

Государственное бюджетное образовательное учреждение
высшего образования Московской области
«Университет «Дубна»
(государственный университет «Дубна»)

Филиал «Протвино»
Кафедра «Информационные технологии»



Рабочая программа дисциплины (модуля)

Технологии тестирования программных продуктов
наименование дисциплины (модуля)

Направление подготовки (специальность)
09.03.01 Информатика и вычислительная техника
код и наименование направления подготовки (специальности)

Уровень высшего образования
бакалавриат
бакалавриат, магистратура, специалитет

Направленность (профиль) программы (специализация)
«Программное обеспечение вычислительной техники и автоматизированных систем»

Форма обучения
очная
очная, очно-заочная, заочная

Протвино, 2021

Преподаватель (преподаватели):
Нурматова Е.В., к.т.н., доцент, кафедра информационных технологий
Фамилия И.О., должность, ученая степень, ученое звание, кафедра; подпись



Рабочая программа разработана в соответствии с требованиями ФГОС ВО по направлению
подготовки (специальности) высшего образования
09.03.01 Информатика и вычислительная техника
(код и наименование направления подготовки (специальности))

Программа рассмотрена на заседании кафедры информационных технологий
(название кафедры)

Протокол заседания №9 от «19» апреля 2021 г.

Заведующий кафедрой  Нурматова Е.В.
(Фамилия И.О., подпись)

Оглавление

1 Цели и задачи освоения дисциплины (модуля)	4
2 Объекты профессиональной деятельности при изучении дисциплины (модуля)	4
3 Место дисциплины (модуля) в структуре ОПОП	4
4 Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников).....	4
5 Объем дисциплины (модуля) в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся	5
6 Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и виды учебных занятий	6
7 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю) и методические указания для обучающихся по освоению дисциплины (модулю)	10
8 Применяемые образовательные технологии для различных видов учебных занятий и для контроля освоения обучающимися запланированных результатов обучения.....	10
9 Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю).....	11
10 Ресурсное обеспечение	18
11 Язык преподавания	21

1 Цели и задачи освоения дисциплины (модуля)

Курс призван познакомить студентов с методологиями тестирования программных продуктов, написанных на высокоуровневых языках программирования с учетом специфики кроссплатформенного кода. Основной целью студента является ознакомление с этими методиками и их апробация на практике. Дополнительное внимание уделено анализу данных, используемых при тестировании. Кроме методик тестирования, в курсе особое место уделено аспектам «чистого» и «защищенного» программирования, а также анализу кода. Таким образом, вспомогательной целью курса является выработка у слушателей представлений о качественном программном коде, работе со средствами коллективной разработки и необходимости непрерывного процесса тестирования.

Задачи изучения дисциплины охватывают теоретические и практические компоненты деятельности подготавливаемого специалиста. Основными задачами курса являются: освоение методик написания качественного кроссплатформенного программного кода с использованием модульной архитектуры приложений; анализ методологий составления модульных, интегральных и системных тестов; анализ входных данных; изучение инструментальных средств отладки программного кода, его профилирования и анализ; освоение методик отладки и тестирования многопоточных приложений; изучение платформ автоматизированного тестирования программного обеспечения и пользовательских интерфейсов.

2 Объекты профессиональной деятельности при изучении дисциплины (модуля)

Объектами профессиональной деятельности в рамках изучаемой дисциплины (модуля) являются:

- автоматизированные системы обработки информации и управления;
- программное обеспечение средств вычислительной техники и автоматизированных систем (программы, программные комплексы и системы).

3 Место дисциплины (модуля) в структуре ОПОП

Дисциплина Б1.В.ДВ.8.2 «Технологии тестирования программных продуктов» относится к дисциплинам по выбору вариативной части блока дисциплин учебного плана. Изучается в VII семестре IV курса.

Дисциплина «Технологии тестирования программных продуктов» базируется на курсах «Информатика», «Математическая логика и теория алгоритмов», «Офисные информационные технологии», «Программирование на языке высокого уровня», «Информационные системы и технологии», «Базы данных», «Теория и технология проектирования», «Архитектура вычислительных систем», «Параллельные и распределенные вычисления», «Операционные системы».

Приступая к изучению дисциплины «Технологии тестирования программных продуктов», студент должен знать принципы функционирования ЭВМ и базовые программные алгоритмы. Работать с разными типами операционных систем. Знать хотя бы один язык программирования высокого уровня. Владеть широким спектром сопутствующих технологий: от case-платформ до СУБД.

После освоения дисциплины «Технологии тестирования программных продуктов» студент будет подготовлен к написанию выпускной квалификационной работы и к профессиональной деятельности на предприятии по профильному направлению.

4 Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников)

Формируемые компетенции <i>(код компетенции, уровень (этап) освоения)</i> <i>(последний – при наличии в карте компетенции)</i>	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций
<p><i>ОПК-4 – способностью участвовать в настройке и наладке программно-аппаратных комплексов, I уровень (пороговый)</i></p>	<p><i>Знать</i> *)</p> <ul style="list-style-type: none"> – инструменты для изменения элементов программы во время выполнения, тестирования программы в сложных условиях (многократного запуска, нехватки оперативной памяти, системной ошибки) – методы и средства верификации работоспособности выпусков программных продуктов <p><i>Уметь</i></p> <ul style="list-style-type: none"> – изменять значения элементов программы во время выполнения, тестировать программу со сложными условиями, предугадывать ошибки выполнения, выявлять подверженные ошибкам места программы; составлять условия применения программы <p><i>Владеть</i></p> <ul style="list-style-type: none"> – методами определения требуемых от программы характеристик и их достижения; методами разбиения работы над программным проектом на этапы и выделения в них элементов для тестирования программы
<p><i>ПК-3 - способность обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности</i></p>	<p><i>Знать:</i></p> <ul style="list-style-type: none"> – основы общей теории сложных систем, классификацию и закономерности систем, методы и модели описания и анализа систем <p><i>Уметь:</i></p> <ul style="list-style-type: none"> – использовать полученные теоретические знания: для получения, хранения, переработки информации; при решении различных задач с использованием специализированных программ <p><i>Владеть:</i></p> <ul style="list-style-type: none"> – навыками применения современных технических средств и информационных технологий для решения задач

*) результат обучения сформулирован на основании требований профессиональных стандартов:
– «Программист» №4 (приказ Министерства труда и социальной защиты РФ от 18 декабря 2013 г. № 679н)

5 Объем дисциплины (модуля) в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Объем дисциплины (модуля) составляет 5 зачетных единиц, всего 180 часов, из которых:

51 час составляет контактная работа обучающегося с преподавателем¹:

17 часов – лекционные занятия;

34 часа – практические занятия.

¹ Перечень видов учебных занятий уточняется в соответствии с учебным планом.

_____ часов – мероприятия текущего контроля успеваемости²;
27 часов – мероприятия промежуточной аттестации⁴ (экзамен),

102 часа составляет самостоятельная работа обучающегося.

6 Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и виды учебных занятий

² В скобках необходимо сделать уточнение, если мероприятия текущего контроля успеваемости и (или) промежуточной аттестации (например, зачет, дифференцированный зачет) проводятся в рамках занятий семинарского типа, групповых или индивидуальных консультаций.

Наименование и краткое содержание разделов и тем дисциплины (модуля) Форма промежуточной аттестации по дисциплине (модулю)	Всего (часы)	В том числе:										
		Контактная работа (работа во взаимодействии с преподавателем), часы из них ³								Самостоятельная работа обучающегося, часы, из них		
		Лекционные занятия	Семинарские занятия	Практические занятия	Лабораторные занятия	...	Групповые консультации	Индивидуальные консультации	Учебные занятия, направленные на проведение текущего контроля успеваемости (коллоквиумы, практические контрольные занятия и др.)*	Всего	Выполнение домашних заданий	Подготовка рефератов и т.п.
VII семестр												
<i>Введение в построение программных продуктов и их тестирование. Введение в предмет. Настройка средств разработки и системы сборки и тестирования для OS Windows, Linux. Введение в теорию приближенных вычислений. Представление чисел в ЭВМ и погрешности. Методы суммирования. Обзор курса лекций.</i>		2		4						4		
<i>«Чистое» кодирование. Понятие модульности при разработке ПО. Повторное использование кода. Создание простейших (модульных) тестов. Визуализация данных, как дополнительное средство тестирования. Представление данных для визуализации в сторонних программах.</i>		2		4						8		
<i>«Защищенное» кодирование. Многократное использование кода. Проверка качества входных данных. Многократное применение кода. Библиотеки и их использование. Связывание библиотек статическое и динамическое. Динамически загружаемые библиотеки. Создание динамически загружаемых библиотек. Поддержка версионности. Динамическая загрузка библиотек.</i>		2		4						8		15
<i>Автоматизация тестирования, модульное тестирование. Документирование программного кода проекта. Языки и среды автоматизации. Написание простого кода для тестов. Написание модульных тестов. Понятие интегрального тестирования. Документирование программного кода проекта. Виды документации. Стандарты доку-</i>		2		4						8		15

³ Перечень видов учебных занятий уточняется в соответствии с учебным планом.

ментации. Утилиты документирования. Автоматизация документирования.													
<i>Системное тестирование. Дерево проекта. Свойства дерева проекта для DevOps. Создание артефактов проекта для повторного использования кода. Создание артефактов, варианты их повторного использования. Введение в создание дистрибутивов.</i>	2		4							6		15	15
<i>Интеграция со средами коллективной разработки. Сохранение исходных кодов, восстановление зависимостей. Автоматизация создания окружения тестирования программных единиц, находящихся в git. Использование данных для тестирования из git, из файловых систем, по различным протоколам (http, ftp). Использование команд ОС для задания прав на информацию.</i>	2		4							6			
<i>Основные проблемы реального ПО. Нахождение участков кода с утечками памяти. Написание кода с утечками памяти. Признаки переполнения памяти. Средства ОС, которые это показывают. Утечками как признак неверного кода. Тестирование библиотек, работающих с памятью. Встроенные средства в компиляторы. Сторонние утилиты MS Visual Studio, Intel, Bounds Checker. Нахождение участков кода с утечками ресурсов. Средства ОС. Application Verifier. Поиск узких мест производительности, методы повышения производительности.</i>	2		4							6			
<i>Обработка необрабатываемых ошибок. Поиск ошибок в "production". Проблемы чрезмерного потребления ресурсов. Создание дампов памяти средствами ОС, программно, внешними утилитами. Анализ дампов памяти для нахождения проблемного кода. Типовые проблемы: отладочные символы, загрузка символов с сайта MS, рекомендации по созданию собственного сервера символов.</i>	2		4							5			
<i>Введение в использование библиотек GUI и методы тестирования. Кроссплатформенные библиотеки. Библиотека QT5. Библиотека FLTK. Построение графиков. Проблемы нахождения зависимостей внутри графического интерфейса. Отслеживание потоков исполнения, контролируемых пользователем, которые не выполняют никаких интерактивных функций и не перерисовывают элементы интерфейса.</i>	1		2										
Промежуточная аттестация <u>экзамен</u> (указывается форма проведения)	27										X		
Итого	17		34							51		30	30

7 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю) и методические указания для обучающихся по освоению дисциплины (модулю)

Методические указания к практическим занятиям

На занятиях – практические задания по следующим темам:

1. Установка средств разработки, установка системы сборки CMake. Создание кросс-платформенного приложения. Решение типовых задач, связанных с точностью вычислений.
2. Понятие модульности при разработке ПО. Повторное использование кода. Создание простейших тестов. Сохранение результатов выполнения файл для дальнейшего тестирования. Визуализация как отдельная часть тестирования.
3. Проверка качества входных данных. Динамические библиотеки. Многократное использование кода.
4. Автоматизация тестирования. Документирование программного кода проекта.
5. Варианты повторного использования артефактов в CMake. Создание артефактов. Введение в создание дистрибутивов.
6. Интеграция со средами коллективной разработки. Получение данных для тестирования средствами автоматизации сборки.
7. Инструментальные средства нахождения проблемного кода.

Методическое обеспечение инновационных форм учебных занятий

Разбор конкретных ситуаций при решении задач на практических занятиях.

Методические указания для самостоятельной работы обучающихся и прочее

<i>№ п/п</i>	<i>№ раздела дисциплины</i>	<i>Содержание самостоятельной работы</i>	<i>Трудоемкость</i>
1	1-8	ПР-9. Расчетно-графическое задание «составление комплекса тестов по заданному типу проекта»	15
2	1-9	УО2. Реферат на произвольную тему в рамках заданных областей (развитие методологий тестирования ПО, инструменты и платформы тестирования, тестирование ПО платформах)	15

8 Применяемые образовательные технологии для различных видов учебных занятий и для контроля освоения обучающимися запланированных результатов обучения

Перечень обязательных видов учебной работы студента:

- посещение лекционных занятий;
- ответы на теоретические вопросы на практических занятиях;
- решение практических задач и заданий на практических занятиях;
- выполнение реферата
- выполнение расчетно-графического задания

Инновационные формы проведения учебных занятий

Семестр	Вид учебных занятий ⁴	Используемые инновационные формы проведения учебных занятий	Количество академ. часов
VII семестр	Практические занятия	Разбор конкретных ситуаций при решении задач	34
Всего:			34

9 Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю)

- Перечень компетенций выпускников образовательной программы с указанием результатов обучения (знаний, умений, владений), характеризующих этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования.

ОПК-4 - способностью участвовать в настройке и наладке программно-аппаратных комплексов.

ПК-3 - способность обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности.

Полная карта компетенции ОПК-4, ПК-3 приведена в документе «Матрица формирования компетенций» по направлению бакалавриата 09.03.01 Информатика и вычислительная техника»

- Описание шкал оценивания.

При балльно-рейтинговой системе все знания, умения и навыки, приобретаемые студентами в результате изучения дисциплины, оцениваются в баллах.

Оценка качества работы в рейтинговой системе является накопительной и используется для оценивания системной работы студентов в течение всего периода обучения.

По итогам работы в семестре студент может получить максимально **70** баллов. Итоговой формой контроля в VII семестре является экзамен. На экзамене студент может набрать максимально **30** баллов.

В течение VI семестра студент может заработать баллы за следующие виды работ:

№	Вид работы	Сумма баллов
1	Работа на практических занятиях	33
2	Расчетно-графическое задание «составление комплекса тестов по заданному типу проекта» (ПР-9)	10
3	Реферат на произвольную тему в рамках заданных областей (развитие методологий тестирования ПО, инструменты и платформы тестирования, тестирование ПО на современных мобильных и облачных платформах) (УО2)	10
4	Аудиторные занятия (посещение)	17
Итого:		70

Если к моменту окончания семестра студент набирает от **51** до **70** баллов, то он получает допуск к экзамену.

Если студент к моменту окончания семестра набирает от **61** до **70** баллов, то он может получить автоматическую оценку «удовлетворительно». При желании повысить свою оценку, студент имеет право отказаться от автоматической оценки и сдать экзамен.

⁴ Перечень видов учебных занятий уточняется в соответствии с учебным планом.

Если студент не набрал минимального числа баллов (**51** балл), то он не получает допуск к экзамену.

Соответствие рейтинговых баллов и академических оценок

Общая сумма баллов за семестр	Итоговая оценка
86-100	Отлично
71-85	Хорошо
51-70	Допуск к экзамену
в том числе:	
61-70	Возможность получения автоматической оценки «удовлетворительно»
51-60	Только допуск к экзамену
0-50 *	Неудовлетворительно (студент не допущен к экзамену)

Текущий контроль успеваемости осуществляется в процессе выполнения практических и самостоятельных работ в соответствии с ниже приведенным графиком.

График выполнения самостоятельных работ студентами во VII семестре

Виды работ	Недели учебного процесса																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
ПР-9		ВЗ										33					
УО2								ВЗ								33	

ВЗ – выдача задания

33 – защита задания

- Критерии и процедуры оценивания результатов обучения по дисциплине (модулю), характеризующих этапы формирования компетенций

Компетенция **ОПК-4** способностью участвовать в настройке и наладке программно-аппаратных комплексов

код и формулировка компетенции

РЕЗУЛЬТАТ ОБУЧЕНИЯ по дисциплине (модулю) *)	Уровень освоения компетенции**) I - пороговый	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ по дисциплине (модулю) ШКАЛА оценивания <i>(критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)</i>					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		1	2	3	4	5	
32 (ОПК-4) Знать: инструменты для изменения элементов программы во время выполнения, тестирования программы в сложных условиях (многократ-	I - пороговый	Отсутствие знаний	Не знает или слабо знает инструменты для изменения элементов программы во время выполнения, тестирования программы в сложных условиях (многократного запуска, нехватки	Удовлетворительно знает инструменты для изменения элементов программы во время выполнения, тестирования программы	Хорошо знает инструменты для изменения элементов программы во время выполнения, тестирования программы в сложных условиях (многократного запуска, нехватки оперативной	Демонстрирует свободное и уверенное знание инструментов для изменения элементов программы во время выполнения, тестирования программы в сложных условиях (многократ-	Устное собеседование

<p>ного запуска, нехватки оперативной памяти, системной ошибки); методы и средства верификации работоспособности выпусков программных продуктов</p>			<p>оперативной памяти, системной ошибки); методы и средства верификации работоспособности выпусков программных продуктов. Допускает множественные грубые ошибки.</p>	<p>мы в сложных условиях (многократного запуска, нехватки оперативной памяти, системной ошибки); методы и средства верификации работоспособности выпусков программных продуктов. Допускает достаточно серьезные ошибки.</p>	<p>памяти, системной ошибки); методы и средства верификации работоспособности выпусков программных продуктов. Допускает отдельные негрубые ошибки.</p>	<p>ного запуска, нехватки оперативной памяти, системной ошибки); методы и средства верификации работоспособности выпусков программных продуктов. Не допускает ошибок.</p>	
<p><i>У2 (ОПК-4)</i> Уметь: изменять значения элементов программы во время выполнения, тестировать программу со сложными условиями, предугадывать ошибки выполнения, выявлять подверженные ошибкам места программы; составлять условия применения программы</p>	<p>I - пороговый</p>	<p>Отсутствие умений</p>	<p>Демонстрирует частичное умение изменять значения элементов программы во время выполнения, тестировать программу со сложными условиями, предугадывать ошибки выполнения, выявлять подверженные ошибкам места программы; составлять условия применения программы. Допускает множественные грубые ошибки.</p>	<p>Демонстрирует удовлетворительное умение изменять значения элементов программы во время выполнения, тестировать программу со сложными условиями, предугадывать ошибки выполнения, выявлять подверженные ошибкам места программы; составлять условия применения программы,</p>	<p>Демонстрирует достаточно устойчивое умение изменять значения элементов программы во время выполнения, тестировать программу со сложными условиями, предугадывать ошибки выполнения, выявлять подверженные ошибкам места программы; составлять условия применения программы, но допускает отдельные негрубые ошибки.</p>	<p>Демонстрирует устойчивое умение изменять значения элементов программы во время выполнения, тестировать программу со сложными условиями, предугадывать ошибки выполнения, выявлять подверженные ошибкам места программы; составлять условия применения программы, не допускает ошибок.</p>	<p><i>Выполнение практического задания</i></p>

				но допускает достаточно серьезные ошибки.			
<i>B2 (ОПК-4)</i> Владеть: методами определения требуемых от программы характеристик и их достижения; методами разбиения работы над программным проектом на этапы и выделения в них элементов для тестирования программы	I - пороговый	Отсутствие владения	Не владеет или демонстрирует низкий уровень владения методами определения требуемых от программы характеристик и их достижения; методами разбиения работы над программным проектом на этапы и выделения в них элементов для тестирования программы. Допускает многочисленные грубые ошибки.	Демонстрирует удовлетворительный уровень владения методами определения требуемых от программы характеристик и их достижения; методами разбиения работы над программным проектом на этапы и выделения в них элементов для тестирования программы. Допускает достаточно серьезные ошибки.	Демонстрирует хороший уровень владения методами определения требуемых от программы характеристик и их достижения; методами разбиения работы над программным проектом на этапы и выделения в них элементов для тестирования программы. Допускает отдельные негрубые ошибки.	Демонстрирует хороший уровень владения методами определения требуемых от программы характеристик и их достижения; методами разбиения работы над программным проектом на этапы и выделения в них элементов для тестирования программы. Не допускает ошибок.	<i>Выполнение практического задания</i>

Компетенция **ПК-3** - способность обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности.

РЕЗУЛЬТАТ ОБУЧЕНИЯ по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ по дисциплине (модулю) ШКАЛА оценивания <i>(критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)</i>					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		1	2	3	4	5	
Знать (ПК-3): – основы общей теории сложных	I - пороговый	Отсутствие знаний	Не знает или слабо знает основные понятия по основы об-	Удовлетворительно знает основные понятия основы общей теории слож-	Хорошо знает основные понятия основы общей теории сложных си-	Демонстрирует свободное и уверенное знание основных понятий по	<i>Устный опрос</i>

<p>систем, классификацию и закономерности систем, методы и модели описания и анализа систем</p> <p>–</p>			<p>щей теории сложных систем, классификацию и закономерности систем, методы и модели описания и анализа систем</p> <p>Допускает множественные грубые ошибки.</p>	<p>ных систем, классификацию и закономерности систем, методы и модели описания и анализа систем</p> <p>Допускает достаточно серьезные ошибки.</p>	<p>стем, классификацию и закономерности систем, методы и модели описания и анализа систем</p> <p>Допускает отдельные негрубые ошибки.</p>	<p>основы общей теории сложных систем, классификацию и закономерности систем, методы и модели описания и анализа систем</p> <p>Не допускает ошибок.</p>	
<p>Уметь (ПК-3):</p> <p>– использовать полученные теоретические знания: для получения, хранения, переработки информации; при решении различных задач с использованием специализированных программ</p>	<p>I - пороговый</p>	<p>Отсутствие умений</p>	<p>Демонстрирует частичное умение использовать полученные теоретические знания: для получения, хранения, переработки информации; при решении различных задач с использованием специализированных программ</p> <p>Допускает множественные грубые ошибки.</p>	<p>Демонстрирует удовлетворительное умение использовать полученные теоретические знания: для получения, хранения, переработки информации; при решении различных задач с использованием специализированных программ</p> <p>Допускает достаточно серьезные ошибки.</p>	<p>Демонстрирует достаточно устойчивое умение использовать полученные теоретические знания: для получения, хранения, переработки информации; при решении различных задач с использованием специализированных программ</p> <p>Допускает отдельные негрубые ошибки.</p>	<p>Демонстрирует устойчивое умение использовать полученные теоретические знания: для получения, хранения, переработки информации; при решении различных задач с использованием специализированных программ</p> <p>Не допускает ошибок.</p>	<p><i>Выполнение практического задания</i></p>
<p>Владеть (ПК-3):</p> <p>– навыками применения современных технических средств и информационных технологий для решения задач</p>	<p>I - пороговый</p>	<p>Отсутствие владения</p>	<p>Не владеет или демонстрирует низкий уровень владения навыками применения современных технических средств и информационных технологий для решения задач</p> <p>Допускает множе-</p>	<p>Демонстрирует удовлетворительный уровень владения навыками применения современных технических средств и информационных технологий для решения задач.</p> <p>Допускает достаточно серьезные ошибки.</p>	<p>Демонстрирует хороший уровень владения навыками применения современных технических средств и информационных технологий для решения задач</p> <p>Допускает отдельные негрубые ошибки.</p>	<p>Демонстрирует высокий уровень владения навыками применения современных технических средств и информационных технологий для решения задач</p> <p>Не допускает ошибок.</p>	<p><i>Выполнение практического задания</i></p>

			ственные грубые ошибки.				
--	--	--	-------------------------------	--	--	--	--

- Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения, характеризующих этапы формирования компетенций и (или) для итогового контроля сформированности компетенции.

Список вопросов к экзамену

1. Взаимосвязь разработки программного кода и его тестирования.
2. С++ и кросс-платформенная разработка, варианты установки средств разработки, сборки и тестирования.
3. Методы тестирования программных продуктов.
4. Точность вычислений, погрешности вычислений и тестирование.
5. Понятие «чистого» кода. Повторное использование кода.
6. Понятие защищенного кода. Риски от использования незащищенных компонентов.
7. Динамические библиотеки, версионность, многократное использование кода.
8. Автоматизация тестирования. Языки и среды автоматизации.
9. Документирование программного кода проекта. Виды документации.
10. Стандарты документации. Утилиты документирования. Автоматизация документирования.
11. Дерево проекта. Свойства дерева проекта для DevOps.
12. Создание артефактов проекта для повторного использования кода.
13. Интеграция со средами коллективной разработки. Сохранение исходных кодов, восстановление зависимостей.
14. Использование данных для тестирования из git, из файловых систем, по различным протоколам (http, ftp). Использование команд ОС для задания прав на информацию.
15. Утечки памяти как признак неверного кода. Признаки переполнения памяти. Средства ОС, которые это показывают. Нахождение участков кода с утечками памяти, встроенными в компилятор средствами.
16. Нахождение участков кода с утечками памяти при помощи сторонних программных средств.
17. Нахождение участков кода с утечками ресурсов, средствами ОС, также при помощи Application Verifier.
18. Поиск ошибок в "production". Обработка необрабатываемых ошибок. Проблемы чрезмерного потребления ресурсов. Создание дампов памяти средствами ОС, программно, внешними утилитами.
19. Средства для анализа дампов памяти для нахождения проблемного кода. Типовые проблемы: отладочные символы, загрузка символов с сайта MS, рекомендации по созданию собственного сервера символов.
20. Виды кроссплатформенных библиотек графических библиотек.
21. Тестирование графических интерфейсов.

Варианты расчетно-графического задания «составление комплекса тестов по заданному типу проекта» (ПР-9)

1. Функциональные возможности
2. Надежность
3. Практичность
4. Эффективность

5. Сопровождаемость
6. Мобильность

Варианты реферата на произвольную тему в рамках заданных областей (развитие методологий тестирования ПО, инструменты и платформы тестирования, тестирование ПО на современных платформах) (УО2)

1. Методы тестирования программных продуктов
2. Зависимость процесса тестирования от поколений аппаратного и программного обеспечения
3. Влияние парадигм на методологии тестирования
4. Понятие «чистого» кода. Методики приведения кода к нормальному виду. Именованные классы, методов и атрибутов
5. Умолчания и договоренности при кодировании. Их влияние на коллективную разработку программного обеспечения
6. Понятие «защищенного» кода. Риски от использования незащищенных компонентов
7. Аудит безопасности. «Защищенное» кодирование применительно к многопоточности. Атомарные транзакции и обработка ошибок
8. Понятие анализа исходного кода. Методики анализа
9. Анализ кода как неотъемлемая часть написания программы
10. Анализ кода в различных парадигмах и платформах
11. Типы инструментальных средств отладки
12. Инструменты, встроенные в интегрированные среды разработки (IDE)
13. Независимые инструменты профилирования и контроля защищенности программного кода
14. Типы тестов: тестирование «черного ящика», «белого ящика» и «серого ящика»
15. Планирование модульных тестов
16. Контрольные таблицы и подходы к тестированию методов и классов
17. Средства автоматизации модульного тестирования
18. Процесс интеграции программной системы
19. Стандарты документов по интеграции
20. Параметры качества при интеграции, их влияние на процесс тестирования. Персонал, участвующий в интеграции
21. Понятие канонической машины
22. Типы системных тестов. Метрики системного тестирования
23. Инспектирование системных тестов
24. Системная интеграция и модель СММ
25. Альфа-бета версии программного продукта
26. Типы инструментальных средств по поддержке процесса тестирования
27. Инструменты тестирования каналов связи, вычислительной нагрузки, пользовательских интерфейсов
28. Экспертиза программного обеспечения – ее составляющие
29. Автоматизированные платформы тестирования, их взаимодействие с системами управления проектом и с системами управления версиями программного кода
30. Концептуальные отличия технологий тестирования объектно-ориентированного программного обеспечения
31. Тестирование аналитических и проектных модулей
32. Тестирования иерархии наследования, классов и взаимодействий между объектами
33. Тестирование распределенных объектов
34. Методы отделения ошибок среды от ошибок приложения
35. Ошибки кодировки и проблемы конфигурации
36. Тестирование комплексов: «клиент – web-сервер – БД», «клиент – web-сервер – сервер приложений»

37. От канонической машины к каноническому комплексному решению для тестирования web-порталов
38. Инструменты web-диагностики
39. Влияние новых платформ и парадигм на методологии тестирования
40. Тестирование мобильных платформ и «облачных» решений

– Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Процедура промежуточной аттестации проходит в соответствии с «Положением балльно-рейтинговой системе оценки и текущем контроле успеваемости студентов», а также «Положением о промежуточной аттестации» университета «Дубна».

10 Ресурсное обеспечение

• Перечень основной и дополнительной учебной литературы

Основная учебная литература

1. Черников, Б. В. Управление качеством программного обеспечения : учебник / Б.В. Черников. — Москва : ИД «ФОРУМ» : ИНФРА-М, 2019. — 240 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-16-102705-9. - Текст : электронный. // ЭБС "Znanium.com" [сайт]. - URL: <https://new.znanium.com/catalog/product/1018037> (дата обращения: 07.04.2021). Режим доступа: ограниченный по логину и паролю
2. Мартин, Р. Чистый код. Создание, анализ и рефакторинг : Библиотека программиста / Р. Мартин. - СПб : Питер, 2016. - 464с. : ил. - ISBN 978-5-496-00487-9.
3. Гагарина, Л. Г. Технология разработки программного обеспечения : учеб. пособие / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Сидорова-Виснадул ; под ред. Л.Г. Гагариной. — Москва : ИД «ФОРУМ» : ИНФРА-М, 2019. — 400 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-16-104071-3. - Текст : электронный. // ЭБС "Znanium.com" [сайт]. - URL: <https://new.znanium.com/catalog/product/1011120> (дата обращения: 29.04.2021) . Режим доступа: ограниченный по логину и паролю
Гагарина, Л.Г. Технология разработки программного обеспечения: учебное пособие. / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Виснадул, под. Ред. Л.Г. Гагариной. – М.: ИД «Форум»: ИНФРА-М, – 2009. – 400 с.: ил. – (Высшее образование).
4. Мартишин, С. А. Основы теории надежности информационных систем : учебное пособие / С.А. Мартишин, В.Л. Симонов, М.В. Храпченко. — Москва : ИД «ФОРУМ» : ИНФРА-М, 2020. — 255 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-16-106294-4. - Текст : электронный. // ЭБС "Znanium.com" [сайт]. - URL: <https://new.znanium.com/catalog/product/1062374> (дата обращения: 07.05.2021). Режим доступа: ограниченный по логину и паролю

Дополнительная учебная литература

1. Ананьева, Т. Н. Стандартизация, сертификация и управление качеством программного обеспечения : учеб. пособие / Т.Н. Ананьева, Н.Г. Новикова, Г.Н. Исаев. — Москва : ИНФРА-М, 2019. — 232 с. — (Высшее образование: Бакалавриат). — ISBN 978-5-16-104095-9. - Текст : электронный. // ЭБС "Znanium.com" [сайт].- URL: <https://new.znanium.com/catalog/product/1002357> (дата обращения: 07.04.2021). Режим доступа: ограниченный по логину и паролю
2. Бахвалов, Н.С. Численные методы: анализ, алгебра, обыкновенные дифференциальные уравнения / Н.С. Бахвалов ; ред. И.М. Овчинникова, Е.В. Шикин. - Москва : Наука, 1975. - 632 с. : ил. Текст : электронный. // ЭБС "Университетская библиотека онлайн" [сайт]. – URL: <http://biblioclub.ru/index.php?page=book&id=456941> (дата обращения: 07.04.2021). - Режим доступа: ограниченный по логину и паролю
3. Дубров, Д.В. Система построения проектов CMake : учебник / Д.В. Дубров. - Ростов-на-Дону : Издательство Южного федерального университета, 2015. - 419 с. : табл., ил. - Библиогр.: с. 407- 408 – ISBN 978-5-9275-1852-4. – Текст : электронный. // ЭБС "Универси-

- тетская библиотека онлайн" [сайт]. – URL: <http://biblioclub.ru/index.php?page=book&id=461879> (дата обращения: 07.04.2021). - Режим доступа: ограниченный по логину и паролю
4. Калиткин, Н.Н. Численные методы : Учебное пособие / Н.Н. Калиткин ; под ред. А.А. Самарского. - Москва : Наука, 1978. - 512 с. : ил.– Текст : электронный. // ЭБС "Университетская библиотека онлайн" [сайт]. – URL: <http://biblioclub.ru/index.php?page=book&id=456957> (дата обращения: 07.04.2021). - Режим доступа: ограниченный по логину и паролю
5. Страуструп, Б. Язык программирования С++ для профессионалов / Б. Страуструп. - Москва : Интернет-Университет Информационных Технологий, 2006. - 568 с. Текст : электронный. // ЭБС "Университетская библиотека онлайн" [сайт]. – URL: <http://biblioclub.ru/index.php?page=book&id=234816> (дата обращения: 07.04.2021). - Режим доступа: ограниченный по логину и паролю

• **Периодические издания**

1. Информационные технологии и вычислительные системы: научный журнал / Учредитель Федеральное государственное учреждение "Федеральный исследовательский центр "Информатика и управление" РАН; гл. ред. Попков Ю.С. - М.: ФГУ Федеральный исследовательский центр "Информатика и управление" РАН. – Журнал выходит 2 раза в полуг. – Основан в 1995 г. - ISSN 2071-8632. – Текст : электронный. Полные электронные версии статей журнала доступны по подписке на сайте научной электронной библиотеки «eLIBRARY.RU»: https://www.elibrary.ru/title_about_new.asp?id=8746
2. Информация и безопасность: научный журнал / Учредители: Воронежский государственный технический университет; гл. ред. Остапенко А.Г. – Воронеж: Воронежский государственный технический университет. – Журнал выходит 2 раза в полуг. - Основан в 1998 году. - ISSN 1682-7813. – Текст : электронный. Полные электронные версии статей журнала доступны на сайте научной электронной библиотеки «eLIBRARY.RU»: <http://elibrary.ru/contents.asp?titleid=8748>
3. Информатика и системы управления: научное издание / Учредитель: Амурский государственный университет; гл. ред. Е.Л. Еремин. – Благовещенск: Амурский государственный университет. – журнал выходит 2 раза в полуг. - Основан в 2001 г. – ISSN: 1814-2400. - Текст : электронный. Полные электронные версии статей журнала доступны на сайте научной электронной библиотеки «eLIBRARY.RU»: <https://www.elibrary.ru/contents.asp?titleid=9793>
4. Открытые системы СУБД / Учредитель: ООО «Издательство «Открытые системы»; гл. ред. Д. Волков. – М.: Издательство «Открытые системы». – журнал выходит 2 раза в полуг. - Основан в 1993 году. – ISSN: 1028-7493. – Текст : электронный. – Полные электронные версии статей представлены на сайте журнала: <https://www.osr.ru/os/archive>
5. Программные продукты и системы: международный научно-практический журнал / Учредитель: Куприянов В.П.; гл. ред. Савин Г.И. - Тверь: Центрпрограммсистем. – журнал выходит 2 раза в полуг. - Основан в 1988 году. – ISSN: 0236-235X. - – Текст : электронный. – Полные электронные версии статей представлены на сайте журнала: <http://swsys.ru/>
6. Системный администратор / Учредитель: "Издательский дом "Положевец и партнеры"; гл. ред. Г. Положевец. – М.: Общество с ограниченной ответственностью "Издательский дом "Положевец и партнеры". – Журнал выходит 12 раз в год. - Основан в 2002 году. - ISSN 1813-5579. – Текст : электронный. Полные электронные версии статей журнала доступны по подписке на сайте научной электронной библиотеки «eLIBRARY.RU»: https://elibrary.ru/title_about.asp?id=9973
7. Вестник Московского университета. Серия 15. Вычислительная математика и кибернетика: научный журнал / Учредитель: МГУ им. М.В. Ломоносова; гл. ред. академик РАН Моисеев Е.И. – М.: ФГБОУ ВО МГУ им. М.В. Ломоносова – Журнал

выходит 2 раза в полуг. - Основан в 1977 году. – ISSN 0137-0782. – Текст : электронный.
– Полные электронные версии статей журнала доступны по подписке на сайте научной электронной библиотеки «eLIBRARY.RU»: https://www.elibrary.ru/title_about.asp?id=8373

• **Перечень ресурсов информационно-телекоммуникационной сети «Интернет»**
Электронно-библиотечные системы и базы данных

1. ЭБС «Znaniium.com»: <http://znaniium.com/>
2. ЭБС «Лань»: <https://e.lanbook.com/>
3. ЭБС «Юрайт»: <https://biblio-online.ru/>
4. ЭБС «Университетская библиотека онлайн»: <http://biblioclub.ru/>
5. Научная электронная библиотека (РУНЭБ) «eLIBRARY.RU»: <http://elibrary.ru>
6. Национальная электронная библиотека (НЭБ): <http://нэб.рф/>
7. Базы данных российских журналов компании «East View»: <https://dlib.eastview.com/>

Научные поисковые системы

1. ArXiv.org - научно-поисковая система, специализируется в областях: компьютерных наук, астрофизики, физики, математики, квантовой биологии. <http://arxiv.org/>
2. Google Scholar - поисковая система по научной литературе. Включает статьи крупных научных издательств, архивы препринтов, публикации на сайтах университетов, научных обществ и других научных организаций. <https://scholar.google.ru/>
3. WorldWideScience.org - глобальная научная поисковая система, которая осуществляет поиск информации по национальным и международным научным базам данных и порталам. <http://worldwidescience.org/>
4. SciGuide - навигатор по зарубежным научным электронным ресурсам открытого доступа. <http://www.prometeus.nsc.ru/sciguide/page0601.ssi>

Профессиональные ресурсы сети «Интернет»

1. Федеральная информационная система «Единое окно доступа к информационным ресурсам»: <http://window.edu.ru/>.
2. Проект Инициативного Народного Фронта Образования - ИНФО-проект. Школа программирования Coding Craft <http://codingcraft.ru/>.
3. Портал Life-prog <http://life-prog.ru/>.
4. OpenNet www.opennet.ru.
5. Алгоритмы, методы, программы algotlist.manual.ru.
6. Сервер министерства высшего образования www.informika.ru.

• **Перечень информационных технологий, используемых при осуществлении образовательного процесса, включая программное обеспечение, информационные справочные системы (при необходимости)**

Проведение лекционных занятий предполагает использование комплектов слайдов и программных презентаций по рассматриваемым темам.

Проведение практических занятий по дисциплине предполагается использование специализированных аудиторий, оснащенных персональными компьютерами, объединенными в локальную сеть и имеющих доступ к ресурсам глобальной сети Интернет.

Для выполнения заданий самостоятельной подготовки обучающиеся обеспечиваются литературой, а также в определенном порядке могут получать доступ к информационным ресурсам Интернета.

Подписка на Microsoft DreamSpark Premium. Дает возможность скачивания программного обеспечения и установки его в Филиале для студентов и преподавателей для некоммерческих целей. Число мест: не ограничено. Программы для ЭВМ DreamSpark Premium Electronic Software Delivery Renewal (MS Imagine Premium, договор Tr000104809/м18 от

01.09.2016 г.). Дисциплина обеспечена необходимым программным обеспечением, которое находится в свободном доступе (программы Open office, свободная лицензия, код доступа не требуется).

- **Описание материально-технической базы**

Компьютерный класс (15 ПК): оборудование в собственности

11 Язык преподавания

Русский