

**МИНОБРНАУКИ РОССИИ**  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ  
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»**  
**(БГТУ им. В.Г. Шухова)**

УТВЕРЖДАЮ  
Директор института энергетики,  
информационных технологий и  
управляющих систем  
Белоусов А.В.  
« 20 » мая 2021 г.



**РАБОЧАЯ ПРОГРАММА**  
**дисциплины**

**Технологии и методы программирования систем реального времени**

направление подготовки:

10.05.03 Информационная безопасность автоматизированных систем

Специализация программы:

Безопасность открытых информационных систем

Квалификация

Специалист по защите информации

Форма обучения

очная

Институт энергетики, информационных технологий и управляющих систем

Кафедра Программного обеспечения вычислительной техники и  
автоматизированных систем

Белгород 2021

Рабочая программа составлена на основании требований:

- Федерального государственного образовательного стандарта высшего образования – специалитет по специальности 10.05.03 Информационная безопасность автоматизированных систем, утвержденного приказом Минобрнауки России от 26.11.2020 №1457
- учебного плана, утвержденного ученым советом БГТУ им. В.Г. Шухова в 2021 году.

Составитель: ст. преподаватель \_\_\_\_\_ (Лукьянов А.М.)  
(ученая степень и звание, подпись) \_\_\_\_\_ (инициалы, фамилия)

Рабочая программа обсуждена на заседании кафедры

« 14 » \_\_\_\_\_ 05 \_\_\_\_\_ 2021 г., протокол № \_\_\_\_\_ 8 \_\_\_\_\_

Заведующий кафедрой: к.т.н., доцент \_\_\_\_\_ (Поляков В.М.)  
(ученая степень и звание, подпись) \_\_\_\_\_ (инициалы, фамилия)

Рабочая программа согласована с выпускающей кафедрой программного обеспечения вычислительной техники и автоматизированных систем  
(наименование кафедры/кафедр)

Заведующий кафедрой: к.т.н., доцент \_\_\_\_\_ (Поляков В.М.)  
(ученая степень и звание, подпись) \_\_\_\_\_ (инициалы, фамилия)

« 14 » \_\_\_\_\_ 05 \_\_\_\_\_ 2021 г.

Рабочая программа одобрена методической комиссией института

« 20 » \_\_\_\_\_ 05 \_\_\_\_\_ 2021 г., протокол № \_\_\_\_\_ 9 \_\_\_\_\_

Председатель к.т.н., доцент \_\_\_\_\_ (Семернин А.Н.)  
(ученая степень и звание, подпись) \_\_\_\_\_ (инициалы, фамилия)

## 1. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Категория (группа) компетенций	Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Наименование показателя оценивания результата обучения по дисциплине
	ПК-2. Способен разрабатывать системы защиты информации автоматизированных систем реального времени (проектный)	ПК-2.1. Осуществляет выявление уязвимостей информационно-технологических ресурсов автоматизированных систем	<p>Знать:</p> <ul style="list-style-type: none"> <li>-теоретические основы построения и функционирования вычислительных средств;</li> <li>-классификацию видов уязвимостей информационно-технологических ресурсов;</li> <li>-базовые возможности языков программирования, применимых для поиска уязвимостей в системах реального времени.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>-разрабатывать алгоритмы и программы для решения задач поиска уязвимостей.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>-современными технологиями анализа информационно-технологических ресурсов на предмет их уязвимости.</li> </ul>
		ПК-2.2. Выполняет проектирование и реализацию систем защиты информации автоматизированных систем реального времени	<p>Знать:</p> <ul style="list-style-type: none"> <li>-принципы объектно ориентированного программирования;</li> <li>- виды осуществляемых атак и основные способы противодействия</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>-применять методы решения типовых задач при проектировании программных средств защиты информации автоматизированных систем;</li> <li>-разрабатывать программы, оптимизированные по быстродействию и занимаемой памяти.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>-навыками использования различных инструментальных средств для защиты информации автоматизированных систем;</li> <li>- навыками проектирования и реализации систем защиты информации автоматизированных систем реального времени.</li> </ul>

Категория (группа) компетенций	Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Наименование показателя оценивания результата обучения по дисциплине
		ПК-2.3. Применяет прогрессивные технические решения при разработке систем защиты информации автоматизированных систем реального времени	<p>Знать:</p> <ul style="list-style-type: none"> <li>–перспективы развития информационно-технологических ресурсов автоматизированных систем;</li> <li>–направления развития систем защиты информации автоматизированных систем реального времени.</li> </ul> <p>Уметь:</p> <ul style="list-style-type: none"> <li>–разрабатывать алгоритмы и проектировать программные средства обнаружения атак на информационно-технологические ресурсы;</li> <li>–анализировать доступный инструментарий защиты информации и делать обоснованный выбор для использования.</li> </ul> <p>Владеть:</p> <ul style="list-style-type: none"> <li>–навыками выполнения проектов по реализации профессиональных задач;</li> <li>–навыками выработки рекомендаций по использованию систем защиты информации.</li> </ul>

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

### 2.1. Компетенция ПК-2. Способен разрабатывать системы защиты информации автоматизированных систем реального времени (проектный).

Данная компетенция формируется следующими дисциплинами.

Стадия	Наименования дисциплины
1.	Технологии и методы программирования систем реального времени
2.	Средства защиты от разрушающих программных компонентов
3.	Контроль и тестирование программно-аппаратных систем
4.	Технология построения защищенных распределенных приложений
5.	Практикум по подготовке инженерной документации
6.	Программирование встраиваемых систем
7.	Системы и среды программирования
8.	Производственная проектно-технологическая практика
9.	Производственная преддипломная практика
10.	Подготовка к процедуре защиты и защита выпускной квалификационной работы

## 3. ОБЪЁМ ДИСЦИПЛИНЫ

3.1. Общая трудоемкость дисциплины составляет 4 зач. единиц, 144 часа.

Форма промежуточной аттестации экзамен.

Вид учебной работы	Всего часов	Семестр № 7
1	2	3
<b>Общая трудоемкость дисциплины, час</b>	<b>144</b>	<b>144</b>
<b>Контактная работа (аудиторные занятия), в т.ч.:</b>	<b>73</b>	<b>73</b>
лекции	34	34

1	2	3
лабораторные	34	34
практические		
групповые консультации в период теоретического обучения и промежуточной аттестации	5	5
<b>Самостоятельная работа студентов, включая индивидуальные и групповые консультации, в том числе:</b>	<b>71</b>	<b>71</b>
Курсовой проект		
Курсовая работа		
Расчетно-графическое задание		
Индивидуальное домашнее задание		
Самостоятельная работа на подготовку к аудиторным занятиям (лекции, практические занятия, лабораторные занятия)	35	35
Экзамен	экзамен 36	экзамен 36

#### 4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

##### 4.1. Наименование тем, их содержание и объем

№ п/п	Наименование раздела (модуля)	Объем на тематический раздел по видам учебной нагрузки, час			
		Лекции	Практические занятия	Лабораторные занятия	Самостоятельная работа на подготовку к аудиторным занятиям
1	2	3	4	5	6
семестр № 7					
1.	<b>Введение.</b> Содержание предмета, цели и задачи курса. Процесс производства ПО: методы, технология и инструментальные средства. Технология разработки ПО и основные этапы ее развития.	2	-	-	1
2.	<b>Качество ПО.</b> Проблемы разработки сложных программных систем. Метрология ПО. Критерии качества ПО: сложность, корректность, надежность, трудоемкость. Оценка качества ПО.	2	-	2	2
3.	<b>Жизненный цикл ПО.</b> Жизненный цикл и этапы разработки программного обеспечения. Эволюция моделей жизненного цикла. Гибкие методологии разработки ПО. Scrum, Lean-методологии. Технологический цикл разработки ПО. Оценка качества процессов создания ПО	4	-	6	6

1	2	3	4	5	6
4.	<b>Приемы обеспечения технологичности ПО.</b> Понятие технологичности ПО. Модули и их свойства. Нисходящая и восходящая разработка ПО. Структурное и неструктурное программирование. Стиль оформления программы. Способы обеспечения эффективности программ. Программирование с защитой от ошибок..	4	-	4	6
5.	<b>Определение требований к ПО.</b> Определение требований к ПО и исходных данных для его проектирования. Основные эксплуатационные требования к ПО. Предварительные проектные исследования предметной области. Разработка технического задания..	2	-	6	6
6.	<b>Проектирование ПО при структурном подходе.</b> Структурный подход к специфицированию и проектированию ПО. Функциональные диаграммы. Диаграммы потоков данных. Диаграммы переходов состояний. Структуры данных и диаграммы отношений компонентов данных: Диаграммы Джексона, Диаграммы Орра, диаграммы сущность-связь. Разработка структурной и функциональной схем. Использование метода пошаговой детализации для проектирования структуры ПО. Структурные карты Константайна..	4	-	8	6
7.	<b>Проектирование ПО при объектно-ориентированном подходе.</b> Объектно-ориентированный подход к специфицированию и проектированию ПО. Унифицированный язык моделирования UML. Диаграммы вариантов использования. Диаграммы классов. Диаграммы взаимодействия. Диаграммы деятельности. Диаграммы компонентов. Диаграммы размещения. Паттерны проектирования	4	-	6	4
8.	<b>Тестирование и отладка ПО.</b> Тестирование и отладка. Общая схема отладки. Классификация ошибок. Методы диагностики и локализации ошибок. Принципы и методы тестирования. Статическое тестирование. Структурное и функциональное тестирование. Организация процесса тестирования. Тестирование модулей и комплексное тестирование. Критерии завершения тестирования. Оценочное тестирование..	8	-	-	1
9.	<b>Коллективная работа по созданию программ.</b> Требования к проекту. Сообщения в проекте. Задачи. Ошибки. Сборка проектов. Риски. Проблемы. Инструментальные средства коллективной разработки программного обеспечения.	2	-	2	2
10.	<b>Составление программной документации.</b> Единая система программной документации. Виды программных документов. Пояснительная записка. Руководство оператора (пользователя). Руководство программиста и системного программиста. Основные правила оформления программной документации.,	2	-	-	1
	<b>ВСЕГО</b>	34		34	35

#### 4.2. Содержание практических (семинарских) занятий.

Не предусмотрено учебным планом

#### 4.3. Содержание лабораторных занятий

№ п/п	Наименование раздела дисциплины	Тема лабораторной работы	К-во часов	Самостоятельная работа на подготовку к аудиторным занятиям
семестр № 7				
1.	<b>Качество ПО. Жизненный цикл ПО.</b>	Описание предметной области с помощью диаграмм UML	8	6
2.	<b>Приемы обеспечения технологичности ПО. Определение требований к ПО</b>	Требования к программному обеспечению	10	11
3.	<b>Проектирование ПО при структурном подходе.</b>	Шаблоны проектирования Проектирование программных систем, используя шаблоны	8	5
4.	<b>Проектирование ПО при объектно-ориентированном подходе.</b>	Разработка программного продукта. Обработка ошибок в программных системах Тестирование программного обеспечения	6	3
5.	<b>Коллективная работа по созданию программ.</b>	Работа с исходным кодом ПО. Системы контроля версий.	2	1
ИТОГО:			34	26
ВСЕГО:				60

#### 4.4. Содержание курсового проекта/работы

Не предусмотрено учебным планом

#### 4.5. Содержание расчетно-графического задания, индивидуальных домашних заданий

Не предусмотрено учебным планом

### 5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

#### 5.1. Реализация компетенций

**Компетенция ПК-2.** Способен разрабатывать системы защиты информации автоматизированных систем реального времени (проектный).

Наименование индикатора достижения компетенции	Используемые средства оценивания
ПК-2.1. Осуществляет выявление уязвимостей информационно-технологических ресурсов автоматизированных систем	Устный опрос, выполнение лабораторных работ
ПК-2.2. Выполняет проектирование и реализацию систем защиты информации автоматизированных систем реального времени	Устный опрос, выполнение лабораторных работ

ПК-2.3. Применяет прогрессивные технические решения при разработке систем защиты информации автоматизированных систем реального времени	Устный опрос, выполнение лабораторных работ, экзамен
---	--

## 5.2. Типовые контрольные задания для промежуточной аттестации

### 5.2.1. Перечень контрольных вопросов (типовых заданий)

#### для экзамена

№ п/п	Наименование раздела дисциплины	Содержание вопросов (типовых заданий)
1	2	3
1.	<b>Качество ПО. Жизненный цикл ПО.</b>	<p>Объектно-ориентированный анализ и проектирование программных систем. Основные определения.</p> <p>Основные принципы объектно-ориентированной разработки программ.</p> <p>Обязанности объектов. Разложение системы на объекты. CRC-карты.</p> <p>Инкапсуляция. Связность внутри классов и зацепление между классами.</p> <p>Композиция и наследование. Абстрактные классы. Интерфейс класса. Рекомендации.</p> <p>Методы, операции, сообщения. Разделение команд и запросов.</p> <p>Диаграмма классов UML. Система обозначений. Отношения между классами.</p>
2.	<b>Приемы обеспечения технологичности ПО.</b>	<p>Технология разработки программного обеспечения. Определение. Основные этапы на примере классического жизненного цикла.</p> <p>Два взгляда на программное обеспечение: научная разработка, программное изделие.</p>
3.	<b>Определение требований к ПО.</b>	<p>Функциональные требования.</p> <p>Нефункциональные требования.</p> <p>Требования предметной области.</p> <p>Пользовательские требования.</p> <p>Системные требования.</p> <p>Спецификация требований. Виды требований.</p>
4.	<b>Проектирование ПО при структурном подходе. Проектирование ПО при объектно-ориентированном подходе.</b>	<p>Парадигмы проектирования программных систем. Макетирование.</p> <p>Парадигмы проектирования программных систем. Инкрементная модель.</p> <p>Парадигмы проектирования программных систем. Быстрая разработка приложений.</p> <p>Парадигмы проектирования программных систем. Спиральная модель.</p> <p>Парадигмы проектирования программных систем. Компонентно-ориентированная модель.</p> <p>Парадигмы проектирования программных систем. Унифицированный процесс. RUP.</p> <p>Парадигмы проектирования программных систем. Экстремальное программирование.</p>

1	2	3
		Прецеденты. Определение. Актеры. Сценарии. Задачи и рамки прецедентов. Степень формализации прецедентов. Сжатый, свободный и развёрнутый формат описания. Пояснения к прецедентам. Предусловия и постусловия. Альтернативные сценарии. Диаграмма прецедентов. Система обозначений UML. Отношения между прецедентами. Проектирование по контракту. Предусловия и постусловия в методах. Инварианты. Паттерны проектирования. Определение. Формат описания. Виды паттернов по уровню абстракции и по цели. Примеры.
5.	<b>Тестирование и отладка ПО.</b>	Разработка через тестирование. Минимальное грубое тестирование. Тестирование. Основные определения. Виды. Тестирование черного ящика. Тестирование белого ящика. Автоматизированные тесты. Основные определения. XUnit. Системы управления версиями. Определения. Использование в проектах.
6.	<b>Коллективная работа по созданию программ. Составление программной документации.</b>	Командная разработка. Парное программирование. SCRUM. Рефакторинг. Цели. Описание рефакторингов. Примеры. Кодирование. Стандарты на кодирование. Кодирование и проектирование. Описание технического задания по ГОСТ.

### 5.3. Типовые контрольные задания (материалы) для текущего контроля в семестре

*Текущий контроль* осуществляется в течение семестра в форме выполнения и защиты лабораторных работ.

Защита лабораторных работ проводится в форме устного опроса студента по выполненной работе и направлена на проверку степени усвоения материала и понимания теоретических сведений, используемых в процессе выполнения работы.

После изучения каждой темы раздела для закрепления изученного материала проводится **тестирование**. Тестирование проходит с использованием системы MyTest. Время выполнения заданий теста составляет 15 минут.

### Тестовые задания по темам

№ п/п	Наименование раздела дисциплины	Содержание вопросов (типовых заданий)
1.	<b>Введение.</b>	<p>Какие языки относятся к объектно-ориентированным? Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ Python;</li> <li>➤ HTML;</li> <li>➤ C++(C#);</li> <li>➤ Java;</li> <li>➤ Ассемблер.</li> </ul> <hr/> <p>Какой язык относится к языкам низкого уровня? Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ Ассемблер;</li> <li>➤ Python;</li> <li>➤ Haskell;</li> <li>➤ C#;</li> <li>➤ PHP.</li> </ul> <hr/> <p>Каждый объект графической оболочки имеет: Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ Контейнер;</li> <li>➤ собственный определенный набор свойств;</li> <li>➤ реакцию на события;</li> <li>➤ уникальный идентификатор;</li> <li>➤ уникальный вид.</li> </ul>
2.	<b>Качество ПО.</b>	<p>Качество ПС отражается тремя группами показателей, характеризующими. Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ внутреннее, внешнее, качество при использовании;</li> <li>➤ требуемое, обусловленное, реальное;</li> <li>➤ номинальное, идеальное, реальное;</li> <li>➤ определенное, достигнутое, недостигнутое.</li> </ul> <hr/> <p>Какие факторы влияют на степень качества программного средства? Выберите варианты из списка</p> <ul style="list-style-type: none"> <li>➤ качество технологий проектирования;</li> <li>➤ качество разработки ПС;</li> <li>➤ качество сопровождения;</li> <li>➤ качество документирования.</li> </ul>
3.	<b>Жизненный цикл ПО.</b>	<p>Установите последовательность этапов жизненного цикла информационной системы:</p> <ul style="list-style-type: none"> <li>➤ тестирование;</li> <li>➤ эксплуатация;</li> <li>➤ сопровождение;</li> <li>➤ технический проект;</li> <li>➤ концепция;</li> <li>➤ техническое задание;</li> <li>➤ формирование требований.</li> </ul>

№ п/п	Наименование раздела дисциплины	Содержание вопросов (типовых заданий)
		<p>Первый этап в жизненном цикле программы. Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ формулирование требований;</li> <li>➤ анализ требований;</li> <li>➤ проектирование;</li> <li>➤ автономное тестирование;</li> <li>➤ комплексное тестирование.</li> </ul> <p>Самый большой этап в жизненном цикле программы. Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ эксплуатация;</li> <li>➤ изучение предметной области;</li> <li>➤ программирование;</li> <li>➤ тестирование;</li> <li>➤ корректировка ошибок.</li> </ul>
4.	<b>Приемы обеспечения технологичности ПО.</b>	<p>Какие существуют методы программирования. Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ логическое;</li> <li>➤ структурное;</li> <li>➤ модульное;</li> <li>➤ компиляторное;</li> <li>➤ линейное.</li> </ul> <p>Достоинство структурного программирования Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ можно приступить к комплексному тестированию на раннем этапе разработки;</li> <li>➤ можно приступить к автономному тестированию на раннем этапе разработки;</li> <li>➤ нет необходимости выполнять тестирование;</li> <li>➤ можно пренебречь отладкой.</li> </ul> <p>Недостаток структурного программирования Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ увеличивает размер программы;</li> <li>➤ снижает эффективность;</li> <li>➤ уменьшает количество ошибок;</li> <li>➤ не требует отладки.</li> </ul>
5.	<b>Определение требований к ПО.</b>	<p>Основные эксплуатационные требования к разрабатываемому программному обеспечению. Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ правильность;</li> <li>➤ универсальность;</li> <li>➤ надежность (помехозащищенность);</li> <li>➤ проверяемость;</li> <li>➤ точность результатов;</li> <li>➤ защищенность;</li> <li>➤ программная совместимость;</li> <li>➤ аппаратная совместимость;</li> <li>➤ эффективность;</li> <li>➤ адаптируемость;</li> <li>➤ повторная входимость;</li> <li>➤ реентерабельность.</li> </ul>

№ п/п	Наименование раздела дисциплины	Содержание вопросов (типовых заданий)
6.	<b>Проектирование ПО при структурном подходе.</b>	<p>Какие виды диаграмм используются при структурном программировании?</p> <p>Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ функциональные диаграммы;</li> <li>➤ диаграммы потоков данных;</li> <li>➤ диаграммы переходов состояний;</li> <li>➤ диаграммы Джексона;</li> <li>➤ диаграммы Орра;</li> <li>➤ диаграммы сущность-связь.</li> </ul>
7.	<b>Проектирование ПО при объектно-ориентированном подходе.</b>	<p>Какие виды диаграмм используются при объектно-ориентированном программировании?</p> <p>Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ диаграммы вариантов использования;</li> <li>➤ диаграммы классов;</li> <li>➤ диаграммы взаимодействия;</li> <li>➤ диаграммы деятельности;</li> <li>➤ диаграммы компонентов;</li> <li>➤ диаграммы размещения.</li> </ul>
8.	<b>Тестирование и отладка ПО.</b>	<p>Ошибки при написании программы бывают</p> <p>Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ синтаксические;</li> <li>➤ орфографические;</li> <li>➤ лексические;</li> <li>➤ фонетические;</li> <li>➤ морфологические.</li> </ul> <p>Причины синтаксических ошибок</p> <p>Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ плохое знание языка программирования;</li> <li>➤ ошибки в исходных данных;</li> <li>➤ ошибки, допущенные на более ранних этапах;</li> <li>➤ неправильное применение процедуры тестирования;</li> <li>➤ неправильная установка ПО.</li> </ul> <p>Когда можно обнаружить синтаксические ошибки</p> <p>Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ при компиляции;</li> <li>➤ при отладке;</li> <li>➤ при тестировании;</li> <li>➤ на этапе проектирования;</li> <li>➤ при эксплуатации.</li> </ul> <p>Ошибки компоновки заключаются в том, что</p> <p>Выберите варианты из списка:</p> <ul style="list-style-type: none"> <li>➤ указано внешнее имя, но не объявлено;</li> <li>➤ неправильно использовано зарезервированное слово;</li> <li>➤ составлено неверное выражение;</li> <li>➤ указан неверный тип переменной.</li> </ul>

№ п/п	Наименование раздела дисциплины	Содержание вопросов (типовых заданий)
		Процедура поиска ошибки, когда известно, что она есть это Выберите варианты из списка: <ul style="list-style-type: none"> <li>➤ отладка;</li> <li>➤ тестирование;</li> <li>➤ компоновка;</li> <li>➤ транзакция;</li> <li>➤ трансляция.</li> </ul>
9.	<b>Коллективная работа по созданию программ.</b>	Какие существуют средства поддержки коллективной разработки? Выберите варианты из списка: <ul style="list-style-type: none"> <li>➤ системы управления версиями файлов;</li> <li>➤ системы управления правами пользователей;</li> <li>➤ системы синхронизации удаленных производств;</li> <li>➤ системы контроля качества интегрируемого кода;</li> <li>➤ системы мониторинга распределения нагрузки в коллективе.</li> </ul>
10.	<b>Составление программной документации.</b>	Перечислите виды документов, предусмотренных Единой системой программной документации. Какие из них являются обязательными, а какие нет.

#### 5.4. Описание критериев оценивания компетенций и шкалы оценивания

При промежуточной аттестации в форме дифференцированного зачета и экзаменов используется следующая шкала оценивания: 2 – неудовлетворительно, 3 – удовлетворительно, 4 – хорошо, 5 – отлично.

Критериями оценивания достижений показателей являются:

Наименование показателя оценивания результата обучения по дисциплине	Критерий оценивания
Знания	Знание терминов, определений, понятий
	Знание основных закономерностей, соотношений, принципов
	Объем освоенного материала
	Полнота ответов на вопросы
	Четкость изложения и интерпретации знаний
Умения	Умение решать стандартные профессиональные задачи с применением программных средств
	Умение использовать теоретические знания для выбора методики решения профессиональных задач
Навыки	Владение навыками теоретического и экспериментального исследования разрабатываемого программного обеспечения
	Качество выполнения исследований объектов профессиональной деятельности
	Самостоятельность выполнения исследований объектов профессиональной деятельности

Оценка преподавателем выставляется интегрально с учётом всех показателей и критериев оценивания.

Оценка сформированности компетенций по показателю Знания.

Критерий	Уровень освоения и оценка			
	2	3	4	5
Знание терминов, определений, понятий	Не знает терминов и определений	Знает термины и определения, но допускает неточности формулировок	Знает термины и определения	Знает термины и определения, может корректно сформулировать их самостоятельно
Знание основных закономерностей, соотношений, принципов	Не знает основные закономерности и соотношения, принципы построения знаний	Знает основные закономерности, соотношения, принципы построения знаний	Знает основные закономерности, соотношения, принципы построения знаний, их интерпретирует и использует	Знает основные закономерности, соотношения, принципы построения знаний, может самостоятельно их получить и использовать
Объем освоенного материала	Не знает значительной части материала дисциплины	Знает только основной материал дисциплины, не усвоил его деталей	Знает материал дисциплины в достаточном объеме	Обладает твердым и полным знанием материала дисциплины, владеет дополнительными знаниями
Полнота ответов на вопросы	Не дает ответы на большинство вопросов	Дает неполные ответы на все вопросы	Дает ответы на вопросы, но не все - полные	Дает полные, развернутые ответы на поставленные вопросы
Четкость изложения и интерпретации знаний	Излагает знания без логической последовательности	Излагает знания с нарушениями в логической последовательности	Излагает знания без нарушений в логической последовательности	Излагает знания в логической последовательности, самостоятельно их интерпретируя и анализируя
	Не иллюстрирует изложение поясняющими схемами, рисунками и примерами	Выполняет поясняющие схемы и рисунки небрежно и с ошибками	Выполняет поясняющие рисунки и схемы корректно и понятно	Выполняет поясняющие рисунки и схемы точно и аккуратно, раскрывая полноту усвоенных знаний
	Неверно излагает и интерпретирует знания	Допускает неточности в изложении и интерпретации знаний	Грамотно и по существу излагает знания	Грамотно и точно излагает знания, делает самостоятельные выводы

### Оценка сформированности компетенций по показателю Умения.

Критерий	Уровень освоения и оценка			
	2	3	4	5
Умение решать стандартные профессиональные задачи с применением программных средств	Не умеет решать стандартные профессиональные задачи с применением программных средств.	Допускает неточности в решении стандартных профессиональных задач с применением программных средств	Умеет решать стандартные профессиональные задачи с применением программных средств.	Безошибочно решает стандартные профессиональные задачи с применением программных средств
Умение	Не умеет	Использование	Умеет	Умело использует

Критерий	Уровень освоения и оценка			
	2	3	4	5
использовать теоретические знания для выбора методики решения профессиональных задач	использовать теоретические знания для выбора методики решения профессиональных задач	теоретических знаний для выбора методики решения профессиональных задач вызывает затруднения	использовать теоретические знания для выбора методики решения профессиональных задач	теоретические знания для выбора методики решения профессиональных задач

### Оценка сформированности компетенций по показателю Навыки.

Критерий	Уровень освоения и оценка			
	2	3	4	5
Владение навыками теоретического и экспериментального исследования разрабатываемого программного обеспечения	Не владеет навыками теоретического и экспериментального исследования разрабатываемого программного обеспечения	Не достаточно хорошо владеет навыками теоретического и экспериментального исследования разрабатываемого программного обеспечения	Владеет навыками теоретического и экспериментального исследования разрабатываемого программного обеспечения	Профессионально владеет навыками теоретического и экспериментального исследования разрабатываемого программного обеспечения
Качество выполнения исследований объектов профессиональной деятельности	Не качественно выполняет исследования объектов профессиональной деятельности, допускает грубые ошибки	Не достаточно качественно выполняет исследования объектов профессиональной деятельности, допускает и исправляет ошибки с посторонней помощью	Не достаточно качественно выполняет исследования объектов профессиональной деятельности, допускает и исправляет ошибки самостоятельно	Качественно выполняет исследования объектов профессиональной деятельности
Самостоятельность выполнения исследований объектов профессиональной деятельности	Не может самостоятельно выполнять исследования объектов профессиональной деятельности	Выполняет исследования объектов профессиональной деятельности с посторонней помощью	При выполнении исследования объектов профессиональной деятельности иногда требуется посторонняя помощь	Самостоятельно выполняет исследования объектов профессиональной деятельности

## 6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

### 6.1. Материально-техническое обеспечение

№	Наименование специальных помещений и помещений для самостоятельной работы	Оснащенность специальных помещений и помещений для самостоятельной работы
1.	Учебная аудитория для проведения лекционных занятий	Специализированная мебель. Мультимедийная установка, экран, доски
2.	Учебная аудитория для проведения лабораторных занятий	Специализированная мебель. Компьютерная техника,

3.	Читальный зал библиотеки для самостоятельной работы	Специализированная мебель. Компьютерная техника, подключенная к сети интернет и имеющая доступ в электронно-образовательную среду
----	---	--

### 6.2. Лицензионное и свободно распространяемое программное обеспечение

№	Перечень лицензионного программного обеспечения.	Реквизиты подтверждающего документа
1.	Операционная система Linux	Свободно распространяемое ПО согласно условиям лицензионного соглашения
2.	LibreOffice 7.4.0	Свободно распространяемое ПО согласно условиям лицензионного соглашения
3.	Kaspersky Endpoint Security 11 for Linux	??????????
4.	Среды программирования CodeBlocks	Свободно распространяемое ПО согласно условиям лицензионного соглашения

### 6.3. Перечень учебных изданий и учебно-методических материалов

1. Орлов, С. А. Технологии разработки программного обеспечения. Разработка сложных программных систем : учебник / С. А. Орлов. – 3-е изд. – СПб. : Питер, 2004. – 526 с. – (Учебник для вузов). – ISBN 5-94723-820-9
2. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник. – М.: Финансы и статистика, 2002. – 352 с.
3. Липаев, В. В. Программная инженерия. Методологические основы : учебник / В. В. Липаев ; Гос. ун-т – Высшая школа экономики. – М. : ТЕИС, 2006. – 607 с. – ISBN 5-7598-0424-3
4. Буч Грэди Объектно-ориентированный анализ и проектирование с примерами приложений. – М.: Вильямс, 2010. – 720 с.
5. Калянов Г.Н. CASE структурный системный анализ: Автоматизация и применение. – М.: Лори, 1996. – 242 с.
6. Вельбицкий И.В. Технология программирования. – К.: Техника, 1984. – 279 с.
7. Липаев В.В. Тестирование программ. – М.: Финансы и статистика, 1986. – 296 с.
8. Майерс Г. Искусство тестирования программ. - М.: Финансы и статистика, 1982. - 176 с.
9. Единая система программной документации. – М.: Издательство стандартов, 1998. – 164 с.

### 6.4. Перечень интернет ресурсов, профессиональных баз данных, информационно-справочных систем

1. Алексеев В.А. Паттерны проектирования программных систем [Электронный ресурс] : методические указания к проведению лабораторных работ по курсу «Архитектура программных систем» / В.А. Алексеев. — Электрон. текстовые данные. — Липецк: Липецкий государственный технический университет, ЭБС АСВ, 2016. — 33 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/74412.html>

2. Сузи Р.А. Язык программирования Python [Электронный ресурс]: учебное пособие/ Сузи Р.А.— Электрон. текстовые данные.— М.: БИНОМ. Лаборатория знаний, Интернет-Университет Информационных Технологий (ИНТУИТ), 2007.— 326 с.— Режим доступа: <http://www.iprbookshop.ru/22450>.— ЭБС «IPRbooks», по паролю

## 7. УТВЕРЖДЕНИЕ РАБОЧЕЙ ПРОГРАММЫ<sup>1</sup>

Рабочая программа утверждена на 202\_\_ /202\_\_ учебный год  
без изменений / с изменениями, дополнениями<sup>2</sup>

Протокол № \_\_\_\_\_ заседания кафедры от « \_\_\_\_ » \_\_\_\_\_ 202\_\_ г.

Заведующий кафедрой \_\_\_\_\_  
подпись, ФИО

Директор института \_\_\_\_\_  
подпись, ФИО

---

<sup>1</sup> Заполняется каждый учебный год на отдельных листах

<sup>2</sup> Нужно подчеркнуть