

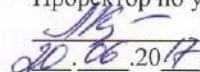
Утверждено приказом ГОУ ВПО ДонГУУ от 23.08.2016г. №675

ДОНЕЦКАЯ НАРОДНАЯ РЕСПУБЛИКА  
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«ДОНЕЦКАЯ АКАДЕМИЯ УПРАВЛЕНИЯ И ГОСУДАРСТВЕННОЙ СЛУЖБЫ  
ПРИ ГЛАВЕ ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ»

ФАКУЛЬТЕТ ГОСУДАРСТВЕННОЙ СЛУЖБЫ И УПРАВЛЕНИЯ  
КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

УТВЕРЖДАЮ

Проректор по учебной работе

 Л.Н.Костина

20.08.2017 г.

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

**«Тестирование ИТ-систем»**

Направление подготовки

09.04.03 «Прикладная информатика»

Донецк

2017

Рабочая программа учебной дисциплины «Тестирование ИТ-систем» для студентов 1 курса образовательного уровня «магистр» направления подготовки 09.04.03 «Прикладная информатика» очной формы обучения.

Автор(ы),  
разработчик(и): старший преподаватель, Э.М. Лебезова

Программа рассмотрена на  
заседании ПМК кафедры

«Прикладная информатика»

Протокол заседания ПМК от

08.06.2017 № 10

Председатель ПМК



А. Н. Верзилов

Программа рассмотрена на  
заседании кафедры

Информационных технологий

Протокол заседания кафедры от

09.06.2017 № 13

Заведующий кафедрой



Н. В. Брадул

## 1. Цель освоения дисциплины и планируемые результаты обучения по дисциплине (соотнесенные с планируемыми результатами освоения образовательной программы)

Обеспечение профессионального образования, способствующего социальной, академической мобильности, востребованности на рынке труда, успешной карьере в области тестирования современных информационных технологий.

Формирование у студентов устойчивых теоретических знаний и практических навыков в области обеспечения качества ИТ-систем, использования средств автоматизированного тестирования интернет-ориентированных приложений и программных продуктов, реализующих управление процессом обеспечения качества ИТ-систем.

Планируемые результаты:

- формирование у магистрантов навыков анализа технической документации, обнаружения и исправления дефектов в документации; использования автоматизированных средств и сред управления технической документацией в контексте обеспечения качества ИТ-систем;
- использования магистрантами средств автоматизированного тестирования ИТ-систем;
- изучение принципов анализа и тестирования проектной и продуктной документации, проектирования и разработки тестовых случаев и тестовых сценариев, автоматизации тестирования ИТ-систем
- овладение магистрантами методами выполнения тестовых случаев и документирования обнаруженных дефектов

Компетенции обучающегося, формируемые в результате освоения дисциплины:

Код соответствующей компетенции по ФГОС	Наименование компетенций	Результат освоения (знать, уметь, владеть)
ОК-7	Способность к абстрактному мышлению, анализу, синтезу	<b>Знать:</b> способы анализа и синтеза информационных технологий. <b>Уметь:</b> абстрактно мыслить и использовать в практической деятельности эти умения. <b>Владеть:</b> навыками самостоятельного анализа новых технологий и систем программирования.
ОПК-3	Способность исследовать современные проблемы и методы прикладной	<b>Знать:</b> - сущность и проблемы прикладной информатики;

	информатики и научно-технического развития ИКТ	<p>- сущность и значение научно-технического развития ИКТ.</p> <p><b>Уметь:</b></p> <ul style="list-style-type: none"> <li>- выявлять проблемы развития прикладной информатики;</li> <li>- оценивать значение научно-технического развития ИКТ, опасности и угрозы, возникающие в этом процессе.</li> </ul> <p><b>Владеть:</b></p> <ul style="list-style-type: none"> <li>- способами выявления проблем развития прикладной информатики;</li> <li>- методами оценивания значения научно-технического развития ИКТ, оценивания опасностей и угроз, возникающих в этом процессе.</li> </ul>
ПК-8	Способность анализировать данные и оценивать требуемые знания для решения нестандартных задач с использованием математических методов и методов компьютерного моделирования	<p><b>Знать:</b></p> <p>способы анализа данных для решения нестандартных задач с использованием математических методов и методов компьютерного моделирования.</p> <p><b>Уметь:</b></p> <p>работать с документами, применяемыми на этапах планирования и выполнения тестирования.</p> <p><b>Владеть:</b></p> <p>навыками разрабатывать планы тестирования и тестовые примеры с помощью методов компьютерного моделирования.</p>
ПК-10	Способность принимать участие в реализации профессиональных коммуникаций в рамках проектных групп, обучать пользователей КИС	<p><b>Знать:</b></p> <p>процесс тестирования программного обеспечения и жизненный цикл программного продукта.</p> <p><b>Уметь:</b></p> <p>коммуницировать в рамках проектных групп, обучать тестировщиков требованиям с точки зрения пригодности к тестированию</p> <p><b>Владеть:</b></p> <p>технологиями тестирования и управления качеством.</p>
ОПК-6	Способность к профессиональной эксплуатации современного электронного оборудования в соответствии с целями основной образовательной программы магистратуры	<p><b>Знать:</b> принципы и методы системного подхода и методы формализации решения прикладных задач.</p> <p><b>Уметь:</b></p> <p>применять принципы и методы системного подхода и методы формализации решения прикладных задач.</p> <p><b>Владеть:</b></p> <p>принципами и методами системного подхода и методы формализации решения прикладных задач.</p>

## 2. Место дисциплины в структуре основной образовательной программы

Дисциплина «Тестирование ИТ-систем» относится к базовым дисциплинам профессионального цикла учебного плана по направлению подготовки 09.04.03

«Прикладная информатика».

### 2.1. Требования к предварительной подготовке обучающегося

Список дисциплин, результаты которых необходимы для изучения данной дисциплины: «Информатика и программирование», «Проектирование информационных систем», «Базы данных», «Разработка информационных систем».

### 2.2. Дисциплины и/или практики, для которых освоение данной дисциплины необходимо как предшествующее:

Основные положения дисциплины могут быть использованы в таких учебных дисциплин, как «Методология и технология проектирования информационных систем», «Управление проектами информатизации предприятий», «Технологии создания интернет-приложений», а также в дальнейшем при подготовке магистерской диссертации.

### 3. Объем дисциплины в кредитах (зачетных единицах) с указанием количества академических часов, выделенных на аудиторную (по видам учебных занятий) и самостоятельную работу студента

Вариант таблицы при изучении дисциплины в одном семестре

	Зачетные единицы (кредиты ECTS)	Всего часов		Форма обучения (вносятся данные по реализуемым формам)	
		О	З	Очная	Заочная
				Семестр №2	Семестр №
<b>Общая трудоемкость</b>	<b>3</b>	<b>108</b>	<b>108</b>	<b>Количество часов на вид работы:</b>	
<b>Виды учебной работы, из них:</b>					
<b>Аудиторные занятия (всего)</b>				<b>42</b>	<b>6</b>
В том числе:					
Лекции				<b>14</b>	<b>2</b>
Семинарские занятия / Практические занятия				<b>28</b>	<b>4</b>
<b>Самостоятельная работа (всего)</b>				<b>66</b>	<b>102</b>
<b>Промежуточная аттестация</b>					
В том числе:					
зачет /экзамен				экзамен	

### 4. Содержание дисциплины, структурированное по разделам (темам) с указанием отведенного на них количества академических часов и видов учебных занятий

#### 4.1. Разделы (темы) дисциплины с указанием отведенного на них количества академических часов и видов учебных занятий

Наименование раздела, темы дисциплины	Виды учебной работы (бюджет времени) (вносятся данные по реализуемым формам)									
	Очная форма обучения					Заочная форма обучения				
	Лекции	Практические занятия	Семинарские занятия	Самостоятель- ная работа	Всего	Лекции	Практические занятия	Семинарские занятия	Самостоятель- ная работа	Всего
1	2	3	4	5	6	7	8	9	10	11
<b>«Основы тестирования»</b>										
Тема 1.1. Вопросы организации тестирования	2	4		10	16	1			17	18
Тема 1.2. Модульное тестирование.	2	4		10	16				18	18
Тема 1.3. Интеграционное тестирование.	2	4		10	16				18	18
<b>Итого по разделу:</b>	6	12		30	48	1			53	54
<b>«Методы тестирования»</b>										
Тема 2.1. Системное и регрессионное тестирование. Автоматизация тестирования	4	6		12	22	1	2		15	18
Тема 2.2. Особенности индустриального тестирования.	2	4		12	18		2		16	18
Тема 2.3. Регрессионное тестирование: цели и задачи, методики, алгоритм и программная система поддержки	2	6		12	20				18	18
<b>Итого по разделу:</b>	8	16		36	60	1	4		49	56
<b>Всего за семестр:</b>	14	28		66	108	2	4		102	108

## 4.2. Содержание разделов дисциплины:

Наименование раздела, темы дисциплины	Содержание разделов дисциплины	Содержание семинарских/практических занятий		
			Кол-во часов	
			0	3
1	2	3	4	5
<b>Раздел 1. «Основы тестирования»</b>				
Тема 1.1. Вопросы организации тестирования	Понятия отладки и тестирования. Методы поиска ошибок и процедура тестирования. Фазы тестирования, основные проблемы тестирования. Требования к идеальному критерию тестирования и классы частных критериев. Особенности применения структурных и функциональных критериев на базе конкретных примеров. Особенности применения методов стохастического тестирования и метод оценки скорости выявления ошибок. Метрики оценки оттестированности проекта, примеры плоской и иерархической моделей проекта.	<b>Практическое занятие №</b>		
		1. Описание тестируемой системы и ее окружения. Планирование тестирования	4	
Тема 1.2. Модульное тестирование.	Особенности модульного тестирования, подходы к тестированию на основе потока управления, потока данных. Динамические и статические методы при структурном подходе. Взаимосвязь сборки модулей и методов интеграционного тестирования. Подходы монолитного, инкрементального, нисходящего и восходящего тестирования.	<b>Практическое занятие №</b>		
		2 Модульное тестирование на примере классов.	4	
Тема 1.3. Интеграционное тестирование.	Особенности интеграционного тестирования в процедурном программировании. Модель объектно-ориентированной программы, использующая понятие Р-путей и ММ-путей. Оценки сложности тестирования и методика тестирования объектно-	<b>Практическое занятие</b>		
		3. Интеграционное тестирование.	4	

Наименование раздела, темы дисциплины	Содержание разделов дисциплины	Содержание семинарских/практических занятий		
			Кол-во часов	
			0	3
1	2	3	4	5
	ориентированной программы.			
<b>Раздел 2. «Методы тестирования»</b>				
Тема 2.1. Системное и регрессионное тестирование. Автоматизация тестирования	Задачи и категории тестов, применяемые в системном тестировании. Регрессионное тестирование и комбинирование различных уровней тестирования. Структура тестового набора для автоматического прогона. Структура инструментальной системы автоматизации тестирования. Эффективность методов тестирования.	<b>Практическое занятие</b>		
		4. Системное тестирование.	4	2
		5. Ручное тестирование	4	2
Тема 2.2. Особенности индустриального тестирования.	Подход к обеспечению качества программного продукта средствами тестирования. Фазы процесса тестирования и шаги тестового цикла, применяемые в индустриальном тестировании. Структура документа «Тестовый план». Подходы к тестированию спецификаций и сценариев. Ручной подход и подход генерации тестовых наборов при разработке тестов. Методы автоматизации исполнения тестов. Особенности документирования тестовых процедур для ручных и автоматизированных тестов, описаний тестовых наборов и тестовых отчетов.	<b>Практическое занятие</b>		
		6. Автоматизация тестирования с помощью скриптов.	4	
		7. Автоматическая генерация тестов на основе формального описания	4	
Тема 2.3. Регрессионное тестирование: цели и задачи, методики, алгоритм и программная система поддержки	Виды регрессионного тестирования. Применение и классификация методов выборочного регрессионного тестирования. Возможности повторного использования тестов. Случайные методы, безопасные методы, методы минимизации, методы, основанные на покрытии кода.	<b>Практическое занятие</b>		
		8. Описание автоматической генерации MSC тестов.	4	
		9. Использование MS Visio для генерации MPR-файлов.	4	



Наименование раздела, темы дисциплины	Содержание разделов дисциплины	Содержание семинарских/практических занятий		
			Кол-во часов	
			0	3
1	2	3	4	5
	Интеграционное регрессионное тестирование и регрессионное тестирование объектно-ориентированных программ. Метод уменьшения объема тестируемой программы, методы упорядочения тестов. Алгоритм регрессионного тестирования и программная система его поддержки.			
			36	4

## 5. Перечень учебной литературы, необходимой для освоения дисциплины

### 5.1. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Методическое обеспечение для самостоятельной работы представлено на образовательном портале «ДонАУиГС»: Режим доступа: <http://elearn.dsum.org>

#### Контрольные вопросы для самоподготовки:

1. - Тестирование в модели жизненного цикла разработки ПО
2. - Связь проектирования и тестирования.
3. - Связь тестирования и качества разрабатываемого ПО.
4. - Документирование тестов и рабочего продукта (Test documentation and work products)
5. - Аутсорсинг тестирования программного обеспечения
6. - Тестирование баз данных.
7. - Модульное и интеграционное тестирование.
8. - Системное и регрессионное тестирование. Автоматизация тестирования.
9. - Особенности промышленного тестирования.
10. - Регрессионное тестирование: цели и задачи, методики, алгоритм и программная система поддержки.

#### Тематика заданий для индивидуальной работы:

Описать назначение программ тестирования ПО:

1. IBM Rational TestManager - управление тестированием;

2. IBM Rational PurifyPlus (Purify, PureCoverage, Quantify) - анализ работы системы в режиме RunTime;
3. IBM Rational Robot - функциональное и нагрузочное тестирование;
4. IBM Rational TestFactory - автоматизация создания тестов;
5. IBM Rational XDE Tester - функциональное тестирование Java и web-приложений.
6. QA - компонентам: QA Partner - среда для разработки, компиляции и выполнения тестов; QA Organizer - модуль для разработки планов тестирования и управления процессом тестирования; Agent - модуль, поддерживающий работу в сети;
7. TestRecorder – набор библиотек для MS VisualStudio

### **5.2. Перечень основной учебной литературы**

1. Коробейник А. - Краткие основы тестирования программного обеспечения. Методические рекомендации. Режим доступа: [http://elearn.dsum.org/pluginfile.php/889/mod\\_resource/content/0/testirovaniepo.pdf](http://elearn.dsum.org/pluginfile.php/889/mod_resource/content/0/testirovaniepo.pdf)
2. Сэм Канер, Джек Фолк. Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений. Режим доступа
3. Макгрегор Дж., Сайкс Д. Тестирование объектно-ориентированного программного обеспечения: практическое пособие разработчикам, менеджерам проектов, программистам, - М: DiaSoft. 2002. -432 с. Режим доступа:
4. <http://forcoder.ru/testing/testirovanie-obektno-orientirovannogo-programmnogo-obespecheniya-943>

### **5.3. Перечень дополнительной литературы**

1. Ошероув Рой. Искусство автономного тестирования с примерами на C#, - М: ДМК Пресс. 2014. - 360 с. Режим доступа:
2. [http://elearn.dsum.org/pluginfile.php/889/mod\\_resource/content/0/testirovaniepo.pdf](http://elearn.dsum.org/pluginfile.php/889/mod_resource/content/0/testirovaniepo.pdf)
3. Котляров В.П. Основы современного тестирования программного обеспечения, разработанного на C#: Учебное пособие. - СПб.: СПбГПУ, 2004. - 170 с. Режим доступа: <http://window.edu.ru/resource/713/41713>
4. Роман Савин - Тестирование Дот Ком, - М: Дело. 2007. - 380 с. Режим доступа: [http://elearn.dsum.org/pluginfile.php/888/mod\\_resource/content/2/testirovanie\\_dot-com.pdf](http://elearn.dsum.org/pluginfile.php/888/mod_resource/content/2/testirovanie_dot-com.pdf)

### **6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

1. Википедия - <http://ru.wikipedia.org/>
2. Интернет-журнал по ИТ - <http://www.rsdn.ru>

3. Интернет-портал образовательных ресурсов по ИТ - <http://www.intuit.ru>
4. Интернет-портал ресурсов по ИТ (Microsoft) - <http://msdn.microsoft.com/ru-ru/ms348103.aspx>
5. Образовательный портал «ДонАУиГС» [Электронный ресурс]: Режим доступа: <http://elearn.dsum.org>
6. Научная электронная библиотека [www.eLIBRARY.RU](http://www.eLIBRARY.RU)

## **7. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем**

### **7.1. Перечень информационных технологий**

- Equivalence Class Testing (разбиение тестов на классы эквивалентности)
- Boundary Values Testing
- Use Case Testing
- Decision Table Testing
- State-Transition Testing
- Pairwise Testing
- Domain Analysis Testing
- Ad hoc Testing
- Exploratory Testing
- Технология командной разработки проектов на GitHub.
- Дистанционное обучение с помощью СДО Moodle.

### **7.2. Перечень программного обеспечения**

1. Операционная система Microsoft Windows 10 (или выше);
2. Microsoft Office 2010 Professional (или выше);
3. Microsoft Visual Studio 2015 (или выше);
4. Microsoft Sql Server 2012 R2 Express (или выше);
5. Среда SQL Server Management Studio 2012 (или выше)

### **7.3. Перечень информационных справочных систем**

1. Вопросы тестирования и повышения качества программного обеспечения  
<http://www.software-testing.ru>
2. Сайт, посвященный тестированию программного обеспечения  
<http://www.protesting.ru/testing/>

3. Тестирование. Фундаментальная теория <https://dou.ua/forums/topic/13389/>
4. <http://qalight.com.ua/baza-znaniy/otkuda-berutsya-oshibki-v-po/>
5. <http://devopswiki.net/index.php>
6. [https://dou.ua/calendar/15479/?from=premiuevents\\_3069](https://dou.ua/calendar/15479/?from=premiuevents_3069)
7. Подкаст тестировщиков <http://radio-qa.com/>

## **8. Фонд оценочных средств для контроля уровня сформированности компетенций**

### **8.1. Виды промежуточной аттестации**

Текущий контроль успеваемости позволяет оценить уровень сформированности элементов компетенций (знаний и умений), компетенций с последующим объединением оценок и проводится в форме устного опроса (фронтальный, индивидуальный, комплексный), письменной проверки (ответы на вопросы, тестовые задания), включая задания для самостоятельной работы.

#### **Доклады**

Доклад—это краткое изложение в письменном виде содержания и результатов индивидуальной учебно-исследовательской деятельности, имеет регламентированную структуру, содержание и оформление. Его задачами являются:

1. Формирование умений самостоятельной работы студентов с источниками литературы, их систематизация;
2. Развитие навыков логического мышления;
3. Углубление теоретических знаний по проблеме исследования.

Текст доклада должен содержать аргументированное изложение определенной темы. Доклад должен быть структурирован (по главам, разделам, параграфам) и включать разделы: введение, основная часть, заключение, список используемых источников. В зависимости от тематики доклада к нему могут быть оформлены приложения, содержащие документы, иллюстрации, таблицы, схемы и т.д. Доклады предусмотрены по темам №1-9.

#### **Тестовые задания**

В качестве оценочных средств на протяжении семестра используется тестирование.

Тестовые задания могут формулироваться как в форме, используемой в федеральном электронном интернет-тестировании (интернет-экзамене), так и оригинальной авторской форме, с открытыми вариантами ответов. Тестовые задания по дисциплине включены в базу тестовых заданий образовательного портала [elearn.dsum.org](http://elearn.dsum.org). Тестовые задания предусмотрены по темам №1-9

### **Примеры тестовых заданий**

1. Что такое отладка программного средства?

2. Что такое тестирование программного средства?
3. Что такое автономная отладка программного средства?
4. Что такое комплексная отладка программного средства?
5. Что такое ведущий отладочный модуль?
6. Что такое отладочный имитатор программного модуля?

## 8.2. Показатели и критерии оценки результатов освоения дисциплины

Средним баллом за дисциплину является средний балл за текущую учебную деятельность.

Механизм конвертации результатов изучения студентом дисциплины в оценки по традиционной (государственной) шкале и шкале ECTS представлен в таблице.

<b>Средний балл по дисциплине (текущая успеваемость)</b>	<b>Отношение полученного студентом среднего балла по дисциплине к максимально возможной величине этого показателя</b>	<b>Оценка по государственной шкале</b>	<b>Оценка по шкале ECTS</b>	<b>Определение</b>
4,5 – 5,0	90% – 100%	5	A	отлично – отличное выполнение с незначительным количеством неточностей (до 10%)
4,0 – 4,49	80% – 89%	4	B	хорошо – в целом правильно выполненная работа с незначительным количеством ошибок (до 20%)
3,75 – 3,99	75% – 79%	4	C	хорошо – в целом правильно выполненная работа с незначительным количеством ошибок (до 25%)
3,25 – 3,74	65% – 74%	3	D	удовлетворительно – неплохо, но со значительным количеством недостатков (до 35%)
3,0 – 3,24	60% – 64%	3	E	достаточно – выполнение удовлетворяет

				минимальные критерии, но со значительным количеством недостатков (до 40%)
до 3,0	35% – 59%	2	FX	неудовлетворительно с возможностью повторной сдачи (ошибок свыше 40%)
	0 – 34%	2	F	неудовлетворительно – надо поработать над тем, как получить положительную оценку (ошибок свыше 65%)

### 8.3. Критерии оценки работы студента

При усвоении каждой темы за текущую учебную деятельность студента выставляются оценки по 5-балльной (государственной) шкале. Оценка за каждое задание в процессе текущей учебной деятельности определяется на основе процентного отношения операций, правильно выполненных студентом во время выполнения задания:

- 90-100% – «5»,
- 75-89% – «4»,
- 60-74% – «3»,
- менее 60% – «2».

Если на занятии студент выполняет несколько заданий, оценка за каждое задание выставляется отдельно.

#### 8.3.1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или опыта деятельности, характеризующие этапы формирования компетенций в процессе освоения образовательной программы)

Текущий контроль успеваемости

Тема 1.1. Вопросы организации тестирования:

Контрольный письменный опрос по следующим вопросам:

Вариант 1

**Теоретические задания:**

Что такое метод выборочного регрессионного тестирования?

В чем заключаются особенности интеграционного тестирования для процедурного программирования?

Возможно ли тестирование программы на всех возможных путях УПП?

Возможно ли тестирование программы на всех допустимых значениях параметров?

Зачем нужен Log-файл?

Зачем нужна спецификация тестирования?

Как определить цели тестирования программного проекта?

Как определяется тестовая стратегия?

Как осуществить планирование тестирования?

Как разрабатываются тесты для тестирования спецификаций?

### **Практические задания:**

Дана функция  $P$ , ее измененная версия  $P'$  и набор тестов  $T$ , разработанный для тестирования  $P$ . Требуется, используя метод минимизации, отобрать подмножество  $T'$  для тестирования  $P'$ .

Дано: функция  $P$ , ее измененная версия  $P'$  и набор тестов  $T$ , разработанный для тестирования  $P$ . Требуется, используя безопасный метод, отобрать подмножество  $T'$  для тестирования  $P'$ .

Модуль  $E$  исходной программы содержал ошибку. Ее исправление потребовало изменения модулей  $B$ ,  $D$  и  $E$ . Тест 1 покрывает модули  $A$ ,  $B$  и  $C$ . Тест 2 покрывает модули  $A$ ,  $C$  и  $E$ . Тест 3 покрывает модули  $D$  и  $E$ . Каким должен быть порядок прогона тестов, если при упорядочении ставится цель скорейшего достижения наибольшей степени покрытия модулей?

На предыдущей версии программы тест 1 завершился в состоянии  $A$ , тест 2 – в состоянии  $B$ , а тест 3 – в состоянии  $C$ . На текущей версии программы тест 1 завершился в состоянии  $A$ , тест 2 – в состоянии  $C$ , а тест 3 – в состоянии  $D$ . На базе какого теста наиболее целесообразна разработка новых тестов?

Вариант 2

### **Теоретические задания:**

Как реализуются динамические методы построения тестовых путей?

Какая информация должна собираться при тестировании для применения метода оценки скорости выявления ошибок?

Какая информация должна сопровождать действие по исправлению ошибки и перевод дефекта в состояние Resolved?

Какие активности входят в систему обеспечения качества программного продукта?

Какие бывают состояния дефекта?

Какие возможности используются для построения дуг графовой модели в ООП?

Какие задачи решает тестировщик, проверяя изменения, внесенные разработчиком в код?

Какие задачи решаются на этапе системного тестирования?

Какие задачи у модульного тестирования?

Какие из перечисленных средств использует в своей работе система поддержки регрессионного тестирования?

### **Практические задания:**

1. Дана функция  $P$ , ее измененная версия  $P'$  и набор тестов  $T$ , разработанный для тестирования  $P$ . Требуется, используя метод стопроцентного покрытия кода, отобрать подмножество  $T'$  для тестирования  $P'$ .

2. Класс  $A$  является потомком класса  $B$ , который, в свою очередь, является потомком класса  $C$ . Класс  $B$  был изменен. Тестирование каких классов необходимо обеспечить?

3. Модуль  $E$  исходной программы содержал ошибку. Ее исправление потребовало изменения модулей  $B$ ,  $D$  и  $E$ . Тест 1 покрывает модули  $A$ ,  $B$  и  $C$ . Тест 2 покрывает модули  $A$ ,  $C$  и  $E$ . Тест 3 покрывает модули  $D$  и  $E$ . Каким должен быть порядок прогона тестов, если при упорядочении ставится цель скорейшего роста вероятности того, что тестируемая система надежна?

4. На предыдущей версии программы тест 1 завершился в состоянии  $A$ , тест 2 – в состоянии  $B$ , а тест 3 – в состоянии  $C$ . На текущей версии программы тест 1 завершился в состоянии  $A$ , тест 2 – в состоянии  $C$ , а тест 3 – в состоянии  $D$ . На базе какого теста наиболее целесообразна разработка новых тестов?

Тема 1.2. Модульное тестирование:

Контрольный письменный опрос по следующим вопросам:

Вариант 1

### **Теоретические задания:**

Какие категории тестов разрабатываются для системного тестирования?

Какие классы частных критериев тестируемости известны?

Какие методы регрессионного тестирования применяются в условиях высоких требований к качеству программного продукта?

Какие методы регрессионного тестирования применяются в условиях отсутствия программных средств поддержки регрессионного тестирования?

Какие методы регрессионного тестирования применяются, если исходный набор состоит из структурных тестов?



Какие метрики обычно входят в список тестовых?

Какие операции позволяет автоматизировать система поддержки регрессионного тестирования?

Какие операции при использовании системы поддержки регрессионного тестирования выполняются вручную?

Какие основные уровни выделены в процессе тестирования?

Какие подходы используются для обоснования истинности программ?

### **Практические задания:**

Ожидаемое время работы метода отбора тестов – 1 час. Среднее время выполнения одного теста – 1 час. Тестовый набор состоит из 8 тестов. Каково значение порога целесообразности?

Ожидаемое время работы метода отбора тестов – 2 часа. Среднее время выполнения одного теста – 2 часа. Тестовый набор состоит из 8 тестов. Каково значение порога целесообразности?

При создании очередной версии программы была добавлена функция A, функция D была удалена, функция C – изменена, а функция U – оставлена без изменений. К какой группе относится тест, покрывающий только функцию D?

Тестируемая программа состоит из модулей A, B, C и D, взаимодействующих по принципу «каждый с каждым». Модули A и B были изменены. Тестирование каких интерфейсов необходимо обеспечить, если используется брандмауэр?

Функция `int function(int number)` была изменена и возвращает  $(number + 2)$  вместо  $(number + 1)$ . На каком уровне возможно повторное использование теста «1»?

### Вариант 2

#### **Теоретические задания:**

На основе каких принципов строятся тесты для модульного тестирования?

Назовите критерии стохастического тестирования.

Назовите недостатки структурных критериев.

Назовите недостатки функциональных критериев.

Перечислите метрики оценки оттестированности программного проекта?

Перечислите разновидности функциональных критериев.

Сколько тестов потребуется для проверки программы, реализующей задержку на неопределенное количество тактов?

Сопровождение какого типа соответствует случаю исправления ошибки в коде?

Сопровождение какого типа соответствует случаю реализации новых требований пользователя?

Сопровождение какого типа соответствует случаю улучшения временных характеристик программы?

### **Практические задания:**

1. Ожидаемое время работы метода отбора тестов – 2 часа. Среднее время выполнения одного теста – 1 час. Тестовый набор состоит из 8 тестов. Каково значение порога целесообразности?

2. При создании очередной версии программы была добавлена функция А, функция D была удалена, функция С – изменена, а функция U – оставлена без изменений. К какой группе относится тест, покрывающий только функцию А?

3. При создании очередной версии программы была добавлена функция А, функция D была удалена, функция С – изменена, а функция U – оставлена без изменений. К какой группе относится тест, покрывающий только функции С и U?

4. Функция `int function(int number)` была изменена и возвращает  $(number + 2)$  вместо  $(number + 1)$ . На каком уровне возможно повторное использование теста «1»?

5. Функция `int function(int number)` была изменена и приобрела вид `int function(char* string)`. На каком уровне возможно повторное использование теста «1»?

### **Вопросы на экзамен:**

1. Основные понятия тестирования: концепция тестирования, подходы.
2. Различия тестирования и отладки.
3. Фазы и технология тестирования.
4. Проблемы тестирования.
5. Критерии выбора тестов: структурные, функциональные, стохастические.
6. Критерии выбора тестов: мутационные, оценки покрытия проекта.
7. Разновидности тестирования: модульное, интеграционное.
8. Разновидности тестирования: системное, регрессионное.
9. Автоматизация тестирования, издержки тестирования.
10. Особенности процесса и технологии индустриального тестирования: планирование тестирования, подходы к разработке тестов.
11. Особенности ручной разработки и генерации тестов.
12. Автоматизация тестового цикла.
13. Документирование тестирования, обзоры и метрики.

14. Регрессионное тестирование: особенности и виды регрессионного тестирования.
15. Методы отбора тестов, оценка эффективности.
16. Проблемы и перспективы развития современной программной инженерии.
17. Различные подходы в программировании: «снизу-вверх», «сверху-вниз» (структурный подход), объектно-ориентированный.
18. Связь тестирования и качества разрабатываемого ПО.
19. Типы тестов и их роль в процессе разработки ПО.
20. Документирование и анализ ошибок.
21. Разработка тестов.
22. Устранение ошибок в программе с помощью инструментов отладки
23. Построение управляющего графа программы
24. Составление набора тестов на основе управляющего графа
25. Создание юнит-тестов для классов
26. Проведение регрессионного тестирования
27. Составление тестов для проведения функционального тестирования
28. Тестирование программы на основе функциональных требований
29. Составление тестов для автоматического тестирования
30. Инструментальные средства автоматизации тестирования
31. Тестирование баз данных
32. Определение предельных параметров производительности программы на основе нагрузочного тестирования
33. Моделирование исключительных ситуаций
34. Корректная обработка исключительных ситуаций
35. Составление наборов тестов для интеграционного тестирования
36. Тестирование пользовательских интерфейсов
37. Оценка полноты наборов тестов для компонентной программы с пользовательским интерфейсом
38. Примеры построения тестов.
39. Оценка степени тестируемости ПО.
40. Критерии структурного тестирования.
41. Построение управляющего графа программы.
42. Функциональное тестирование (Метод « черного ящика»).
43. Тестирование циклов.
44. Тестирование потоков данных.
45. Тестирование транзакций.

46. Характеристики хорошего теста.
47. Нагрузочные испытания.
48. Тестирования баз данных.
49. Стандарты на разработку интерфейса, примеры основных ошибок при разработке интерфейсов программ.

**Темы докладов:**

1. Специализированное тестирование(Ad hoc testing).
2. Исследовательское тестирование (Exploratory testing)
3. Тесты на основе конечного автомата(Finite-state machine-based).
4. Случайное тестирование(Random testing)
5. Тесты, базирующиеся на блок-схеме (Control-flow-based criteria)
6. Тесты на основе потоков данных (Data-flow-based criteria).
7. Предположение ошибок(Error guessing)
8. Тестирование мутаций(Mutation testing)
9. Тестирование, базирующееся на надежности инженерного процесса (Software Reliability Engineered Testing).
10. Компонентно-ориентированное тестирование
11. Web-ориентированное тестирование
12. Тестирование на соответствие протоколам
13. Тестирование систем реального времени
14. Автоматизация тестирования с использованием HP LoadRunner
15. Автоматизация тестирования с использованием HP QuickTest Professional
16. Автоматизация тестирования с использованием HP Quality Center
17. Автоматизация тестирования с использованием IBM Rational FunctionalTester.
18. Автоматизация тестирования с использованием IBM Rational PerformanceTester.
19. Автоматизация тестирования с использованием IBM Rational TestStudio.
20. Автоматизация тестирования с использованием SmartBear Software TestComplete
21. Автоматизация тестирования с использованием AutoIt
22. Автоматизация тестирования с использованием Selenium

### 8.3.2. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и/или опыта деятельности

Виды самоподготовки	Часть курса	Рекомендации по выполнению и правила оценки	Методологическое обеспечение
Написать программы и провести тестирование их.	3 задачи 1-3	Практические задачи принимаются в течение 2-3 недель с момента, как были заданы (срок зависит от степени сложности задания)	См. раздел 8.3.2.1
Подготовка к вопросам по теоретическим разделам курса	Теоретическая часть, разделы 1-2	Выполняют все студенты. Лектор выборочно задает вопросы на теоретических занятиях (длительность опроса – не более 5 минут до начала лекции)	См. раздел 8.3.2.2
Подготовка к зачёту	Теоретическая часть, разделы 1-2	Итоговая аттестация в форме диф. зачёта	

#### 8.3.2.1. Список практических задач

1. Составление тестов, проведение тестирования и оценка полноты проведенных тестов на примере программной системы «Калькулятор валют».
2. Составление тестов, проведение тестирования и оценка полноты проведенных тестов на примере программной системы «Проверка орфографии».
3. Составление тестов, проведение тестирования и оценка полноты проведенных тестов на примере автоматизированной программной системы инвентаризации оборудования предприятия

#### 8.3.2.2. Список контрольных вопросов

1. абстракция «черный ящик» (black box testing)
2. абстракция «белый ящик» (white box testing)
3. функциональное тестирование (functional testing)
4. тестирование новых функциональностей (new feature testing)
5. регрессионное тестирование (regression testing)

6. позитивное тестирование (positive testing)
7. негативное тестирование (negative testing)
8. тестирование по документации (formal /documented testing)
9. тестирование опыта пользователя (usability testing)
10. альфа-тестирование (alpha testing)
11. beta testing – Избранные пользователи
12. тестирование «на задымление» (smoke test, sanity test, confidence test)
13. приемочное тестирование (acceptance testing)
14. компонентное тестирование (component testing)
15. интеграционное тестирование (integration testing)
16. системное (или энд-ту-энд) тестирование (system or end-to-end testing)
17. тестирование совместимости (compatibility testing)
18. цикл разработки ПО, и где в нём тестирование
19. что такое тестирование, цель и смысл тестирования (основная и косвенные цели)
20. разница между QA и QC
21. что и как и зачем надо/не надо тестировать
22. «баги» (нахождение, репорт, трэкинг)
23. тест-кейсы (как придумать, как поддерживать, как убить)
24. требования (как читать, как анализировать, как использовать)
25. принципы грамотного составления тест-плана (и решение – нужен ли он кому-то тут)

## **9. Методические рекомендации (указания) для обучающихся по освоению дисциплины**

В курсе широко используются актуальные и эффективные техники для более качественного обучения, социализации:

- Обсуждения тем в специальных группах популярных социальных сетях,
- Командная работа,
- Удаленные технические консультации и видеоконференции,
- Наглядные демонстрации технологий тестирования

## **10. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине**

Компьютерный класс. 10 компьютеров с минимальными характеристиками:

Процессор - Intel Core2Duo 2.4GHz

ОЗУ - 2GB

Жёсткий диск - 7200 RPM

Сетевые возможности - доступ к локальной сети 100 Mb/s, доступ к интернет.

Лекционная аудитория с мультимедийной проекционной техникой в стационарном или переносном варианте. Чтение лекций сопровождается демонстраций электронных презентаций по каждой теме.