте потоковый граф и тестовые наборы данных для проверки перечисленных ниже функций программы:

Четырехугольник задается координатами вершин. Программа проверяет, является ли он квадратом .

Задача 6. Составьте потоковый граф и тестовые наборы данных для проверки перечисленных ниже функций программы:

Четырехугольник задается координатами вершин. Программа проверяет, является ли он ромбом.

Задача 7. Составьте потоковый граф и тестовые наборы данных для проверки перечисленных ниже функций программы:

Определение взаимного положения прямой и окружности. Прямая описывается уравнением $Y=kX+b$. Окружность с центром в начале координат задается радиусом R. Результат – линия и окружность не пересекаются, пересекаются в двух точках, прямая линия является касательной к окружности.

Задача 8. Составьте потоковый граф и тестовые наборы данных для проверки перечисленных ниже функций программы:

Определение взаимного положения двух окружностей. Окружности задаются координатами центра X,Y и радиусом R. Результат – окружности не пересекаются, пересекаются в двух точках, касаются в одной точке, совпадают.

Задача 9. Требуется выполнить структурное тестирование текста программы, которая определяет значение x в зависимости от значений параметров процедуры.

```
Procedure t (a, b:real; var x:real); begin if(a>1) and (b=0) then x:=x /a;
if(a=2) or (x>1) then x:=x+1;
end;
```

Для формирования тестов программу представляют в виде графа, вершины которого соответствуют операторам программы, а дуги представляют возможные варианты передачи управления.

Задача 10. В соответствии с концепцией максимально полного тестирования всех маршрутов программы определить независимые пути потоковых графов, представленных на рисунке:

<p>А)</p>	<p>Б)</p>	<p>В)</p>
<p>Задача 11. Для заданной процедуры составить потоковый граф и составить тестовые наборы по критерию покрытия маршрутов: procedure m(a,b: real; var x: real) begin if (a>0)and(b<0) then x:=x+1; if ((a=2)or(x>3))and(b>-10) then x:=x-1; end;</p>		
<p>Задача 12. Для заданной процедуры составить потоковый граф и составить тестовые наборы по критерию покрытия маршрутов: procedure m(a,b,c: real; var x: real) begin if (a>0)and(b<0)and(x>6) then x:=x+1; if (a=4)or(c<0) then x:=x+11; end;</p>		

В филиале используется система с традиционной шкалой оценок – "отлично", "хорошо", "удовлетворительно", "неудовлетворительно", "зачтено", "не зачтено" (далее - пятибалльная система).

Форма промежуточной аттестации по настоящей дисциплине – *экзамен*.

Применяемые критерии оценивания по дисциплинам (в соответствии с инструктивным письмом НИУ МЭИ от 14 мая 2012 года № И-23):

Оценка по дисциплине	Критерии оценки результатов обучения по дисциплине
«отлично»/ «зачтено (отлично)»/ «зачтено»	Выставляется обучающемуся, обнаружившему всестороннее, систематическое и глубокое знание материалов изученной дисциплины, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной рабочей программой дисциплины; проявившему творческие способности в понимании, изложении и использовании материалов изученной дисциплины, безупречно ответившему не только на вопросы билета, но и на дополнительные вопросы в рамках рабочей программы дисциплины, правильно выполнившему практическое задание. Оценка по дисциплине выставляется обучающемуся с учётом результатов текущего контроля. Компетенции, закреплённые за дисциплиной, сформированы на уровне – «эталонный».

Оценка по дисциплине	Критерии оценки результатов обучения по дисциплине
«хорошо»/ «зачтено (хорошо)»/ «зачтено»	Выставляется обучающемуся, обнаружившему полное знание материала изученной дисциплины, успешно выполняющему предусмотренные задания, усвоившему основную литературу, рекомендованную рабочей программой дисциплины; показавшему систематический характер знаний по дисциплине, ответившему на все вопросы билета, правильно выполнивший практическое задание, но допустивший при этом принципиальные ошибки. Оценка по дисциплине выставляется обучающемуся с учётом результатов текущего контроля. Компетенции, закреплённые за дисциплиной, сформированы на уровне – «продвинутой».
«удовлетворительно»/ «зачтено (удовлетворительно)»/ «зачтено»	Выставляется обучающемуся, обнаружившему знание материала изученной дисциплины в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющемуся с выполнением заданий, знакомому с основной литературой, рекомендованной рабочей программой дисциплины; допустившему погрешность в ответе на теоретические вопросы и/или при выполнении практических заданий, но обладающему необходимыми знаниями для их устранения под руководством преподавателя, либо неправильно выполнившего практическое задание, но по указанию преподавателя выполнившего другие практические задания из того же раздела дисциплины. Компетенции, закреплённые за дисциплиной, сформированы на уровне – «пороговой».
«неудовлетворительно»/ не зачтено	Выставляется обучающемуся, обнаружившему серьезные пробелы в знаниях основного материала изученной дисциплины, допустившему принципиальные ошибки в выполнении заданий, не ответившему на все вопросы билета и дополнительные вопросы и неправильно выполнившего практическое задание (неправильное выполнение только практического задания не является однозначной причиной для выставления оценки «неудовлетворительно»). Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение по образовательной программе без дополнительных занятий по соответствующей дисциплине. Оценка по дисциплине выставляется обучающемуся с учётом результатов текущего контроля. Компетенции на уровне «пороговой», закреплённые за дисциплиной, не сформированы.

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Учебное и учебно-лабораторное оборудование

Для проведения лекционных занятий используется учебная аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная специализированной мебелью; доской аудиторной; демонстрационным оборудованием: персональным компьютером (ноутбуком); переносным (стационарным) проектором.

Для проведения занятий лабораторного типа используется учебная аудитория для лабо-

раторных работ, выполняемых в компьютерном классе, оснащенная специализированной мебелью; доской аудиторной; персональными компьютерами, связанными локальной вычислительной сетью с подключением к сети Интернет и доступом в ЭИОС филиала, укомплектованная стойкой с активным сетевым оборудованием (коммутаторами и маршрутизаторами).

Для самостоятельной работы обучающихся по дисциплине используется помещение для самостоятельной работы обучающихся, оснащенное специализированной мебелью; доской аудиторной; персональным компьютерами с подключением к сети Интернет и доступом в ЭИОС филиала.

Программное обеспечение

При проведении лекционных занятий предусматривается использование пакета Microsoft Office.

При проведении лабораторных работ студентами предусматривается использование среды разработки, программы N-Unit (Свободная лицензия, условия использования по ссылке: <http://nunit.org>), программы Jmeter (Свободная лицензия, условия использования по ссылке: <https://jmeter.apache.org/>), текстового редактора для оформления отчетов.

8. ОБЕСПЕЧЕНИЕ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ И ИНВАЛИДОВ

В ходе реализации дисциплины используются следующие дополнительные методы обучения, текущего контроля успеваемости и промежуточной аттестации обучающихся в зависимости от их индивидуальных особенностей:

для слепых и слабовидящих:

- лекции оформляются в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением;
- письменные задания выполняются на компьютере со специализированным программным обеспечением, или могут быть заменены устным ответом;
- обеспечивается индивидуальное равномерное освещение не менее 300 люкс;
- для выполнения задания при необходимости предоставляется увеличивающее устройство; возможно также использование собственных увеличивающих устройств;
- письменные задания оформляются увеличенным шрифтом;
- экзамен и зачёт проводятся в устной форме или выполняются в письменной форме на компьютере.

для глухих и слабослышащих:

- лекции оформляются в виде электронного документа;
- письменные задания выполняются на компьютере в письменной форме;
- экзамен и зачёт проводятся в письменной форме на компьютере; возможно проведение в форме тестирования.

для лиц с нарушениями опорно-двигательного аппарата:

- лекции оформляются в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением;
- письменные задания выполняются на компьютере со специализированным программным обеспечением;
- экзамен и зачёт проводятся в устной форме или выполняются в письменной форме на компьютере;
- используется специальная учебная аудитория для лиц с ЛОВЗ – ауд. 106 главного учебного корпуса по адресу 214013, г. Смоленск, Энергетический пр-д, д.1, здание энергетического института (основной корпус).

При необходимости предусматривается увеличение времени для подготовки ответа.

Процедура проведения промежуточной аттестации для обучающихся устанавливается с учётом их индивидуальных психофизических особенностей. Промежуточная аттестация может проводиться в несколько этапов.

При проведении процедуры оценивания результатов обучения предусматривается использование технических средств, необходимых в связи с индивидуальными особенностями обучающихся. Эти средства могут быть предоставлены филиалом, или могут использоваться собственные технические средства.

Проведение процедуры оценивания результатов обучения допускается с использованием дистанционных образовательных технологий.

Обеспечивается доступ к информационным и библиографическим ресурсам в сети Интернет для каждого обучающегося в формах, адаптированных к ограничениям их здоровья и восприятия информации:

для слепых и слабовидящих:

- в печатной форме увеличенным шрифтом;

- в форме электронного документа;
- в форме аудиофайла.

для глухих и слабослышащих:

- в печатной форме;
- в форме электронного документа.

для обучающихся с нарушениями опорно-двигательного аппарата:

- в печатной форме;
- в форме электронного документа;
- в форме аудиофайла.

9. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература.

1. Святослав Куликов «Тестирование программного обеспечения. Базовый курс». Тестирование программного обеспечения. Издательство ДиаСофт, 2001.
2. Орлов С.А. Технологии разработки программного обеспечения: Учебник /С.Орлов –СПб. Питер, 2012. -464 с.: ил.
3. Черников Б.В. Оценка качества программного обеспечения: Практикум; учебное пособие / Б.В. Черников, Б.Е. Поклонов /Под ред. Б.В. Черникова –М: ИД «ФОРУМ»: ИНФРА-М, 2012, - 400 с.: ил.

Дополнительная литература.

1. Л. Тамре. Введение в тестирование программного обеспечения. - М.: Издательский дом «Вильяме», 2003. - С. 368.
2. Джек Фолк, Сэм Канер, Енг. Кек Нгуен. Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений — К.: Издательство «ДиаСофт», 2001. — 544 с.

Список авторских методических разработок.

А.И. Гаврилов, комплект лекций по дисциплине «тестирование программного обеспечения» в формате электронных файлов расположен на кафедральных ресурсах в ауд. Б-209

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

Но- мер изме- мене- ния	Номера страниц				Всего стра- ниц в доку- менте	Наименование и № документа, вводящего изменения	Подпись, Ф.И.О. внесшего измене- ния в данный эк- земпляр	Дата внесения из- менения в данный эк- земпляр	Дата введения из- менения
	изме- нен- ных	заме- нен- ных	но- вых	анну- лиро- ванн- ых					
1	2	3	4	5	6	7	8	9	10