

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Декан  
математического факультета  
*Бурлуцкая М.Ш.*  
*подпись, расшифровка подписи*



14.04.2022

**ПРОГРАММА ПРАКТИКИ**

**УП.01.01 «Практические основы интеграции программных модулей»**

*Код и наименование практики в соответствии с учебным планом*

1. Код и наименование ППСЗ: 09.02.07 Информационные системы и программирование
2. Профиль подготовки: технический
3. Квалификация выпускника: специалист по информационным системам
4. Форма обучения: очная
5. Кафедра, отвечающая за реализацию практики: математического анализа
6. Составители программы: Найдюк Филипп Олегович, доцент кафедры математического анализа, кандидат физико-математических наук
7. Рекомендована: Научно-методическим советом математического факультета протокол от 24.03.2022 № 0500-03
8. Учебный год: 2023-2024 Семестр(ы): 5

## 9. Цель практики:

Целями учебной практики УП.01.01 «Практические основы интеграции программных модулей» являются: формирование у обучающихся первичных практических умений, приобретение первоначального практического опыта деятельности в рамках профессионального модуля ПМ.01 Осуществление интеграции программных модулей для последующего освоения общих и профессиональных компетенций.

## Задачи практики:

В результате освоения программы практики обучающийся должен **знать**:

- модели процесса разработки программного обеспечения;
- основные принципы процесса разработки программного обеспечения;
- основные подходы к интегрированию программных модулей;
- основы верификации и аттестации программного обеспечения;
- основные понятия и стандарты требований к программному обеспечению;
- основные понятия документирования разработки ПО;
- понятие жизненного цикла ПО;
- основы моделирования;
- модели жизненного цикла ПО;
- структурный подход к проектированию и разработке ПО;
- современные структурные языки программирования;
- объектно-ориентированный подход к разработке ПО;
- язык моделирования UML;
- этапы реализации ПО.

В результате освоения программы практики обучающийся должен **уметь**:

- использовать выбранную систему контроля версий;
- использовать методы для получения кода с заданной функциональностью и степенью качества;
- использовать инструментарий тестирования и анализа качества программных средств;
- использовать современные технологии и инструменты интеграции;
- пользоваться средствами разработки ПО;
- пользоваться средствами интеграции ПО.

## 10. Место практики в структуре ОПОП по ППССЗ:

Практика является обязательным разделом ППССЗ. Она представляет собой вид учебной деятельности, направленный на формирование, закрепление и развитие практических навыков и компетенций в процессе выполнения определенных видов работ, связанных с будущей профессиональной деятельностью. Практика УП.01.01 входит в профессиональный модуль ПМ.01 «Осуществление интеграции программных модулей» по специальности 09.02.07 Информационные системы и программирование. Знания и умения, формируемые в процессе прохождения учебной практики, будут использоваться в дальнейшем при освоении профессии.

## 11. Вид практики, способ и форма ее проведения

**Вид практики:** учебная.

**Способ проведения практики:** стационарная.

**Формат проведения практики:** концентрированно.

12. Планируемые результаты обучения при прохождении практики (компетенции, приобретаемый практический опыт, знания и умения), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников):

Компетенция		Планируемые результаты обучения
Код	Название	
ПК 2.1	Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент	<p>Обучающийся должен <b>знать</b>:</p> <p>Модели процесса разработки программного обеспечения.</p> <p>Основные принципы процесса разработки программного обеспечения.</p> <p>Основные подходы к интегрированию программных модулей.</p> <p>Виды и варианты интеграционных решений.</p> <p>Современные технологии и инструменты интеграции.</p> <p>Основные протоколы доступа к данным.</p> <p>Методы и способы идентификации сбоев и ошибок при интеграции приложений.</p> <p>Методы отладочных классов.</p> <p>Стандарты качества программной документации.</p> <p>Основы организации инспектирования и верификации.</p> <p>Встроенные и основные специализированные инструменты анализа качества программных продуктов.</p> <p>Графические средства проектирования архитектуры программных продуктов.</p> <p>Методы организации работы в команде разработчиков.</p> <p><b>уметь</b>:</p> <p>Анализировать проектную и техническую документацию.</p> <p>Использовать специализированные графические средства построения и анализа архитектуры программных продуктов.</p> <p>Организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов.</p> <p>Определять источники и приемники данных.</p> <p>Проводить сравнительный анализ. Выполнять отладку, используя методы и инструменты условной компиляции (классы Debug и Trace).</p> <p>Оценивать размер минимального набора тестов.</p> <p>Разрабатывать тестовые пакеты и тестовые сценарии.</p> <p>Выявлять ошибки в системных компонентах на основе спецификаций.</p>
ПК 2.2	Выполнять интеграцию модулей в программное обеспечение	<p>Обучающийся должен <b>уметь</b>:</p> <p>Использовать выбранную систему контроля версий.</p> <p>Использовать методы для получения кода с заданной функциональностью и степенью качества.</p> <p>Организовывать заданную интеграцию модулей в</p>

		<p>программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов. Использовать различные транспортные протоколы и стандарты форматирования сообщений. Выполнять тестирование интеграции. Организовывать постобработку данных. Создавать классы-исключения на основе базовых классов. Выполнять ручное и автоматизированное тестирование программного модуля. Выявлять ошибки в системных компонентах на основе спецификаций. Использовать приемы работы в системах контроля версий.</p> <p><b>знать:</b> Модели процесса разработки программного обеспечения. Основные принципы процесса разработки программного обеспечения. Основные подходы к интегрированию программных модулей. Основы верификации программного обеспечения. Современные технологии и инструменты интеграции. Основные протоколы доступа к данным. Методы и способы идентификации сбоев и ошибок при интеграции приложений. Основные методы отладки. Методы и схемы обработки исключительных ситуаций. Основные методы и виды тестирования программных продуктов. Стандарты качества программной документации. Основы организации инспектирования и верификации. Приемы работы с инструментальными средствами тестирования и отладки. Методы организации работы в команде разработчиков.</p>
<p><b>ПК 2.3</b></p>	<p><b>Выполнять отладку программного модуля с использованием специализированных программных средств</b></p>	<p>Обучающийся должен</p> <p><b>уметь:</b> Использовать выбранную систему контроля версий. Использовать методы для получения кода с заданной функциональностью и степенью качества. Анализировать проектную и техническую документацию. Использовать инструментальные средства отладки программных продуктов. Определять источники и приемники данных. Выполнять тестирование интеграции. Организовывать постобработку данных. Использовать приемы работы в системах контроля версий. Выполнять отладку, используя методы и инструменты условной компиляции. Выявлять ошибки в системных компонентах на основе</p>

		спецификаций. <b>знать:</b> Модели процесса разработки программного обеспечения.
<b>ПК 2.4</b>	<b>Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения</b>	Обучающийся должен <b>уметь:</b> Использовать выбранную систему контроля версий. Анализировать проектную и техническую документацию. Выполнять тестирование интеграции. Организовывать постобработку данных. Использовать приемы работы в системах контроля версий. Оценивать размер минимального набора тестов. Разрабатывать тестовые пакеты и тестовые сценарии. Выполнять ручное и автоматизированное тестирование программного модуля. Выявлять ошибки в системных компонентах на основе спецификаций. <b>Знать:</b> Модели процесса разработки программного обеспечения. Основные принципы процесса разработки программного обеспечения. Основные подходы к интегрированию
<b>ПК 2.5</b>	<b>Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования</b>	Обучающийся должен <b>уметь:</b> Использовать выбранную систему контроля версий. Использовать методы для получения кода с заданной функциональностью и степенью качества. Анализировать проектную и техническую документацию. Организовывать постобработку данных. Приемы работы в системах контроля версий. Выявлять ошибки в системных компонентах на основе спецификаций. <b>Знать:</b> Модели процесса разработки программного обеспечения. Основные принципы процесса разработки программного обеспечения. Основные подходы к интегрированию программных модулей. Основы верификации и аттестации программного обеспечения. Стандарты качества программной документации. Основы организации инспектирования и верификации. Встроенные и основные специализированные инструменты анализа качества программных продуктов. Методы организации работы в команде разработчиков.

13. Объем практики (в соответствии с учебным планом)

Общая трудоемкость учебной практики составляет: 2 недели, 72 часа.

**Формы промежуточной аттестации (по итогам практики):** дифференцированный зачет.

#### 14. Виды учебной работы

Вид учебной работы	Трудоемкость		
	Всего	По семестрам	
		5 семестр	
		часы	часы практической подготовки
Всего часов	72	72	
в том числе:			
Лекционные занятия (контактная работа)	18	18	
Практические занятия (контактная работа)	36	36	36
Самостоятельная работа (в т.ч. подготовка отчетных документов)	18	18	
Дифференцированный зачет			
Итого:	72	72	

#### 15. Содержание практики<sup>1</sup>

п/п	Разделы (этапы) практики	Содержание раздела
1.	Организация практики	Установочное собрание. Инструктаж по охране труда и технике безопасности.
2.	Подготовительный этап	Основные представления о тематике практики.
3.	Подготовительный этап	Содержательная формулировка задач для решения в ходе практики. Уточнение вида и объема результатов, которые должны быть получены. Изучение списка литературы и составление библиографического списка по теме задания.
4.	Научно-исследовательский этап	Изучение тем по направлению практики.
5.	(Научно-исследовательский этап). Практические занятия в лаборатории. Проведение расчетов. Анализ результатов.	1. Участие в разработке требований к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент. 2. Владение основными методологиями процессов разработки программного обеспечения. 3. Участие в проектировании программного обеспечения с использованием специализированных программных пакетов.
6.	Отчетный этап	Написание и оформление отчета в соответствии с требованиями. Подготовка презентации.
7	Аттестация	Защита отчета по практике. Подведение итогов практики.

<sup>1</sup> При реализации практики частично в форме практической подготовки необходимо отметить (\*) отдельные виды учебной работы, реализуемые в форме практической подготовки.

## Содержание учебной практики

код ПК	Учебная практика					
	Наименование ПК	Виды работ, обеспечивающих формирование ПК	Объем часов	Формат практики (распределено/концентрированно)	Уровень освоения	Показатели освоения ПК
1	2	3	4	5	6	7

ПК 2.1	Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент	<p>1. Участие в разработке требований к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент</p> <p>2. Владение основными методологиями процессов разработки программного обеспечения</p> <p>3. Участие в проектировании программного обеспечения с использованием специализированных программных пакетов</p>	12	концентрировано	2, 3	<ul style="list-style-type: none"> <li>– Анализ требований к программному обеспечению.</li> <li>– Определение характера взаимодействия компонентов программного обеспечения.</li> <li>– Анализ проектной и технической документации на уровне взаимодействия компонент программного обеспечения.</li> <li>– Точность и грамотность оформления технологической документации.</li> </ul>
ПК 2.2	Выполнять интеграцию модулей в программное обеспечение	<p>1. Участие в разработке требований к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент</p> <p>2. Владение основными методологиями процессов разработки программного обеспечения</p> <p>3. Участие в проектировании программного обеспечения с использованием специализированных программных пакетов</p>	12		2, 3	<ul style="list-style-type: none"> <li>– Демонстрация построения концептуальной, логической и физической моделей программного обеспечения и отдельных модулей.</li> <li>– Выбор технологии разработки исходного модуля исходя из его назначения.</li> <li>– Выбор методов разработки программных модулей.</li> <li>– Выбор средств разработки программных модулей.</li> <li>– Демонстрация навыков модификации программных модулей.</li> </ul>
ПК 2.3	Выполнять отладку программного модуля с использованием специализированных программных средств	<p>1. Участие в разработке требований к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент</p> <p>2. Владение основными методологиями процессов разработки программного обеспечения</p> <p>3. Участие в проектировании программного обеспечения с использованием специализированных программных пакетов</p>	12		2, 3	<ul style="list-style-type: none"> <li>– Выявление ошибок в программных модулях.</li> <li>– Определение возможности увеличения быстродействия программного продукта.</li> <li>– Определение способов и принципов оптимизации.</li> <li>– Выбор методов отладки программных модулей и программного продукта.</li> <li>– Выбор специализированных средств для отладки программного продукта.</li> <li>– Демонстрация навыков использования программных средств для отладки программного продукта.</li> </ul>



ПК 2.4	Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.	12	2, 3	<ul style="list-style-type: none"> <li>– Разработка тестовых наборов и тестовых сценариев.</li> <li>– Демонстрация устранения ошибок в программных модулях.</li> <li>– Демонстрация использования методов тестирования программного обеспечения.</li> <li>– Демонстрация навыков внесения изменения в программные модули для обеспечения качества программного обеспечения.</li> <li>– Демонстрация навыков правильного использования инструментальных средств тестирования программных модулей.</li> </ul>
ПК 2.5	Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования	24	2, 3	<ul style="list-style-type: none"> <li>– Выбор методов обеспечения качества и надежности в процессе разработки сложных программных средств.</li> <li>– Изложение основных принципов тестирования.</li> </ul>

Для характеристики уровня освоения вида работ используются следующие обозначения:

2 – репродуктивный (выполнение деятельности по образцу, инструкции или под руководством);

3 – продуктивный (планирование и самостоятельное выполнение деятельности, решение проблемных задач).

**16. Перечень учебной литературы, ресурсов сети «Интернет», необходимых для прохождения практики (список литературы оформляется в соответствии с требованиями ГОСТ и используется общая сквозная нумерация для всех видов источников)**

а) основная литература:

№ п/п	Источник
1	Златопольский, Д. М. Программирование: типовые задачи, алгоритмы, методы : [12+] / Д. М. Златопольский. – 4-е изд. (эл.). – Москва : Лаборатория знаний, 2020. – 226 с. : ил. – Режим доступа: по подписке. – URL: <a href="https://biblioclub.ru/index.php?page=book&amp;id=222873">https://biblioclub.ru/index.php?page=book&amp;id=222873</a>
2	Гуныко, А. В. Программирование : учебно-методическое пособие : [16+] / А. В. Гуныко ; Новосибирский государственный технический университет. – Новосибирск : Новосибирский государственный технический университет, 2019. – 74 с. : табл. – Режим доступа: по подписке. – URL: <a href="https://biblioclub.ru/index.php?page=book&amp;id=576267">https://biblioclub.ru/index.php?page=book&amp;id=576267</a>

б) дополнительная литература:

№ п/п	Источник
3	Рудаков А. Технология разработки программных продуктов: учебник. Изд. <u>Academia</u> . Среднее профессиональное образование. 2013 . – 208 с.
4	Рябов В.А., Несвижский А.И., Современные веб-технологии [Электронный ресурс]: Национальный Открытый Университет "ИНТУИТ", 2016. Режим доступа: <a href="http://www.studentlibrary.ru/book/intuit_364.html">http://www.studentlibrary.ru/book/intuit_364.html</a>
5	<u>Хлебников А.А.</u> Информатика : учебник : [для студ. образоват. учреждений сред. проф. образования] / А.А. Хлебников .— Изд. 5-е, стер. — Москва : Феникс, 2014 .— 428 с.
6	Крылова Г.Д. Основы стандартизации, сертификации, метрологии: Учебник для вузов. — 3-е изд.,— М.: ЮНИТИ-ДАНА, 2010. — 671 с.
7	Орлов В.В. Технологии разработки программных продуктов. – СПб.: Питер, 2013. – 437 с.
8	ГОСТ Р ИСО 9001-2001. Системы менеджмента качества. Требования. М.: ИПК Изд-во стандартов, 2010, – 140 с.
9	ГОСТ 27.002-89. Надёжность в технике. Основные понятия. Термины и определения. – М.: Издательство стандартов, 1990. – 37 с.
10	Гагарина, Л. Г. Технология разработки программного обеспечения: учеб. пособие / Л. Г. Гагарина, Е. В. Кокорева, Б. Д. Виснадул; Под ред. Л. Г. Гагариной. - М.: ФОРУМ: ИНФРА-М, 2017.-4 00 с.
11	Попов И. Ю., Блинова И. В. Теория информации: учебник для СПО; Санкт-Петербург: Лань 2020г. – 160 с.

в) информационные электронно-образовательные ресурсы (официальные ресурсы интернет)\*:

№ п/п	Ресурс
-------	--------

1.	Образовательный портал: <a href="http://www.edu.ru">http://www.edu.ru</a>
2.	Интернет университет информационных технологий - <a href="http://www.intuit.ru">http://www.intuit.ru</a>

### **17. Информационные технологии, используемые при проведении практики, включая программное обеспечение и информационно-справочные системы (при необходимости)**

Microsoft Visual Studio Community 2019, Eclipse IDE, Wing IDE, Java Platform SE 8, JDK Platform SE 16.0.2, MySQL Server, MySQL Workbench, Dr.Web Enterprise, NetBeans IDE, Denwer, VMware Player, Oracle VM VirtualBox, Microsoft Office Professional Plus 2016, LibreOffice 7

### **18. Материально-техническое обеспечение практики:**

Лаборатория «Программного обеспечения и сопровождения компьютерных систем»: автоматизированные рабочие места (моноблоки) для обучающихся и преподавателя, специализированная мебель для сервисного обслуживания ПК с заземлением и защитой от статического напряжения, проектор и экран.

Microsoft Windows 10, Xubuntu, MySQL Server Community, MySQL Shell Community, MySQL WorkBench Community, MySQL Connector Net, MatLab, NetBeans IDE, Denwer Base, DrWeb Esuite, Mozilla Firefox, Oracle VirtualBox, VMware-player. Microsoft Windows 10 Pro 64 bit, Xubuntu 20.4, Microsoft Visual Studio 2019 (Community), Mozilla Firefox, Oracle VirtualBox, VMware-player, Wireshark

#### **Мастерская 508**

- автоматизированные рабочие места специалиста по информационным системам на 8 обучающихся (процессор не ниже Celeron, оперативная память объемом не менее 6 Гб) или аналоги;
- программно-аппаратный комплекс «Соболь»;
- маркерная доска;
- программное обеспечение общего и профессионального назначения.

Microsoft Windows 10 Pro 64 bit, Microsoft Visual Studio 2019 (Community), 1C Enterprise 8 (training), MySQL Server Community, MySQL Shell Community, MySQL WorkBench Community, MySQL Connector Net, Lazarus, Free Pascal, Java, JRE, Deductor (Academic), Loginom, MatLab, NetBeans IDE, Python 2/3, LibreOffice, Gimp, Inkscape, Консультант (Student), MiKTeX, TeXstudio, Denwer Base, DrWeb Esuite, Maxima, WinDjView, Foxit Reader, 7-Zip (Архиватор), Mozilla Firefox, Oracle VirtualBox, VMware-player, Wireshark

#### **Помещение для самостоятельной работы 501**

- автоматизированные рабочие места специалиста по информационным системам на 15-16 обучающихся (процессор не ниже Celeron, оперативная память объемом не менее 6 Гб) или аналоги;
- рабочее место преподавателя (процессор не ниже Celeron, оперативная память объемом не менее 6 Гб);
- проектор и экран;
- маркерная доска;
- программное обеспечение общего и профессионального назначения.

Microsoft Windows 10 Pro 64 bit, Microsoft Visual Studio 2019 (Community), 1C Enterprise 8 (training), MySQL Server Community, MySQL Shell Community, MySQL WorkBench Community, MySQL Connector Net, Lazarus, Free Pascal, Java, JRE, Deductor (Academic), Loginom, MatLab, NetBeans IDE, Python 2/3, LibreOffice, Gimp, Inkscape, Консультант (Student), MiKTeX, TeXstudio, Denwer Base, DrWeb Esuite, Maxima, WinDjView, Foxit Reader, 7-Zip (Архиватор), Mozilla Firefox, Oracle VirtualBox, VMware-player, Wireshark

---

**19. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по практике:**

**19.1. Перечень компетенций с указанием этапов формирования и планируемых результатов обучения**

Код и содержание компетенции (или ее части)	Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенции посредством формирования знаний, умений, навыков)	Этапы формирования компетенции (разделы (этапы) практики)
---	--	---

<p>ПК 2.1</p> <p>Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент</p>	<p>Обучающийся должен <b>знать</b>:</p> <p>Модели процесса разработки программного обеспечения.</p> <p>Основные принципы процесса разработки программного обеспечения.</p> <p>Основные подходы к интегрированию программных модулей.</p> <p>Виды и варианты интеграционных решений.</p> <p>Современные технологии и инструменты интеграции.</p> <p>Основные протоколы доступа к данным.</p> <p>Методы и способы идентификации сбоев и ошибок при интеграции приложений.</p> <p>Методы отладочных классов.</p> <p>Стандарты качества программной документации.</p> <p>Основы организации инспектирования и верификации.</p> <p>Встроенные и основные специализированные инструменты анализа качества программных продуктов.</p> <p>Графические средства проектирования архитектуры программных продуктов.</p> <p>Методы организации работы в команде разработчиков.</p> <p><b>уметь</b>:</p> <p>Анализировать проектную и техническую документацию.</p> <p>Использовать специализированные графические средства построения и анализа архитектуры программных продуктов.</p> <p>Организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов.</p> <p>Определять источники и приемники данных.</p> <p>Проводить сравнительный анализ.</p> <p>Выполнять отладку, используя методы и инструменты условной компиляции (классы Debug и Trace).</p> <p>Оценивать размер минимального набора тестов.</p> <p>Разрабатывать тестовые пакеты и</p>	<p>– Анализ требований к программному обеспечению.</p> <p>– Определение характера взаимодействия компонентов программного обеспечения.</p> <p>– Анализ проектной и технической документации на уровне взаимодействия компонент программного обеспечения.</p> <p>– Оформление технологической документации.</p>
--	--	--

	<p>тестовые сценарии. Выявлять ошибки в системных компонентах на основе спецификаций.</p>	
--	---	--

<p>ПК 2.2 Выполнять интеграцию модулей в программное обеспечение</p>	<p>Обучающийся должен <b>уметь:</b>          Использовать выбранную систему контроля версий.          Использовать методы для получения кода с заданной функциональностью и степенью качества.          Организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов.          Использовать различные транспортные протоколы и стандарты форматирования сообщений.          Выполнять тестирование интеграции.          Организовывать постобработку данных.          Создавать классы-исключения на основе базовых классов.          Выполнять ручное и автоматизированное тестирование программного модуля.          Выявлять ошибки в системных компонентах на основе спецификаций.          Использовать приемы работы в системах контроля версий.</p> <p><b>знать:</b>          Модели процесса разработки программного обеспечения.          Основные принципы процесса разработки программного обеспечения.          Основные подходы к интегрированию программных модулей.          Основы верификации программного обеспечения.          Современные технологии и инструменты интеграции.          Основные протоколы доступа к данным.          Методы и способы идентификации сбоев и ошибок при интеграции приложений.          Основные методы отладки.          Методы и схемы обработки исключительных ситуаций.          Основные методы и виды тестирования программных продуктов.          Стандарты качества программной документации.          Основы организации инспектирования и верификации.</p>	<p>– Построение концептуальной, логической и физической моделей программного обеспечения и отдельных модулей.</p> <p>– Выбор технологии разработки исходного модуля исходя из его назначения.</p> <p>– Выбор методов разработки программных модулей.</p> <p>– Выбор средств разработки программных модулей.</p> <p>– Модификация программных модулей.</p>
--	--	---

	<p>Приемы работы с инструментальными средствами тестирования и отладки. Методы организации работы в команде разработчиков.</p>	
--	--	--



<p>ПК 2.3 Выполнять отладку программного модуля с использованием специализированных программных средств</p>	<p>Обучающийся должен <b>уметь:</b>          Использовать выбранную систему контроля версий.          Использовать методы для получения кода с заданной функциональностью и степенью качества.          Анализировать проектную и техническую документацию.          Использовать инструментальные средства отладки программных продуктов.          Определять источники и приемники данных.          Выполнять тестирование интеграции.          Организовывать постобработку данных.          Использовать приемы работы в системах контроля версий.          Выполнять отладку, используя методы и инструменты условной компиляции.          Выявлять ошибки в системных компонентах на основе спецификаций.  <b>знать:</b>          Модели процесса разработки программного обеспечения.</p>	<ul style="list-style-type: none"> <li>– Выявление ошибок в программных модулях.</li> <li>– Увеличение быстродействия программного продукта.</li> <li>– Определение способов и принципов оптимизации.</li> <li>– Выбор методов отладки программных модулей и программного продукта.</li> <li>– Выбор специализированных средств для отладки программного продукта.</li> <li>– Использование программных средств для отладки программного продукта.</li> </ul>
<p>ПК 2.4 Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.</p>	<p>Обучающийся должен <b>уметь:</b>          Использовать выбранную систему контроля версий.          Анализировать проектную и техническую документацию.          Выполнять тестирование интеграции.          Организовывать постобработку данных.          Использовать приемы работы в системах контроля версий.          Оценивать размер минимального набора тестов.          Разрабатывать тестовые пакеты и тестовые сценарии.          Выполнять ручное и автоматизированное тестирование программного модуля.          Выявлять ошибки в системных компонентах на основе спецификаций.</p>	<ul style="list-style-type: none"> <li>– Разработка тестовых наборов и тестовых сценариев.</li> <li>– Устранение ошибок в программных модулях.</li> <li>– Использование методов тестирования программного обеспечения.</li> <li>– Внесение изменения в программные модули для обеспечения качества</li> </ul>

	<p><b>Знать:</b>          Модели процесса разработки программного обеспечения.          Основные принципы процесса разработки программного обеспечения.          Основные подходы к интегрированию</p>	<p>программного обеспечения.          – Использование инструментальных средств тестирования программных модулей.</p>
<p>ПК 2.5 Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования</p>	<p>Обучающийся должен <b>уметь:</b>          Использовать выбранную систему контроля версий.          Использовать методы для получения кода с заданной функциональностью и степенью качества.          Анализировать проектную и техническую документацию.          Организовывать постобработку данных.          Приемы работы в системах контроля версий.          Выявлять ошибки в системных компонентах на основе спецификаций.  <b>Знать:</b>          Модели процесса разработки программного обеспечения.          Основные принципы процесса разработки программного обеспечения.          Основные подходы к интегрированию программных модулей.          Основы верификации и аттестации программного обеспечения.          Стандарты качества программной документации.          Основы организации инспектирования и верификации.          Встроенные и основные специализированные инструменты анализа качества программных продуктов.          Методы организации работы в команде разработчиков.</p>	<p>– Выбор методов обеспечения качества и надежности в процессе разработки сложных программных средств.          – Изложение основных принципов тестирования.</p>
<p>Промежуточная аттестация по практике</p>		

## 19.2 Описание критериев и шкалы оценивания компетенций (результатов обучения) при промежуточной аттестации

ПК 2.1. Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия  
**компонент**

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
<b>Пороговый</b>	<p>Обучающийся должен <b>знать:</b>            Модели процесса разработки программного обеспечения.            Основные принципы процесса разработки программного обеспечения.            Основные подходы к интегрированию программных модулей.            Виды и варианты интеграционных решений.            Современные технологии и инструменты интеграции.            Основные протоколы доступа к данным.            Методы и способы идентификации сбоев и ошибок при интеграции приложений.            Методы отладочных классов.            Стандарты качества программной документации.            Основы организации инспектирования и верификации.            Встроенные и основные специализированные инструменты анализа качества программных продуктов.            Графические средства проектирования архитектуры программных продуктов.            Методы организации работы в команде разработчиков.</p> <p><b>уметь:</b>            Анализировать проектную и техническую документацию.            Использовать специализированные графические средства построения и анализа архитектуры программных продуктов.            Организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов.</p>	<p>Знает основные источники информации, может найти основные понятия и категории по заданной тематике.</p> <p>Знает основные этапы разработки.</p> <p>Способен применять только типичные, наиболее часто встречающиеся приемы по конкретной сформулированной (выделенной) задаче.</p> <p>Способен применять теоретические знания к конкретному фактическому материалу.</p>

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
<b>Средний</b>	<p>Определять источники и приемники данных.</p> <p>Проводить сравнительный анализ.</p> <p>Выполнять отладку, используя методы и инструменты условной компиляции (классы Debug и Trace).</p> <p>Оценивать размер минимального набора тестов.</p> <p>Разрабатывать тестовые пакеты и тестовые сценарии.</p> <p>Выявлять ошибки в системных компонентах на основе спецификаций.</p>	<p>Допускает единичные ошибки в определениях, нормах, методах, классификациях и основных понятиях.</p> <p>В большинстве случаев способен выявить и применить требуемый метод/классификацию для решения поставленной задачи.</p> <p>Затрудняется в решении сложных, неординарных проблем.</p>
<b>Высокий</b>		<p>Свободно оперирует основными понятиями, терминами, нормами, методами, классификациями.</p> <p>Знает критерии эффективности разработки спецификаций отдельных компонент.</p> <p>Отлично разбирается в основных этапах разработки программного обеспечения.</p> <p>Умеет выбрать эффективный прием решения поставленной профессиональной задачи.</p>

### ПК 2.2. Выполнять интеграцию модулей в программное обеспечение

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
<b>Пороговый</b>	<p>Обучающийся должен уметь:</p> <p>Использовать выбранную систему контроля версий.</p>	<p>Знает основные источники информации, нормативные документы, может найти основные</p>

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
	<p>Использовать методы для получения кода с заданной функциональностью и степенью качества.</p> <p>Организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов.</p> <p>Использовать различные транспортные протоколы и стандарты форматирования сообщений.</p> <p>Выполнять тестирование интеграции.</p> <p>Организовывать постобработку данных.</p> <p>Создавать классы-исключения на основе базовых классов.</p> <p>Выполнять ручное и автоматизированное тестирование программного модуля.</p> <p>Выявлять ошибки в системных компонентах на основе спецификаций.</p> <p>Использовать приемы работы в системах контроля версий.</p>	<p>понятия и категории по заданной тематике.</p> <p>Умеет использовать методы для получения кода с заданной функциональностью и степенью качества.</p> <p>Знает основные принципы процесса разработки программного обеспечения.</p> <p>Способен применять только типичные, наиболее часто встречающиеся приемы по конкретной сформулированной (выделенной) задаче.</p> <p>Способен применять теоретические знания к конкретному фактическому материалу.</p>
<b>Средний</b>	<p><b>знать:</b></p> <p>Модели процесса разработки программного обеспечения.</p> <p>Основные принципы процесса разработки программного обеспечения.</p> <p>Основные подходы к интегрированию программных модулей.</p> <p>Основы верификации программного обеспечения.</p> <p>Современные технологии и инструменты интеграции.</p> <p>Основные протоколы доступа к данным.</p> <p>Методы и способы идентификации сбоев и ошибок при интеграции приложений.</p> <p>Основные методы отладки.</p> <p>Методы и схемы обработки исключительных ситуаций.</p>	<p>Допускает единичные ошибки в определениях, нормах, методах, классификациях и основных понятиях.</p> <p>Умеет использовать методы для получения кода с заданной функциональностью и степенью качества.</p> <p>Знает основные принципы процесса разработки программного обеспечения.</p> <p>В большинстве случаев способен выполнять интеграцию модулей в программное</p>

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
	<p>Основные методы и виды тестирования программных продуктов. Стандарты качества программной документации.</p>	<p>обеспечение. Затрудняется в решении сложных, неординарных проблем.</p>
<b>Высокий</b>	<p>Основы организации инспектирования и верификации. Приемы работы с инструментальными средствами тестирования и отладки. Методы организации работы в команде разработчиков.</p>	<p>Свободно оперирует основными понятиями, терминами, нормами, методами, классификациями. Знает критерии эффективности решения поставленной задачи. Умеет использовать методы для получения кода с заданной функциональностью и степенью качества. Знает основные принципы процесса разработки программного обеспечения. Способен выявлять проблемы в поставленных задачах. Умеет выбрать эффективный прием решения поставленной профессиональной задачи.</p>

**ПК 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств.**

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
<b>Пороговый</b>	<p>Обучающийся должен <b>уметь:</b></p> <p>Использовать выбранную систему контроля версий.</p> <p>Использовать методы для получения кода с заданной функциональностью и степенью качества.</p> <p>Анализировать проектную и техническую документацию.</p> <p>Использовать инструментальные средства отладки программных продуктов.</p> <p>Определять источники и приемники данных.</p> <p>Выполнять тестирование интеграции.</p> <p>Организовывать постобработку данных.</p> <p>Использовать приемы работы в системах контроля версий.</p> <p>Выполнять отладку, используя методы и инструменты условной компиляции.</p> <p>Выявлять ошибки в системных компонентах на основе спецификаций.</p> <p><b>знать:</b></p> <p>Модели процесса разработки программного обеспечения.</p> <p>Основные принципы процесса разработки программного обеспечения.</p> <p>Основные подходы к интегрированию программных модулей.</p> <p>Основы верификации и аттестации программного обеспечения.</p> <p>Методы и способы идентификации сбоев и ошибок при интеграции приложений.</p> <p>Основные методы отладки.</p> <p>Методы и схемы обработки исключительных ситуаций.</p>	<p>Знает основные источники информации, нормативные документы, может найти основные понятия и категории по заданной тематике.</p> <p>Способен применять только типичные, наиболее часто встречающиеся приемы для отладки программных модулей, их тестирования и оптимизации программного кода модуля.</p> <p>Способен применять теоретические знания к конкретному фактическому материалу.</p>

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
<b>Средний</b>		<p>Допускает единичные ошибки в определениях, нормах, методах, классификациях и основных понятиях.</p> <p>В большинстве случаев способен выявить и применить требуемый метод/классификацию для решения поставленной задачи.</p> <p>Затрудняется в решении сложных, неординарных проблем.</p> <p>Умеет создавать программу по разработанному алгоритму как отдельный модуль.</p> <p>Знает основные принципы отладки и тестирования программных продуктов.</p> <p>Умеет обеспечивать эффективность программного кода, знает способы оптимизации программного кода, умеет разрабатывать корректный программный код.</p>



<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
<b>Высокий</b>		<p>Свободно оперирует основными понятиями, терминами, нормами, методами, классификациями.</p> <p>Знает критерии эффективности решения поставленной задачи.</p> <p>Способен решать сложные проблемы в процессе выполнения работы.</p> <p>Умеет выбрать эффективный прием решения поставленной профессиональной задачи.</p> <p>С легкостью умеет осуществлять выбор способов повышения эффективности разработки ПП, выбор вида отладки, выбор метода тестирования, выполнение отладки программных модулей, разработку корректного программного кода, выбор средств работы с файлами DOS, выбор принципов тестирования программных модулей, определение правил тестирования, определение этапов тестирования, разработку корректного программного кода, критерии выбора системных управляющих блоков DOS, определение версии DOS, разработку корректного программного кода на</p>

**ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения**

<b><i>Уровень освоения компетенции</i></b>	<b><i>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</i></b>	<b><i>Критерии оценивания результатов обучения</i></b>
--	---	--

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
<b>Пороговый</b>	<p>Обучающийся должен <b>уметь:</b>  Использовать выбранную систему контроля версий.  Анализировать проектную и техническую документацию.  Выполнять тестирование интеграции.  Организовывать постобработку данных.  Использовать приемы работы в системах контроля версий.  Оценивать размер минимального набора тестов.  Разрабатывать тестовые пакеты и тестовые сценарии.  Выполнять ручное и автоматизированное тестирование программного модуля.  Выявлять ошибки в системных компонентах на основе спецификаций.</p> <p><b>Знать:</b>  Модели процесса разработки программного обеспечения.  Основные принципы процесса разработки программного обеспечения.  Основные подходы к интегрированию программных модулей.  Основы верификации и аттестации программного обеспечения.  Методы и способы идентификации сбоев и ошибок при интеграции приложений.  Методы и схемы обработки исключительных ситуаций.  Основные методы и виды тестирования программных продуктов.  Приемы работы с инструментальными средствами</p>	<p>Знает основные источники информации, нормативные документы, может найти основные понятия и категории по заданной тематике.</p> <p>Способен применять только типичные, наиболее часто встречающиеся приемы по конкретной сформулированной (выделенной) задаче.</p> <p>Знает основные методы и средства разработки всей технической документации.</p> <p>Способен применять теоретические знания к конкретному фактическому материалу.</p>

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
<b>Средний</b>	<p>тестирования и отладки. Стандарты качества программной документации. Основы организации инспектирования и верификации. Встроенные и основные специализированные инструменты анализа качества программных продуктов. Методы организации работы в команде разработчиков.</p>	<p>Допускает единичные ошибки в определениях, нормах, методах, классификациях и основных понятиях.</p> <p>В большинстве случаев способен выявить и применить требуемый метод/классификацию для решения поставленной задачи.</p> <p>Знает методы и средства разработки основной технической документации.</p> <p>Затрудняется в решении сложных, неординарных проблем.</p>
<b>Высокий</b>		<p>Свободно оперирует основными понятиями, терминами, нормами, методами, классификациями.</p> <p>Знает критерии эффективности решения поставленной задачи.</p> <p>Способен выявлять проблемы, связанные с разработкой компонент проектной и технической документации с использованием графических языков спецификаций.</p> <p>Умеет выбрать эффективный прием решения поставленной профессиональной задачи.</p> <p>Владеет всеми соответствующими умениями и навыками.</p>

**ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования**

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
<b>Пороговый</b>	<p>Обучающийся должен</p> <p><b>уметь:</b> Использовать выбранную систему контроля версий. Использовать методы для получения кода с заданной функциональностью и степенью качества. Анализировать проектную и техническую документацию. Организовывать постобработку данных. Приемы работы в системах контроля версий. Выявлять ошибки в системных компонентах на основе спецификаций.</p> <p><b>Знать:</b> Модели процесса разработки программного обеспечения.</p>	<p>Знает основные источники информации, нормативные документы, может найти основные понятия и категории по заданной тематике.</p> <p>Способен применять только типичные, наиболее часто встречающиеся приемы по конкретной сформулированной (выделенной) задаче.</p> <p>Способен применять теоретические знания к конкретному фактическому материалу.</p>
<b>Средний</b>	<p>Основные принципы процесса разработки программного обеспечения.</p> <p>Основные подходы к интегрированию программных модулей.</p> <p>Основы верификации и аттестации программного обеспечения.</p> <p>Стандарты качества программной документации.</p> <p>Основы организации инспектирования и верификации.</p> <p>Встроенные и основные специализированные</p>	<p>Допускает единичные ошибки в определениях, нормах, методах, классификациях и основных понятиях.</p> <p>В большинстве случаев способен выявить и применить требуемый метод/классификацию для решения поставленной задачи.</p> <p>Затрудняется в решении сложных, неординарных проблем.</p>

<b>Уровень освоения компетенции</b>	<b>Результаты обучения (показатели достижения заданного уровня освоения компетенций)</b>	<b>Критерии оценивания результатов обучения</b>
<b>Высокий</b>	инструменты анализа качества программных продуктов. Методы организации работы в команде разработчиков.	<p>Свободно оперирует основными понятиями, терминами, нормами, методами, классификациями.</p> <p>Знает критерии эффективности решения поставленной задачи.</p> <p>Способен выявлять проблемы, связанные с разработкой компонент проектной и технической документации с использованием графических языков спецификаций.</p> <p>Умеет выбрать эффективный прием решения поставленной профессиональной задачи.</p> <p>Владеет всеми соответствующими умениями и навыками.</p>

**19.3 Типовые контрольные задания или иные материалы, необходимые для оценки компетенций, приобретаемого практического опыта, знаний и умений, характеризующие этапы формирования компетенций в процессе освоения образовательной программы**

Не предусмотрены

**19.4. Методические материалы, определяющие процедуры оценивания приобретаемого практического опыта, знаний и умений, характеризующих этапы формирования компетенций.**

**Цель процедуры:**

Целью промежуточной аттестации по учебной практике является оценка сформированности у обучающегося умений, практического опыта (первоначального) в ходе прохождения учебной практики на основании результатов ее прохождения, подтверждаемых документами соответствующих организаций.

**Субъекты, на которые направлена процедура:**

Процедура оценивания должна охватывать всех без исключения обучающихся,

проходивших практику. В случае если обучающийся не проходил процедуру без уважительных причин, то он считается имеющим академическую задолженность.

**Период проведения процедуры:**

Процедура оценивания проводится по окончании прохождения практики, как правило, в последний день практики.

**Требования к помещениям материально-техническим средствам для проведения процедуры:**

Требования к мастерской для проведения процедуры и необходимости специализированных материально-технических средств определяются руководителем практики.

**Требования к кадровому обеспечению проведения процедуры:**

Процедуру проводит руководитель практики.

**Требования к фонду оценочных средств:**

Проведение процедуры не предусматривает применения специально разработанных оценочных средств в виде перечня вопросов, заданий и т.п. Результаты процедуры определяются руководителем практики на основании результатов ее прохождения, подтверждаемых полностью и своевременностью представления дневника практики и отчета о практике в соответствии с заданием на практику, а также успешной защиты отчета по практике.

**Описание проведения процедуры:**

Каждый обучающийся в течение практики обязан выполнить установленный программой практики объем работ, составить отчет по практике в соответствии с заданием на практику, заполнить дневник практики. Успешность, своевременность выполнения указанных работ являются одним условием прохождения процедуры, защита отчета - другим.

**Шкалы оценки результатов проведения процедуры:**

Предусмотрен дифференцированный зачет.

Результаты оценки овладения компетенциями по практике отражаются в аттестационных листах и характеристиках обучающихся.

Система оценки освоения **общекультурных компетенций** по результатам практики должна соответствовать следующим основным принципам:

- знание базовых ценностей мировой культуры и готовность опираться на них в своем личностном и общекультурном развитии;
- владение культурой мышления;
- способность к обобщению, анализу, восприятию информации;
- постановка цели и выбор путей ее достижения.

Для оценки освоения **профессиональных компетенций** по результатам учебной практики необходимо руководствоваться следующими уровнями оценки студента:

- уровень 1 - пороговый, соответствует академической оценке «удовлетворительно»;
- уровень 2 - средний, соответствует академической оценке «хорошо»;
- уровень 3 - высокий, соответствует академической оценке «отлично».

**Пороговый уровень освоения компетенций («удовлетворительно»):**

Обучающийся применяет нормы оформления документов. Способен составить письменный отчет с графической интерпретацией результатов.

Выполняет виды работ:

- разработка алгоритмов для выполнения поставленных задач;
- разработка кода программного продукта по составленному алгоритму решения задачи;

- оптимизация работы программ за счет организации нескольких потоков;
- подбор контрольных данных для проведения тестирования программного продукта по определенному сценарию;
- организация работы с серверами автоматизации из приложений;
- ведение проектной и технической документации с использованием графических языков спецификаций.

В отчете не представлен глубокий анализ и обобщение результатов практики.

При проведении промежуточной аттестации студент демонстрирует ограниченные навыки ведения диалога. Высказываемые положения не всегда убедительны и аргументированы.

Средний уровень освоения компетенций («хорошо»):

Обучающийся применяет нормы оформления документов. Выполняет виды работ, указанные выше. Способен провести наблюдение, анализ и обобщение результатов и сделать выводы по результатам учебной практики. Умеет составить письменный отчет с графической интерпретацией результатов с использованием специализированных программных продуктов.

При проведении промежуточной аттестации студент демонстрирует достаточные навыки ведения диалога, аргументация убедительна, уровень понимания результатов, полученных на практике, достаточно высок.

Высокий уровень освоения компетенций («отлично»):

Обучающийся применяет нормы оформления документов. Выполняет виды работ, указанные выше. Способен провести наблюдение, анализ и обобщение результатов и сделать выводы по результатам учебной практики, предложить мероприятия по внедрению результатов практики по месту прохождения, умеет составить письменный отчет с графической интерпретацией результатов с использованием специализированных программных продуктов и презентацию в формате PowerPoint или другом.

Студент свободно ведет диалог, проявляет при этом инициативу и уверенность. Уровень понимания излагаемого материала свидетельствует о возможности самостоятельной работы на участке прохождения практики.