

Министерство науки  
федеральное государственное бюджетное образовательное учреждение высшего образования  
"Российский экономический университет имени Г.В. Плеханова"  
**МОСКОВСКИЙ ПРИБОРОСТРОИТЕЛЬНЫЙ ТЕХНИКУМ**

**РАБОЧАЯ ПРОГРАММА ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**

**«ПМ.01. Разработка модулей программного обеспечения для компьютерных систем»**

Специальность: 09.02.07 Информационные системы и программирование

Квалификация: Специалист по тестированию в области информационных технологий

2017 год

**СОГЛАСОВАНА:**

Цикловой методической комиссией  
«Профессиональных модулей 09.02.03»

Протокол № 6-16/17 ЗК  
от «09» января 2017 года

Председатель цикловой комиссии

  
\_\_\_\_\_ А.А. Шимбирёв  
Подпись                      Инициалы Фамилия


Разработана на основе Федерального государственного образовательного стандарта по специальности среднего профессионального образования  
09.02.07 Информационные системы и программирование  
Утверждено на заседании Ученого Совета.  
Протокол 260Г от «12» января 2017 г.

Заместитель директора по учебной работе

  
\_\_\_\_\_ Д.А. Клопов  
Подпись                      Инициалы Фамилия

**УТВЕРЖДЕНА:**

Директор техникума

  
\_\_\_\_\_ А.В. Чурилов  
Подпись

**Составители (авторы):**

- Шимбирёв Андрей Андреевич, преподаватель ФГБОУ ВО "РЭУ им. Г.В.Плеханова"
- Волкова Галина Юрьевна, преподаватель ФГБОУ ВО "РЭУ им. Г.В.Плеханова"
- Комаров Андрей Алексеевич, преподаватель ФГБОУ ВО "РЭУ им. Г.В.Плеханова"
- Усачева Екатерина Юрьевна, преподаватель ФГБОУ ВО "РЭУ им. Г.В.Плеханова"

Лист актуализации  
рабочей программы п

В рабочую программу п на 2018/19 уч. год

внесены следующие изменения:

1. На основании Указа Президента РФ от 15.01.2018 года №215 на титульном листе исправлено Министерство образования и науки Российской Федерации на Министерство науки и высшего образования Российской Федерации

Дата актуализации: 30.08.2018 г

## **СОДЕРЖАНИЕ**

- 1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**
- 2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**
- 3. СТРУКТУРА И СОДЕРЖАНИЕ ОБУЧЕНИЯ ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ**
- 4. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**
- 5. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**

# 1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

## «ПМ.01 Разработка модулей программного обеспечения для компьютерных систем»

### 1.1. Область применения рабочей программы

Программа профессионального модуля (далее - программа) - является частью программы подготовки специалистов среднего звена (далее - ППСЗ) в соответствии с ФГОС по специальности 09.02.07 Информационные системы и программирование

ФГОС по специальностям СПО и соответствующих профессиональных компетенций (ПК):

- ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием
- ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием
- ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств
- ПК 1.4 Выполнять тестирование программных модулей
- ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода
- ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ

### 1.2. Цель и планируемые результаты освоения профессионального модуля

С целью овладения указанным видом профессиональной деятельности и соответствующими профессиональными компетенциями обучающийся в ходе освоения профессионального модуля должен:

#### **Иметь практический опыт:**

В разработке кода программного продукта на основе готовой спецификации на уровне модуля; использовании инструментальных средств на этапе отладки программного продукта; проведении тестирования программного модуля по определенному сценарию; использовании инструментальных средств на этапе отладки программного продукта; разработке мобильных приложений

#### **Уметь:**

осуществлять разработку кода программного модуля на языках низкого и высокого уровней; создавать программу по разработанному алгоритму как отдельный модуль; выполнять отладку и тестирование программы на уровне модуля; осуществлять разработку кода программного модуля на современных языках программирования; уметь выполнять оптимизацию и рефакторинг программного кода; оформлять документацию на программные средства

#### **Знать:**

основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования; способы оптимизации и приемы рефакторинга; основные принципы отладки и тестирования программных продуктов

### 1.3. Количество часов, отводимое на освоение профессионального модуля

	Квалификация
	Специалист по тестированию в области информационных технологий
<b>Всего часов:</b>	<b>819</b>
на освоение МДК	555
на практики	252
учебную	108
производственную	144
Экзамен по модулю	12

## 2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Результатом освоения профессионального модуля является овладение обучающимися видом профессиональной деятельности Разработка модулей программного обеспечения для компьютерных систем, в том числе профессиональными (ПК) и общими (ОК) компетенциями:

Код	Наименование результата обучения
ПК 1.1	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием
ПК 1.2	Разрабатывать программные модули в соответствии с техническим заданием
ПК 1.3	Выполнять отладку программных модулей с использованием специализированных программных средств
ПК 1.4	Выполнять тестирование программных модулей
ПК 1.5	Осуществлять рефакторинг и оптимизацию программного кода
ПК 1.6	Разрабатывать модули программного обеспечения для мобильных платформ
ОК 1.	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
ОК 2.	Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
ОК 3	Планировать и реализовывать собственное профессиональное и личностное развитие.
ОК 4	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
ОК 5	Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.
ОК 6	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей
ОК 7	Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.
ОК 8	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности
ОК 9	Использовать информационные технологии в профессиональной деятельности.
ОК 10	Пользоваться профессиональной документацией на государственном и иностранном языках

### 3. СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

#### 3.1. Структура профессионального модуля

Коды профессиональных общих компетенций	Наименования разделов профессионального модуля	Суммарный объем нагрузки, час.	Объем профессионального модуля, час.					Самостоятельная работа
			Обучение по МДК			Практики		
			Всего	Лабораторных и практических занятий	Курсовых работ (проектов)	Учебная	Производственная	
<i>ПК 1.1, ПК 1.2</i>	<i>Раздел 1. Разработка программных модулей</i>	199	197	75	30			2
<i>ПК1.3, ПК 1.4, ПК 1.5</i>	<i>Раздел 2. Поддержка и тестирование программных модулей</i>	145	145	67				
<i>ПК 1.2, ПК 1.6</i>	<i>Раздел 3. Разработка мобильных</i>	123	123	57				
<i>ПК 1.2, ПК 1.3</i>	<i>Раздел 4. Системное программирование</i>	88	88	44				
<i>ПК1.1 – ПК 1.6 ОК.01-ОК.11</i>	<i>УП.01.01 Прикладное программирование УП.01.02 Разработка программных модулей</i>	108				108		-
<i>ПК1.2 – ПК 1.6 ОК 01 – ОК 10</i>	<i>Производственная практика (по профилю специальности), часов</i>	144					144	-
	<i>Экзамен по модулю</i>	12						
	<b>Всего:</b>	819	555	243	30	108	144	2

## 2.2. Тематический план и содержание профессионального модуля (ПМ)

Наименование разделов и тем профессионального модуля (ПМ), междисциплинарных курсов (МДК)	Содержание учебного материала, лабораторные работы и практические занятия		Объем в часах
<b>Раздел 1. МДК. 01.01 Разработка программных модулей</b>			<b>199</b>
<b>Тема 1.1.1</b> <i>Жизненный цикл ПО</i>	<b>Содержание</b>		<b>2</b>
	Жизненный цикл программного обеспечения	Жизненный цикл программного обеспечения Этапы разработки программного обеспечения Стратегии и модели жизненного цикла	2
<b>Тема 1.1.2</b> <i>Структурное программирование</i>	<b>Содержание</b>		<b>14</b>
	Переменные и типы данных	Переменные Ключевые слова Константы Преобразование значений типов (кастинг) Арифметические операторы Области видимости Использование ключевых слов как идентификаторов Проверка и запрет проверки переполнения Сцепление строк Форматированный вывод Флаги форматирования строк Неявно типизированные переменные Сравнение значений разных типов	1



	Условные конструкции	Условная конструкция if Условная конструкция if – else Тернарный (третичный) оператор Условная конструкция switch – case Логика Логические операции Конъюнкция Дизъюнкция Исключающее или Отрицание Битовые логические операции Побитовое отрицание Двоичная арифметика Примеры использования логических операций Побитовые логические операции Работа с портами (конъюнкция и дизъюнкция) Исключающее ИЛИ (пример использования) Логические операции Операции сдвига Короткозамкнутые вычисления Теоремы Де Моргана	1
	Циклические конструкции	Схема работы циклов Оператор безусловного перехода (goto) Цикл с предусловием (while) Использование циклической конструкции while Цикл с постусловием (do-while) Примеры использования цикла do-while Цикл со счетчиком (for) Использование циклической конструкции for Вложенный цикл for Алгоритм Дейкстры (Цикл Дейкстры) Цикл Паук Упрощенный цикл Паук Бесконечные циклы	1
	Методы	Понятие подпрограмм Создание методов Функции и процедуры Примеры создания методов Пример правильного множественного возврата из метода Использование сторожевых операторов Методы с изменяемыми параметрами Методы с выходными параметрами Перегрузка Аргументы (параметры) Использование именованных параметров Перегрузка методов. Именованные параметры Использование опциональных параметров	1

		Метод Main() Перегрузка метода Main() Рекурсия Сложная рекурсия Примеры использования рекурсии	
	Массивы	Индекс массива. Использование одномерных массивов. Создание одномерных массивов. Двумерные массивы. Использование двумерных массивов. Массивы, состоящие из одного элемента. Трехмерные массивы. Использование трехмерных массивов. Четырехмерные массивы. Зубчатые массивы. Использование зубчатых массивов. Ключевое слово <code>params</code>	2
	<b>В том числе практических занятий и лабораторных работ</b>		
	Практическая работа №1. Условные конструкции Практическая работа №2. Циклические конструкции Практическая работа №3. Методы Практическая работа №4. Массивы		8
<b>Тема 1.1.3</b> <b>Объектно-ориентированное программирование</b>	<b>Содержание</b>		72
	Введение в ООП, классы и объекты	Понятие класса Объект и экземпляры Примеры создания классов Использование методов доступа к закрытым полям Использование свойств для доступа к закрытым полям Свойства с одним методом доступа Конструктор Конструкторы, вызывающие другие конструкторы Передача экземпляра класса в качестве аргумента Автоматически реализуемые свойства Создание экземпляра по слабой ссылке Стили использования классов Инкапсуляция	1
	Классы и объекты. Диаграммы классов	Частичные классы Частичные методы Работа с инкапсуляцией Поля только для чтения Язык UML Диаграммы классов Ассоциация Разновидности UML диаграмм Самоассоциация Использование техники фабричных методов	1

	Наследование и полиморфизм	<p>Наследование  Модификаторы доступа  Связанность и связность  Работа с конструктором  Приведение к базовому типу  Полиморфизм  Замещение метода базового класса  Переопределение метода базового класса  Операторы is и as  Герметизированные классы  Герметизированные методы  Подмена объектов</p>	1
	Абстракция. Абстрактные классы и интерфейсы	<p>Понятие абстракции  Абстрактные классы  Ключевое слово abstract  Использование абстрактных классов и методов  Абстрактные методы  Интерфейсы  Примеры использования интерфейсов  Множественное наследование абстракции  Наследование интерфейса от интерфейса  Наследование от интерфейсов  Отличие между классом и типом</p>	1
	Статические и вложенные классы	<p>Статические члены  Статические члены в нестатических классах  Константы  Статические поля и свойства  Статические конструкторы  Статические члены в абстрактных классах  Статические классы  Паттерн проектирования Singleton  Расширяющие методы  Вложенные классы  Техника Делегирования</p>	1
	Структуры и их разновидности	<p>Структура  Примеры работы со структурами  Конструкторы в структурах  Пример плохого и хорошего дизайна  Стек  Куча  Расположение структур в стеке  Расположение структур на куче  Наследование в структурах  Вложенные структуры и классы</p>	1

Перечисления (enum)	<p>Вступление. Структуры Упаковка и распаковка Структура DateTime Перечисления Получение информации о типе элементов Форматированный вывод элементов перечисления Поиск элемента перечисления по имени константы Получение информации об элементах Сравнение элементов перечисления</p>	1
Универсальные шаблоны (generics)	<p>Создание универсальных шаблонов Параметризованный метод Параметризованный делегат Ковариантность обобщений Контрвариантность обобщений Частичные классы и методы Тип Nullable Операция поглощения</p>	1
Ограничения универсальных шаблонов	<p>Ограничения параметров типа Ограничения параметров типа – «naked» Работа с коллекциями Словари Представление рефлектора</p>	1
События (events)	<p>Понятие событий Работа с событиями Абстрактные и виртуальные события Анонимные методы в событиях Использование событий Шаблон MVP</p>	1
Потоки (threads)	<p>Многозадачность Пространство имен System.Threading Работа с потоками Использование одного метода в двух потоках Передача данных в поток Анонимные методы Основные и фоновые потоки Техника синхронизации доступа к ресурсу Критическая секция</p>	1
Коллекции	<p>Понятие коллекций Работа с коллекциями Цикл foreach Оператор yield Восстановление кода из под рефлектора</p>	1
Обработка исключений (exception)	<p>Обработка исключительных ситуаций Таблица исключений Конструкция try-catch Ключевое слово throw Разработка своих пользовательских исключений Конструкция try-catch-finally Обработка внутренних исключений Порядок обработки исключений</p>	1

	Перегрузка операторов	Базовый класс Object Метод ToString Метод GetHashCode Метод Equals Метод ReferenceEquals Метод GetType Техника клонирования Клонирование графов наследования Клонирование ассоциаций Паттерн «Prototype» Операторы языка C# Перегрузка операторов	1
	Анонимные и динамические типы. LINQ	Анонимные типы Вложенные анонимные типы Язык LINQ Примеры использования языка LINQ Вложенные операторы from Операция объединения join Локальный идентификатор let Использование оператора group Оператор into Динамические типы данных Наследование динамических типов	1
	Пространства имен. Директивы препроцессора	Пространства имен Вложенные пространства имен Видимость во вложенных пространствах имен DLL библиотеки Создание DLL библиотек Модификаторы доступа internal и internal protected Директивы препроцессора	1
	Пользовательские коллекции	Коллекции Интерфейсы IEnumerable, IEnumerator Работа цикла foreach Оператор yield Интерфейсы ICollection и ICollection Создание простой коллекции с помощью реализации интерфейса IList Рассмотрение коллекции созданной с использованием паттерна Iterator	2
	Системные коллекции	Использование ArrayList и сбор элементов данных. Предназначение и использование интерфейса IComparer Работа с последовательными списками (Queue, Stack) Работа со словарями (Hashtable, SortedList, ListDictionary, HybridDictionary, OrderedDictionary) Роль интерфейса IEqualityComparer Применение специализированных наборов Работа с двоичными значениями с помощью BitArray и BitVector32 Использование методов CreateMask и CreateSection Хранение строк в наборах (StringCollection, StringDictionary, NameValueCollection) Обобщенные наборы	2

Программирование ввода-вывода	Навигация по файловой системе Чтение и запись файлов Работа с потоками Работа с изолированным хранилищем		2
Работа с текстом. Регулярные выражения	Работа со строками Форматирование вывода Локализация и глобализация Кодирование-декодирование строк. Использование класса Encoding Использование класса StringBuilder Регулярные выражения и пространство System.Text.RegularExpressions Метасимволы и Квантификаторы Метод Regex.Replace и использование внутришаблонных переменных Организация поиска по шаблону. Применение Match Построение регулярных выражений Чтение и запись файлов в разных кодировках		2
XML. Файлы конфигурации. Реестр	Загрузка и чтение из XML файла Применение XmlTextReader Загрузка содержимого XML по URL Загрузка содержимого XML из строки Позэлементное чтение XML файла Использование XmlTextWriter Запись комментариев в XML файл Запись пространства имен и префиксов в XML файл Запись в XML файл со специальным форматированием Выборка из XML с помощью XPath. (Запросы XPath) Вычисление min и max с помощью XPath Создание XPathDocument из строки Создание файлов конфигурации приложения. Чтение и запись настроек приложения Работа с реестром Windows. Использование реестра для хранения настроек приложения		2
Рефлексия	Общее понятие рефлексии и отражения Метаданные, манифест сборки, объект Type Позднее связывание. Класс Assembly Динамическая генерация кода. Класс Activator Генерация кода во время выполнения программы		2
Атрибуты	Общее понятие атрибутов Класс System.Attribute Позиционные и именованные параметры атрибутов Создание атрибутов Использование атрибутов Определение атрибутов при помощи рефлексии Системные атрибуты		2

Сериализация	Общее понятие сериализации Процесс сериализации и десериализации. Совместимость версий Управление сериализацией Форматы сериализации XML сериализация Binary сериализация SOAP сериализация Собственные методы сериализации	1
Управление памятью. Сборщик мусора	Время жизни объекта в памяти Сборщик мусора. Поколения объектов Деструкторы и финализаторы Освобождение неуправляемых ресурсов. Интерфейс IDisposable	1
Полиморфизм. Версионность	Шаблон NVI Классический полиморфизм Ad hoc полиморфизм Замещение, переопределение и перекрытие методов	1
Потоки	Многозадачность. Многозадачность на основе процессов. Многозадачность на основе потоков Работа с потоками в .NET Framework: пространство имен System.Threading, класс Thread Создание потоков. Состояния потоков Завершение потоков. Определение момента окончания потока Передача аргументов потоку Использование IsBackground Приоритеты потоков Классы Interlocked, Monitor Синхронизация потоков Сообщение между потоками. Методы Pulse(), Wait(), PulseAll()	1
Синхронизация при помощи объектов ядра	Конструкций пользовательского режима Волатильные конструкции (volatile) Взаимозапирающие конструкции (interlocked) Конструкции синхронизации режима ядра События AutoResetEvent и ManualResetEvent Семафоры Semaphore Мьютексы Mutex	1
Асинхронная модель Программирования	Пул потоков Шаблон асинхронного вызова методов Асинхронная природа делегатов Интерфейс IAsyncResult Синхронизация вызывающего потока Делегат AsyncCallback Класс AsyncResult Передача и прием специальных данных состояния	1
<b>В том числе практических занятий и лабораторных работ</b>		36
Практическая работа №5. Классы Практическая работа №6. Абстрактные классы Практическая работа №7. Статические классы		

	Практическая работа №8. Структуры Практическая работа №9. События (events) Практическая работа №10. Обработка исключений Практическая работа №11. Перегрузка операторов Практическая работа №12. Обработка текстовой информации Практическая работа №13. Файлы конфигурации. Работа с реестром Практическая работа №14. Рефлексия Практическая работа №15. Управление памятью. Практическая работа №16. Полиморфизм. Версионность Практическая работа №17. Синхронизация при помощи объектов ядра	
<b>Тема 1.1.4</b>	<b>Содержание</b>	<b>6</b>
<b>Паттерны проектирования</b>	Назначение и виды паттернов <ul style="list-style-type: none"> <li>Шаблоны и их использование в объективной реальности</li> <li>Паттерны в спорте</li> <li>Фигуры логики и модусы силлогизмов</li> <li>Шаблоны в музыке</li> <li>Порождающие шаблоны в музыке</li> <li>Структурные шаблоны в музыке</li> <li>Поведенческие шаблоны в музыке</li> <li>Шаблоны в программировании</li> <li>Определение Паттерна</li> <li>Формат описания паттернов проектирования</li> <li>Каталог паттернов проектирования</li> <li>Техника ООП</li> <li>Фасад Подсистемы</li> <li>Принципы организации каталога</li> <li>Моделирование и подходы проектирования</li> <li>Язык моделирования UML</li> <li>Набор используемых диаграмм UML</li> <li>Работа с Visual Studio</li> <li>Создание простейших диаграмм</li> <li>Класс (характеристики)</li> <li>Пакеты (характеристики)</li> <li>Наследование</li> <li>Связи отношений между классами</li> <li>Все основные связи отношений</li> <li>Package import</li> <li>Диаграммы последовательностей</li> <li>Самостоятельное Моделирование</li> <li>Прямое и обратное проектирование</li> </ul>	<b>2</b>



	Язык UML	Моделирование и подходы проектирования Язык моделирования UML Набор используемых диаграмм UML Работа с Visual Studio Создание простейших диаграмм Класс (характеристики) Пакеты (характеристики) Наследование Связи отношений между классами Все основные связи отношений Package import Диаграммы последовательностей Разновидности конечных автоматов Конечный автомат мили Граф соответствий (представление) Табличное представление Конечный автомат мура Представление автомата мура	2
	Шаблоны проектирования	Grammar Abstract Factory Builder Factory Method Prototype Singleton Adapter Bridge Composite Decorator Façade Flyweight Proxy Chain of Responsibility Command Interpreter Iterator Mediator Memento Observer State Strategy Template Method Visitor	2
<b>Тема 1.1.5.</b>	<b>Содержание</b>		39

<i>Событийно-управляемое программирование</i>	Основы разработки оконных приложений	Принципы визуального программирования Инструментальные средства, их функциональное назначение и состав Содержание файлов проекта создаваемых при проектировании и при компиляции Структура кода основного файла приложения, модуля общего назначения и файла формы Настройки выполняемого файла.	1
	Работа с формами	Принцип построения и состав библиотеки визуальных компонент Обзор основных компонентов Способы размещения компонентов Правила формирования имен объектов Основные настройки (свойства) Методы компонент События и коды обработчиков Разъяснения типовых событий каждого источника Классификация источников событий Объединение форм Дочерние формы	1
	Диалоговые сообщения	Сообщения и их классификация Достоинства и недостатки каждого вида сообщений Подпрограммы реализующие диалоговые окна	1
	Односторонние текстовые редакторы	Однострочные текстовые редакторы Преобразование типов данных Маска ввода Защита от ошибок ввода данных. Обработка входных данных	1
	Элементы переключателей	Элементы переключателей Компоненты выбора Контейнеры для хранения объектов Вкладки Компоненты выбора цвета	1
	Обработка табличной информации	Обработка данных в табличном виде Способы обращения к таблицам	1
	Главное и контекстное меню	Главное и контекстное меню Основные свойства и события Командные кнопки, кнопки с растровыми изображениями и быстрого доступа Объекты для создания браузера	1

	Ini-файлы и реестры	Инициализация и запуск приложения Настройки параметров всплывающей подсказки Ini файлы Сохранение данных Работа с реестром Заголовок секции, параметры, хранимые в секциях Чтение и сохранение параметров	1
	Многострочные текстовые редакторы	Многострочные редакторы Обработка строковой информации Методы для работы со строками. Контроль вводимых символов Диалоговые окна	1
	Работа с изображением	Компоненты для работы с графикой Обработка графической информации	1
	Использование технологии перетаскивания и встраивание объектов Drag&Drop	Методы и свойства для перемещения объектов События управления манипулятора "Мышь"	1
	Работа с файловым каталогом	Диалоговые окна Работа с файловым каталогом Объекты для работы с проводником	1
	Работа с внешними приложениями	Запуск внешних приложений Особенности, методы использования и параметры. Организация работы с приложениями Microsoft Office	2
	Организация работы с электронной почтой	Сетевые протоколы прикладного уровня Методы и свойства для работы и передачи данных по сети	2
	<b>В том числе практических занятий и лабораторных работ</b>		
	Практическая работа №18. Форма и ее свойства Практическая работа №19. Диалоговые окна и сообщения Практическая работа №20. Однострочные текстовые редакторы Практическая работа №21. Элементы переключателей и выбора Практическая работа №22. Обработка данных в табличном виде Практическая работа №23. Разработка приложения Браузер Практическая работа №24. Сохранение/чтение настроек приложения, используя Ini файлы. Практическая работа №25. Сохранение/чтение настроек приложения, используя реестр Windows. Практическая работа №26. Обработка строковой информации Практическая работа №27. Обработка графической информации Практическая работа №28. Технология Drag-And-Drop при разработки программных продуктов Практическая работа №29. Работа с файловым каталогом Практическая работа №30. Работа с приложениями Microsoft Office Практическая работа №31. Разработка почтового клиента		23
	<b>Консультация</b>		<b>2</b>
	<b>Экзамен</b>		<b>10</b>
<b>Тема 1.1.6</b>	<b>Содержание</b>		<b>6</b>

<i>Оптимизация и рефакторинг кода</i>	Введение в рефакторинг	Рефакторинг Дублирование кода Длинные методы Большой класс Длинный список параметров Расходящиеся модификации Стрельба дробью Завистливые функции Группы данных Одержимость элементарными типами Операторы типа switch Параллельные иерархии наследования Ленивый класс Теоретическая общность Временное поле Цепочки сообщений Посредник Неуместная близость Альтернативные классы с разными интерфейсами Неполнота библиотечного класса Классы данных Отказ от наследства Комментарии	1
	Составление методов	Рассматриваемые техники рефакторинга Выделение метода Встраивание метода Встраивание временной переменной Замена встраиваемой переменной вызовом метода Введение поясняющей переменной Расщепление временной переменной Замена метода объектом методов Перемещение метода Перемещение поля Выделение класса Скрытие делегирования Удаление посредника	1
	Организация данных	Замена значения данных объектом Замена значений ссылкой Замена однонаправленной связи двунаправленной Инкапсуляция коллекции Замена кода типа подклассом Замена кода типа состоянием/стратегией Замена подкласса полями	1
	Упрощение условных выражений	Декомпозиция условного оператора Консолидация условного выражения Консолидация дублирующихся условных фрагментов Удаление управляющего флага	1

		Замена условного оператора полиморфизмом Введение объекта Null	
	Решение задач обобщения	Подъем поля Подъем метода Спуск метода/поля Выделение подкласса Выделение родительского класса Выделение интерфейса Замена наследования делегированием Замена делегирования наследованием	2
<b>Тема 1.1.7</b>	<b>Содержание</b>		<b>2</b>
<b>Разработка пользовательского интерфейса.</b>	Правила разработки интерфейсов пользователя	Разработка интерфейса пользователя	2
<b>Тема 1.1.8 Основы ADO.Net</b>	<b>Содержание</b>		<b>16</b>
	Архитектура технологии ADO.NET	Рассмотрение архитектуры ADO.NET Рассмотрение «Создание подключения» Пример. Подключение к БД Создание БД и зарегистрированного пользователя Подключение зарегистрированного пользователя Подключение с SqlConnectionStringBuilder Рассмотрение «Пул соединений» Использование файлов конфигурации Использование поставщика данных .NET OleDb	1
	Получение данных. Объект SqlCommand	Создание команд с использованием SqlCommand Получение данных. Объект SqlDataReader Выполнение пакетных запросов Примеры: Создание SqlCommand, выполнение команд Пример. Выполнение вставки и удаления Возврат данных в табличном представлении Строготипизированные средства доступа Получение данных с помощью индексатора Асинхронное выполнение команд Пример. Работа со значениями NULL Рассмотрение «Транзакции» Создание параметризованных запросов Использование коллекции Parameters Пример. Вызов хранимой процедуры Процедура, принимающая параметры Возвращающие значение	1

	Автономная часть архитектуры. Объекты DataColumn, DataTable	<p>Таблицы ADO.NET. Объект DataColumn и DataRow          DataReader для создания схемы DataTable          Создание экземпляров DataColumn          Создание строки для таблицы          Создание новой таблицы          Получение информации о схеме таблицы          Проверка данных          Пример. Использование свойства Readonly          Использование свойства AllowDBNull          Пример. Использование свойства MaxLength          Пример. Использование свойства Unique          Использование свойства UniqueConstraint          Ограничение на уникальность PrimaryKey</p>	1
	Классы DataRow, DataSet и DataAdapter	<p>Рассмотрение «Строк(DataRow)»          Использование свойства RowState объекта DataRow          Пример. Значение Unchanged свойства RowState          Пример. Значение Delete свойства RowState          Пример. Значение Modified свойства RowState          Пример. Метод SetAdded, Метод SetModified          Пример. EditingRow          Применение перечисления DataRowVersion          Пример. Proposed версия строки          Данные строки, подготовленной к удалению          Рассмотрение «Адаптера данных»          Пример. Использование «маппинга»          Пример. MissingSchema          Получение данных по частям (GetPartOfRows)          Получение данных по частям в страницах</p>	1
	Работа с реляционными данными. Объект DataRelation	<p>Получение связанных данных          Пример. Создание отношений между таблицами          Просмотр дочерних строк. Метод GetChildRows          Пример. Метод GetParentRow          Пример. Получение данных при связи сам к себе          Получение данных при связи сам к себе, рекурсивный метод          Пример. Запись схемы и данных БД в XML файл          Чтение схемы и данных БД. Метод выводящий данные          Пример. Создание рассчитываемого столбца          Правила удаления и обновления родительских строк          Создание правил для удаления и обновления данных          Связанная строка, подготовленная к удалению</p>	1

	Фильтрация изменений и сортировка данных	<p>Рассмотрение объекта «DataView»          Пример. Поиск данных с помощью метода Find          Пример. Поиск данных по указанному фильтру          Пример. Поиск данных по указанному шаблону          Поиск данных с использованием метода select          Пример. Сортировка с помощью метода select          Пример. Создание DataView          Пример. Метод FindMethod.          Пример. Создание таблиц с помощью представления          Пример. Создание LINQ запросов для объектов DataTable          Пример. Работа с методом AsData View          Перекрестные запросы между таблицами          Пример. Работа с внутренним объединением          Пример. Реализация Left outer join в LINQ</p>	1
	Объект DataSet со строгим контролем типов	<p>Создание строго типизированного объекта DataSet          Создание DataSet и добавление строк          Пример. Работа со строками stroged DataSet          Поиск данных в DataSet          Пример. Работа со значениями NULL          Метод Merge для слияния таблиц          Работа с реляционными данными          Пример. Получение данных          Пример. Использование метода GetChanges          Работа с адаптерами таблиц          Метод GetData          Вставка данных помощью метода Insert</p>	1
	<b>Самостоятельная работа:</b> Передача обновлений		2
	<b>В том числе практических занятий и лабораторных работ</b>		8
	Практическая работа №32. Разработка приложения базы данных		
<b>Курсовой проект</b>			<b>30</b>
<b>Раздел 2. МДК. 01.02 Поддержка и тестирование программных модулей</b>			<b>145</b>
<b>Тема 1.2.1 Отладка и тестирование программного обеспечения</b>	<b>Содержание</b>		<b>88</b>
	Введение в тестирование программного обеспечения	<p>Понятие качества ПО. Стандарты качества ПО.          Атрибуты и характеристики качества ПО.          Основные определения тестирования.          Цели и задачи процесса тестирования.          Полный цикл тестирования. Фазы тестирования.</p>	4
	Методы и виды тестирования. Анализ требований к ПО	<p>Методы и виды тестирования. Общий обзор.          Критерии покрытия тестирования.          Требования к ПО.          Анализ требований с точки зрения пригодности к тестированию.</p>	4

Тестовая документация. Тест-план, тест-дизайн	Документы, создаваемые в процессе тестирования. Тест план. Связь тестовых планов с другими типами документов. Тест – дизайн. Возможные формы подготовки тест-дизайна.	4
Тестовая документация. Test Case. Отчет о прохождении тестов	Определение Test Case. Правила написания, степень детализации, независимость. Правила описания дефектов, понятие важности, приоритета. Ведение системы отслеживания дефектов. Составление отчетов по результатам тестирования. Учебный проект: создание test cases для учебной программы.	4
Методы	Покрытие входных данных. Допустимые и недопустимые данные. Эквивалентное разбиение. Анализ граничных значений. Попарное комбинирование. Предположение ошибок.	4
Техники тестирования	Техники, базирующиеся на интуиции и опыте инженера Техники, базирующиеся на спецификации Техники, ориентированные на код Тестирование, ориентированное на дефекты Техники, базирующиеся на условиях использования Техники, базирующиеся на природе приложения Выбор и комбинация различных техник	4
Уровни тестирования. Критерии покрытия кода программы тестами	Модульное тестирование. Драйверы. Заглушки. Интеграционное тестирование. Способы интеграционного тестирования. Системное тестирование. Понятие покрытия кода тестами. Критерии покрытия. Метрика покрытия. Анализ покрытия.	4
Виды тестирования: функциональное и нефункциональное тестирование	Функциональные виды тестирования. Тестирование безопасности, тестирование взаимодействия. Нефункциональные виды тестирования. Тестирование производительности. Нагрузочное тестирование.	4
Тестирование пользовательского интерфейса (GUI). Тестирование web-приложений	Задачи и цели тестирования пользовательского интерфейса. Функциональное тестирование пользовательского интерфейса. Тестирование удобства пользовательского интерфейса. Тестирование web-приложений	6



	Регрессионное тестирование	Регрессионное тестирование. Подходы к составлению набора test cases. Жизненный цикл ПО. Каскадный, спиральный жизненные циклы. Методологии разработки ПО. MSF, RUP, Экстремальное программирование. Команда тестирования. Роли.	6
	<b>В том числе практических занятий и лабораторных работ</b>		44
	Практическая работа №1. Тестирование требований к учебной программе. Практическая работа №2. Составление плана тестирования учебной программы. Практическая работа №3. Составление набора входных данных для тестирования учебной программы Практическая работа №4. Инсталляционное тестирование учебной программы. Практическая работа №5. Функциональное тестирование GUI		
<b>Тема 1.2.2 Командное тестирование</b>	<b>Содержание</b>		<b>21</b>
	Процессы тестирования	Место тестирования в процессе разработки ПО Тестирование и качество. Оценка качества продукта. Метрики Базовые принципы тестирования Планирование работ по тестированию. Виды деятельности, осуществляемые при составлении плана тестирования Артефакты тестирования Модели зрелости тестирования ПО (TMMi) Инструментарий тест менеджера	2
	Риски тестирования. Команда тестирования	Создание условий работы в команде. Стили управления. Выстраивание отношений. Коммуникации. Постановка задач. SMART. Хронофаги –поглотители времени Оперативное планирование. Приоритет задач. Матрица Эйзенхауэра. Работа с почтой.	4
	Оценка трудозатрат на тестирование	Определение задач, которые должны быть выполнены. Оценка трудоемкости задач. Эмпирическое правило Брукса. Практические соображения. Метод анализа видов ошибок и их влияния (FMEA). Упрощенный вариант.	4
	<b>В том числе практических занятий и лабораторных работ</b>		11
	Практическая работа №6. Составление списка задач по тестированию и создание оперативного плана		
<b>Тема 1.2.3</b>	<b>Содержание</b>		<b>24</b>

<i>Документирован ие</i>	Unit тестирование. Использование MSTest	Понятие Unit Test Типы тестирования Интеграционное тестирование Верификация и валидация Свойства хорошего Unit теста Расположение Unit тестов Именованние проектов Именованние методов и классов Какой код тестировать Unit Test Frameworks Подход AAA Пример создания простого Unit теста Более сложный пример со значением определяющим сложность пароля Работа атрибутов TestInitialize и ClassInitialize Класс Assert Свойство TextContext Data-driven test Ordered и Gineric тесты Атрибут DeploymentItem Как заставить себя писать Unit тесты Польза Unit тестирования Test Driven Development	4
	Использование Stub объектов	Виды Unit тестов Виды тестовых объектов Inversion of Control (IOC) Шаблон Service Locator Шаблон Dependency Injection Constructor Injection Property Injection Interface Injection Local Factory Method Extract and Override Factory Injection Проблемы инкапсуляции	4

	Использование Mock объектов. Moq Framework	Mock-объекты Разница между Mock и Stub Библиотека moq Пример с библиотекой Logging Library Настройка Mock-объектов когда мы используем Moq framework out и ref параметры Как проверять аргументы передаваемые в mock-объект Как можно контролировать выполнение приложения Исключения Как создавать mock'и для свойств Как создавать stub'ы для свойств Как создавать mock-объекты которые инициализируют события Использование фабрики mock'ов и режимы создания mock'ов	4
	<b>В том числе практических занятий и лабораторных работ</b>		
	Практическая работа №7. Unit тестирование Практическая работа №8. Создание сценариев тестов Практическая работа №9. Описание результатов тестирования		12
<b>Консультация</b>			<b>2</b>
<b>Экзамен</b>			<b>10</b>
<b>Раздел 3. МДК. 01.03 Разработка мобильных приложений</b>			<b>123</b>
<b>Тема 1.3.1 Основные платформы и языки разработки мобильных приложений</b>	<b>Содержание</b>		<b>66</b>
	Введение в Java технологии	Особенности языка программирования Java. Описание Java технологий. Использование интегрированной среды разработки.	2
	Введение в язык программирования Java	Языковые лексемы Java. Введение в систему типов языка Java. Работа с примитивными типами и константами. Операции языка Java. Преобразование простых типов.	2
	Методы и операторы Java	Создание и вызов методов. Перегрузка и методы с переменным числом аргументов. Операторы ветвления. Операторы циклов.	2
	Создание и использование массивов Java	Одномерные массивы. Многомерные массивы. Работа с массивами и класс Arrays.	2
	Работа со строками в Java	Основные методы класса String. Сложение и преобразование строк. Классы динамических строк. Ввод данных с клавиатуры и класс Scanner. Регулярные выражения.	2

Техники тестирования	<p>Индекс массива</p> <p>Использование одномерных массивов</p> <p>Создание одномерных массивов</p> <p>Двумерные массивы</p> <p>Использование двумерных массивов</p> <p>Массивы, состоящие из одного элемента</p> <p>Трехмерные массивы</p> <p>Использование трехмерных массивов</p> <p>Четырехмерные массивы</p> <p>Зубчатые массивы</p> <p>Использование зубчатых массивов</p> <p>Ключевое слово <code>params</code></p>	2
Разработка классов на Java	<p>Обзор основных концепций ООП.</p> <p>Объявление класса.</p> <p>Члены класса и создание объектов класса.</p> <p>Модификаторы доступа.</p> <p>Модификаторы <code>final</code> &amp; <code>static</code>.</p> <p>Использование пакетов, директив импорта и переменной среды <code>CLASSPATH</code></p>	2
Наследование и полиморфизм	<p>Наследование как механизм повторного использования кода.</p> <p>Конструктор при наследовании</p> <p>Преобразование типов и операция <code>instanceof</code>.</p> <p>Виртуальные методы и позднее связывание.</p> <p>Абстрактные классы и методы.</p>	2
Пакет <code>java.lang</code>	<p>Класс <code>Object</code> и переопределение его методов.</p> <p>Метаданные и рефлексия.</p> <p>Классы <code>System</code> и <code>Math</code>.</p>	2
Обработка ошибок в Java	<p>Концепция исключений в Java.</p> <p>Использование операторов <code>try</code>, <code>catch</code> и <code>finally</code>.</p> <p>Проверяемые и непроверяемые исключения.</p> <p>Создание своих классов исключений.</p> <p>Оператор <code>try</code> для освобождения ресурсов.</p>	2
Потоки данных в Java	<p>Обзор классов потоков.</p> <p>Работа с байтовыми потоками.</p> <p>Работа с потоками символов.</p> <p>Использование класса <code>java.io.File</code>.</p> <p>Сжатие файлов.</p> <p>Сериализация объектов в Java.</p>	1
Работа с файловой системой в NIO 2	<p>Использование интерфейса <code>Path</code>.</p> <p>Работа с атрибутами файлов.</p> <p>Основные возможности класса <code>Files</code>.</p> <p>Использование класса <code>Files</code> для обхода дерева каталогов.</p> <p>Мониторинг изменений в файловой системе.</p>	1

	Пакет java.util	Форматирование данных. Работа с датой и временем. Класс Locale и глобализация кода. Локализация и класс ResourceBundle. Генерация псевдослучайных чисел	2
	Коллекции в Java	Иерархия классов коллекций. Концепция параметризованных типов данных. Работа с параметризованным методов и интерфейсом. Обзор возможностей списков, множеств и словарей в Java.	2
	Вложенные классы в Java	Внутренние классы. Вложенные классы. Анонимные классы. Перечисления в Java.	2
	Лямбда-выражения	Синтаксис лямбда-выражений. Ссылки на методы. Функциональные интерфейсы.	2
	Паттерны проектирования	Обзор паттернов. Паттерн одиночка. Паттерн композиция. Паттерн наблюдатель.	2
	<b>В том числе практических занятий и лабораторных работ</b>		
	Практическая работа №1. Установка JDK/IDE, настройка параметров среды. Практическая работа №2. Activity, создание графического приложения Практическая работа №3. Разработка интерфейса мобильного приложения программно в коде Java Практическая работа №4. Разработка интерфейса мобильного приложения в XML Практическая работа №5. Контейнеры и позиционирование элементов в мобильных приложениях Практическая работа №6. Ресурсы Практическая работа №7. Работа с изображениями Практическая работа №8. Адаптеры		34
<b>Тема 1.3.2 Создание и тестирование модулей для мобильных приложений</b>	<b>Содержание</b>		<b>45</b>
	Обзор платформы Android	История возникновения Android Преимущества Android Архитектура Android Особенности платформы Android Основные компоненты Android Безопасность и полномочия (Permissions) Установка и настройка компонентов среды разработки	1
	Активности и ресурсы	Активности (Activity) в Android Создание Активности Жизненный цикл Активности Стеки Активностей Состояния Активностей Отслеживание изменений состояния Активности Ресурсы Отделение ресурсов от кода программы Создание ресурсов	1

	<p>Простые значения  Визуальные стили и темы  Изображения  Разметка  Анимация  Меню</p>	
Приложения и пользовательский интерфейс	<p>Использование внешних ресурсов в коде приложения  Использование ресурсов внутри ресурсов  Локализация приложения с помощью внешних ресурсов  Класс Application  Обработка событий жизненного цикла приложения  Понятие контекста  Пользовательский интерфейс  Представления (View)  Разметка (Layout)</p>	1
Намерения, меню и работа с данными	<p>Адаптеры в Android  Использование Адаптеров для привязки данных  Намерения в Android  Использование Намерений для запуска Активностей  Неявные намерения  Сохранение состояния и настроек приложения  Общие Настройки (Shared Preferences)  Работа с файлами  Использование статических файлов как ресурсов  Меню в Android  Дочерние и контекстные меню  Описание меню с помощью XML</p>	1
СУБД, контент-провайдеры и использование сетевых сервисов	<p>Работа с базами данных в Android  Курсоры (Cursor) и ContentValues  Работа с СУБД SQLite  Работа с СУБД без адаптера  Особенности работы с БД в Android  Выполнение запросов для доступа к данным  Изменение данных в БД  Использование SimpleCursorAdapter  Контент-провайдеры  Использование контент-провайдеров  Создание контент-провайдеров  Использование интернет-сервисов</p>	1
Диалоги в Android	<p>Виды Диалогов  Рекомендации по дизайну Диалогов  Создание и удаление Диалогов  Обработка событий</p>	1

Широковещательные приемники (Broadcast Receivers)	<p>Применение Широковещательных Приемников</p> <p>Жизненный цикл Приемника</p> <p>Регистрация Приемника</p> <p>Использование Ordered Broadcast</p> <p>Использование PendingIntent</p>	1
Извещения (Notifications) в Android	<p>Взаимодействие с Извещениями</p> <p>Управление Извещениями</p> <p>Создание Извещений</p> <p>Обновление Извещений</p>	1
Фрагменты (Fragments)	<p>Создание Фрагментов</p> <p>Добавление пользовательского интерфейса</p> <p>Добавление фрагментов к Активностям</p> <p>Управление Фрагментами</p> <p>Транзакции с Фрагментами</p> <p>Взаимодействие Фрагментов и Активностей</p> <p>Жизненный цикл Фрагментов</p>	1
Процессы и потоки (Threads)	<p>Жизненный цикл процесса</p> <p>Потоки</p> <p>Фоновые потоки</p> <p>Использование AsyncTask</p>	1
Сервисы (Services)	<p>Описание Сервисов в Манифесте приложения</p> <p>Запуск Сервисов</p> <p>Остановка Сервисов</p> <p>Связанные Сервисы</p> <p>Сервисы и Извещения</p> <p>Сервисы переднего плана (Foreground Services)</p> <p>Жизненный цикл Сервисов</p>	1
Виджеты (Widgets)	<p>Описание Виджетов в Манифесте приложения</p> <p>Создание разметки Виджета</p> <p>Класс AppWidgetProvider</p> <p>Создание Виджета</p> <p>Использование Конфигурационной Активности</p> <p>Использование Preview Image</p> <p>Обновление Виджетов</p>	1
Работа картами SD и внутренним хранилищем устройства	<p>Проверка доступности носителя</p> <p>Доступ к файлам</p> <p>Совместно используемые файлы и стандартные каталоги</p> <p>Файлы кэша приложений</p>	1
Загрузчики (Loaders)	<p>Обзор API Загрузчиков</p> <p>Применение Загрузчиков</p> <p>Запуск и перезапуск Загрузчиков</p> <p>Использование LoaderManager</p> <p>Использование LoaderCursor</p>	1
Беспроводные соединения	<p>Проверка сетевых соединений</p> <p>Отслеживание состояния соединений</p> <p>ConnectivityManager и NetworkInfo</p> <p>Эффективное использование сетевых соединений</p>	1

Использование AlarmManager и AlarmClock	Типы будильников в Android Однократные и повторяющиеся события Области применения AlarmManager и альтернативы (Timer и Handler) Использование AlarmClock	1
Сенсоры в Android	Обзор сенсоров Типы сенсоров и получение информации об их доступности Sensor Framework Мониторинг состояния сенсоров Лучшие практики при работе с сенсорами	1
Телефония и SMS	Совершение звонков из приложения Определение состояния и параметров телефона Мониторинг состояния телефонного модуля Использование SMS Отправка SMS Получение SMS	1
Собственные View	Использование Canvas Использование SurfaceView Использование Drawable Shape Drawable и 2D графика Модификация существующих View Создание собственных View	2
Звук и камера в Android	Запись и воспроизведение звука Основы работы с камерой в Android Использование имеющихся приложений работы с камерой Прямое управление камерой Съемка и сохранение фото и видео	2
<b>В том числе практических занятий и лабораторных работ</b>		
	Практическая работа №9. Стили и темы Практическая работа №10. Меню Практическая работа №11. Фрагменты Практическая работа №12. Многопоточность и асинхронность Практическая работа №13. Работа с мультимедиа Практическая работа №14. Настройки и состояние приложения Практическая работа №15. Работа с файловой системой Практическая работа №16. Работа с базами данных SQLite Практическая работа №17. Провайдеры контента	23
<b>Консультация</b>		<b>2</b>
<b>Экзамен</b>		<b>10</b>
<b>Раздел 4. МДК. 01.04 Системное программирование</b>		<b>88</b>
<b>Тема 1.4.1</b>	<b>Содержание</b>	<b>44</b>



<b>Программирование на языке низкого уровня</b>	Ознакомление со спецификой сборки ПО в GNU/Linux (Ubuntu)	Ознакомление с консольными текстовыми редакторами (vi,nano,mcedit) Ручная и автоматическая сборка ПО (Makefile) Модель Клиент-Интерфейс-Сервер (КИС) Статическая сборка библиотек Совместно используемые библиотеки Работа с переменными окружения	2
	Низкоуровневый ввод-вывод и файловые операции	Обзор механизмов ввода-вывода в Linux (Ubuntu) Файловые дескрипторы Системные вызовы: open, close, write, read и lseek Типы файлов Индексные дескрипторы и жесткие ссылки Права доступа к файлу Файловая система proc Два способа прочесть содержимое директории Разреженные файлы и специфика их применения Блокировка областей файла	2
	Межпроцессное взаимодействие	Механизмы межпроцессного взаимодействия Linux (Ubuntu) Неименованные каналы (pipes) Именованные каналы (named pipes) Сообщения (message queue) Разделяемая память (shared memory) Семафоры (semaphores)	2
	Сокеты	Сокеты в файловом пространстве имен (UNIX-сокеты) Парные сокеты (pair sockets) Сетевые сокеты (sockets)	2
	Сигналы	Знакомство с сигналами (signals) Отличие сигналов от других механизмов межпроцессного взаимодействия Специфика обработки сигналов (signal handling)	2
	Процессы	Клонирование процессов — fork() Замена исполняемого процесса — exec() Зомби (zombies) — причины возникновения и способы их устранения	2
	Потоки	Потоки и процессы Специфика построения многопоточных приложений (multithreading) Досрочное завершение потока Создание обработчика завершения потока Средства синхронизации потоков (synchronize primitives) Атрибуты потоков	2
	Демоны (службы)	Отличие демона от консольной утилиты Специфика разработки демонов (daemons) Создание демона использующего сетевые сокеты	2

	Консольный ввод-вывод	Специфика разработки консольных приложений Предотвращение перенаправления вывода Управление терминалом Скрытие пароля пользователя при аутентификации	2
	Отображаемая память	Отображение обычного файла Совместный доступ к файлу Частные отображения Другие применения mmap	2
	Программирование графического интерфейса с помощью GTK+	Специфика разработки 64-битных приложений Использование библиотеки ncurses Программирование графического интерфейса с помощью GTK+	2
<b>В том числе практических занятий и лабораторных работ</b>			
	Практическая работа №1. Переменные окружений Практическая работа №2. Низкоуровневый ввод-вывод и файловые операции Практическая работа №3. Межпроцессное взаимодействие Практическая работа №4. Сокеты Практическая работа №5. Сигналы Практическая работа №6. Процессы Практическая работа №7. Потoki Практическая работа №8. Демоны (службы) Практическая работа №9. Консольный ввод-вывод Практическая работа №10. Отображаемая память Практическая работа №11. Программирование графического интерфейса с помощью GTK+		22
<b>Тема 1.4.2 Разработка драйверов устройств в Linux</b>	<b>Содержание</b>		<b>44</b>
	Ядро Linux	Архитектура ядра linux Отличия разработки драйверов от прикладного ПО Потенциальные проблемы с безопасностью Лицензирование модулей ядра	2
	Модули ядра Linux	Простейший модуль ядра Знакомство с printk() Сборка модулей ядра Макроопределения __init и __exit Вопросы документирования модулей Передача модулю параметров командной строки Модули, состоящие из нескольких файлов Сборка модулей под существующее ядро	2
	Модули ядра, пространства и устройства	Модули ядра и прикладные программы Функции, которые доступны из модулей Пространство пользователя и пространство ядра Пространство имен Адресное пространство Старший и младший номер устройства	2

Символьные устройства	<p>Файлы символьных устройств  Структура file_operations  Структура file  Регистрация устройства  Отключение устройства  Пример драйвера символьного устройства с доступом только на чтение  Специфика поддержки записи для символьных устройств  Пример драйвера символьного устройства с доступом на чтение и запись</p>	2
Файловая система /proc	<p>Особенность файловой системы /proc  Пример драйвера, создающего файл в /proc с доступом на чтение  Пример драйвера, создающего файл в /proc с доступом на чтение  Блокировка процессов при конкурентном доступе к устройству  Пример драйвера, приостанавливающего работу процессов</p>	2
Управление памятью в ядре Linux	<p>Специфика управления памятью в ядре Linux  Преобразование адреса в ядре  Отображение файла в память ядра  Управление отображением в память</p>	2
Блочные устройства	<p>Блочные устройства  Регистрация блочного устройства  Операции, поддерживаемые для блочных устройств  Обработка запросов к блочному устройству  Детали реализации блочного устройства  Пример драйвера блочного устройства</p>	2
Семафоры	<p>Конкуренция и ситуация гонок  Семафоры и мьютексы  Реализация семафора в ядре Linux  Семафоры на чтение и запись  Взаимоблокировки  Пример драйвера, использующего семафоры</p>	4
Данные	<p>Типы данных и структуры ядра  Специфика использования стандартных типов данных  Назначение типам данных явного размера  Специфичные типы данных  Аспекты, связанные с совместимостью  Измерение временных интервалов  Размер страницы  Порядок следования байт в словах  Выравнивание данных  Размер указателя</p>	4
<b>В том числе практических занятий и лабораторных работ</b>		
<p>Практическая работа №12. Модули ядра, пространства и устройства  Практическая работа №13. Символьные устройства  Практическая работа №14. Файловая система /proc  Практическая работа №15. Управление памятью в ядре Linux  Практическая работа №16. Блочные устройства</p>		22

<b>Учебные практики по модулю</b>	УП.01.01 Прикладное программирование	<b>72</b>
	УП.01.02 Разработка программных модулей	<b>36</b>
<b>Производственная практика</b>	ПП.01.01 Разработка модулей программного обеспечения для компьютерных систем	<b>144</b>
<b>Экзамен по модулю</b>		<b>12</b>
<b>Всего</b>		<b>819</b>

#### 4. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

##### 4.1. Требования к минимальному материально-техническому обеспечению

Реализация программы профессионального модуля предполагает наличие

- Лаборатории программного обеспечения и сопровождения компьютерных систем

№ п/п	Оборудование	Технические средства обучения	Количество рабочих мест
1	12 автоматизированных рабочих мест для обучающихся и 1 рабочее место для преподавателя с конфигурацией: Процессор Intel Core i7, оперативная память объемом 16 Гб, жесткий диск - 1 Тб, твердотельный накопитель - 256 Гб, монитор 23", мышь, клавиатура;	проекторы - 1 шт	28
2	Парты - 14 шт	Аудиосистема - 2 шт	
3	стулья - 28 шт	Экран проектора – 1 шт	
4	стол преподавателя - 1 шт		
5	доска маркерная - 1 шт		
6	сетевой шкаф - 1 шт		
7	шкаф - 1 шт		

##### **Программное обеспечение:**

Windows 10 pro, Microsoft Office 2016, Visio 2016, Visual Studio 2019, 1С предприятие 8 (учебная версия), Unity, phpStorm 2020, notepad++, arduino, MySQL, SQL Server, Adobe Illustrator, photoshop cc, AutoCAD, Autodesk, ColerDraw, Mozilla Firefox, Microsoft Edge, Google Chrome

- Лаборатории организации и принципов построения информационных систем

№ п/п	Оборудование	Технические средства обучения	Количество рабочих мест
1	12 автоматизированных рабочих мест для обучающихся и 1 рабочее место для преподавателя с конфигурацией: Процессор Intel Core i7, оперативная память объемом 16 Гб, жесткий диск - 1 Тб, твердотельный накопитель - 256 Гб, монитор 23", мышь, клавиатура;	Проектор - 1	37
2	Столов - 21	Экран проектора - 1	
3	Стульев - 37	Колонки - 2	
4	Стол преподавателя - 1		
5	Доска маркерная		

##### **Программное обеспечение:**

Windows 10 pro, Microsoft Office 2016, Visio 2016, Visual Studio 2019, 1С предприятие 8

(учебная версия), Unity, phpStorm 2020, notepad++, arduino, MySQL, SQL Server, Adobe Illustrator, photoshop cc, AutoCAD, Autodesk, ColerDraw, Mozila Firefox, Microsoft Edge, Google Chrome

- Студия инженерной и компьютерной графики

№ п/п	Оборудование	Технические средства обучения	Количество рабочих мест
1	15 автоматизированных рабочих мест для обучающихся и 1 рабочее место для преподавателя с конфигурацией: Процессор Intel Core i5, оперативная память объемом 8 Гб, дискретная видеокарта, жесткий диск - 1 Тб, монитор 23", мышь, клавиатура;	Проектор 1	26
3	стенды 1	Экран проектора - 1	
2	столов 11	коннекторы RJ-45	
3	стульев 26	Консольный порт для управления маршрутизатором через порт стандарта RS232	
4	шкафы 1	Встроенное программное обеспечение поддерживает статическую и динамическую маршрутизацию	
5	стол преподавателя 1	Маршрутизатор поддерживает управление через локальный последовательный порт и удалённо по протоколу telnet	
6	доска маркерная 1	6 коммутаторов Cisco 296024 порта Ethernet 10/100/1000 Мбит/с Память FLASH: 64 Мб Объем ОЗУ: 128 Мб Высота RM UNIT: 1UB коммутаторе присутствует разъём для связи с ПК по интерфейсу RS-232.	
7	крессножи	6 маршрутизаторов Cisco 2801ОЗУ 256 Мб с возможностью расширения	
8	кабели различного типа	ПЗУ 128 Мб с возможностью расширения	
9	тестеры для кабеля	1 USB порт стандарта USB 1.1	
10	обжимной инструмент	2 встроенных сетевых порта Ethernet скоростью 100Мб/с	
11	кросс-панели	Внутренние разъёмы для установки дополнительных модулей расширения: не менее двух для модулей AIM	
12	Интерактивная доска	Консольный порт для управления маршрутизатором	

		через порт стандарта RS232	
13		Шлюз безопасности Cisco ASA 2 шт	
14		беспроводной маршрутизатор Linksys серии EA 2 шт	
15		IP телефоны 3 шт	

**Программное обеспечение:**

Windows 10 pro, Microsoft Office 2016, Visio 2016, Visual Studio 2019, 1С предприятие 8 (учебная версия), Unity, phpStorm 2020, notepad++, arduino, android studio, MySQL, T-SQL, SQL Server, Adobe Photoshop, Adobe Illustrator, AutoCAD, Autodesk, ColerDraw, Mozilla Firefox, Microsoft Edge, Google Chrome

- Мастерская по наладке технологического оборудования по профилю выбираемой рабочей профессии

№ п/п	Оборудование	Технические средства обучения	Количество рабочих мест
1	Парты - 10 шт	Мониторы - 13 шт	32
2	стулья - 32 шт	проекторы - 1 шт	
3	стол преподавателя - 1 шт	системные блоки - 13	
4	доска маркерная - 1 шт	мыши - 13 шт	
5	сетевой шкаф - 1 шт	клавиатуры - 13 шт	
6		Экран проектора - 1	

**Программное обеспечение:**

Windows 10 pro, Microsoft Office 2016, Visio 2016, Visual Studio 2019, 1 С предприятие 8 (учебная версия), PascalABC.net, XAMPP, Unity, Python, notepad++, arduino, MongoDB, MySql, SqlServer, Adobe Photoshop, Adobe illustrator, Corel Draw, Autodesk 3d max, autocad 2019, Mozilla Firefox, Google Chrome, Explore

- Студии разработки дизайна веб-приложений

№ п/п	Оборудование	Технические средства обучения	Количество рабочих мест
1	12 автоматизированных рабочих мест для обучающихся и 1 рабочее место для преподавателя с конфигурацией: Процессор Intel Core i7, оперативная память объемом 16 Гб, жесткий диск - 1 Тб, твердотельный накопитель - 256 Гб, монитор 23", мышь, клавиатура;	Проектор 1	12
2	столов 16	экран проектора 1	
3	стульев 32	аудиосистема 1-1	
4	шкафы 1	Мониторы 23" – 13	
5	тумбочки 2		
6	доска 1		
7	стенды 1		

**Программное обеспечение:**

Windows 10 pro, Microsoft Office 2016, Visio 2016, 1C Enterprise 8, Visual Studio 2019, Notepad++, unity, Arduino, MySQL, T-SQL, SQL Server, Adobe Photoshop, Adobe Illustrator, AutoCAD, Autodesk, ColerDraw, Mozilla Firefox, Microsoft Edge, Google Chrome

**4.2 Информационное обеспечение обучения**

**Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы**

**Печатные издания не используются. Дисциплина полностью обеспечена**

**электронными изданиями.**

**Основные источники:**

1. Федорова, Г. Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности: учеб. пособие / Г.Н. Федорова. — Москва: КУРС: ИНФРА-М, 2017. — 336 с. (Среднее Профессиональное Образование). - ISBN 978-5-906818-41-6. - Текст: электронный. - URL: <https://znanium.com/catalog/product/791799>

**Электронные издания (электронные ресурсы)**

1. Учебники по программированию <http://programm.ws/index.php>

**Дополнительные источники:**

1. Подбельский В. Язык С#. Базовый курс. Издание второе, переработанное и дополненное. Издательство: Финансы и статистика, 2017. – 408 с. - ISBN: 9785279035342 <https://urait.ru/viewer/programmirovanie-bazovyy-kurs-s-445334#page/1>



### **4.3. Общие требования к организации образовательного процесса**

При реализации образовательных программ независимо от форм получения образования могут применяться электронное обучение, дистанционные образовательные технологии в порядке, установленном федеральным органом исполнительной власти, осуществляющим функции по выработке государственной политики и нормативно-правовому регулированию в сфере образования.

Под электронным обучением понимается организация образовательного процесса с применением содержащейся в базах данных и используемой при реализации образовательных программ информации и обеспечивающих ее обработку информационных технологий, технических средств, а также информационно-телекоммуникационных сетей, обеспечивающих передачу по линиям связи указанной информации, взаимодействие участников образовательного процесса.

Под дистанционными образовательными технологиями понимаются образовательные технологии, реализуемые в основном с применением информационно-телекоммуникационных сетей при опосредованном (на расстоянии) взаимодействии обучающихся и педагогических работников.

### **4.4. Кадровое обеспечение образовательного процесса**

#### **Требования к квалификации педагогических (инженерно-педагогических) кадров, обеспечивающих обучение по междисциплинарному курсу (курсам):**

Квалификация педагогических работников образовательной организации должна отвечать квалификационным требованиям, указанным в квалификационных справочниках, и (или) профессиональных стандартах (при наличии).

Педагогические работники, привлекаемые к реализации образовательной программы, должны получать дополнительное профессиональное образование по программам повышения квалификации, в том числе в форме стажировки в организациях направления деятельности которых соответствует области профессиональной деятельности, указанной в пункте 1.5 настоящего ФГОС СПО, не реже 1 раза в 3 года с учетом расширения спектра профессиональных компетенций.

Доля педагогических работников (в приведенных к целочисленным значениям ставок), обеспечивающих освоение обучающимися профессиональных модулей, имеющих опыт деятельности не менее 3 лет в организациях, направление деятельности которых соответствует области профессиональной деятельности, указанной в пункте 1.5 настоящего ФГОС СПО, в общем числе педагогических работников, реализующих образовательную программу, должна быть не менее 25 процентов.

#### **Требования к квалификации педагогических кадров, осуществляющих руководство практикой Инженерно-педагогический состав:**

Квалификация педагогических работников образовательной организации должна отвечать квалификационным требованиям, указанным в квалификационных справочниках, и (или) профессиональных стандартах (при наличии).

Педагогические работники, привлекаемые к реализации образовательной программы, должны получать дополнительное профессиональное образование по программам повышения квалификации, в том числе в форме стажировки в организациях направления деятельности которых соответствует области профессиональной деятельности, указанной

в [пункте 1.5](#) настоящего ФГОС СПО, не реже 1 раза в 3 года с учетом расширения спектра профессиональных компетенций.

Доля педагогических работников (в приведенных к целочисленным значениям ставок), обеспечивающих освоение обучающимися профессиональных модулей, имеющих опыт деятельности не менее 3 лет в организациях, направление деятельности которых соответствует области профессиональной деятельности, указанной в [пункте 1.5](#) настоящего ФГОС СПО, в общем числе педагогических работников, реализующих образовательную программу, должна быть не менее 25 процентов.

Мастера: Не предусмотрены

## 5. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Код и наименование профессиональных и общих компетенций, формируемых в рамках модуля	Критерии оценки	Методы оценки
<b>Раздел 1. Разработка программных модулей</b>		
ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием	<p>Оценка «<b>отлично</b>» - техническое задание проанализировано, алгоритм разработан, соответствует техническому заданию и оформлен в соответствии со стандартами, пояснены его основные структуры.</p> <p>Оценка «<b>хорошо</b>» - алгоритм разработан, оформлен в соответствии со стандартами и соответствует заданию, пояснены его основные структуры.</p> <p>Оценка «<b>удовлетворительно</b>» - алгоритм разработан и соответствует заданию.</p>	<p>Экзамен / зачет в форме собеседования по практическому заданию</p> <p>Защита портфолио по практическим работам</p>
ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием	<p>Оценка «<b>отлично</b>» - программный модуль разработан по имеющемуся алгоритму в среде разработки методами объектно- ориентированного / структурного программирования и полностью соответствует техническому заданию, соблюдены и пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p> <p>Оценка «<b>хорошо</b>» - программный модуль разработан по имеющемуся алгоритму в среде разработки методами объектно- ориентированного / структурного программирования и практически соответствует техническому заданию с</p>	<p>Экзамен / зачет в форме собеседования по практическому заданию</p> <p>Защита портфолио по практическим работам</p>

	<p>незначительными отклонениями, пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p> <p>Оценка <b>«удовлетворительно»</b> - программный модуль разработан по имеющемуся алгоритму в среде разработки методами объектно-ориентированного/ структурного программирования и соответствует техническому заданию; документация на модуль оформлена без существенных отклонений от стандартов.</p>	
<b>Раздел 2. Поддержка и тестирование программных модулей</b>		
ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств	<p>Оценка <b>«отлично»</b> - выполнена отладка модуля с пояснением особенностей отладочных классов; сохранены и представлены результаты отладки.</p> <p>Оценка <b>«хорошо»</b> - выполнена отладка модуля сохранены и представлены результаты отладки.</p> <p>Оценка <b>«удовлетворительно»</b> - выполнена отладка модуля, пояснены ее результаты.</p>	<p>Экзамен / зачет в форме собеседования по практическому заданию</p> <p>Защита портфолио по практическим работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p>
ПК 1.4 Выполнять тестирование программных модулей	<p>Оценка <b>«отлично»</b> - выполнено тестирование модуля, в том числе с помощью инструментальных средств, и оформлены результаты тестирования в соответствии со стандартами: выполнено функциональное тестирование, выполнена и представлена оценка тестового покрытия, сделан вывод о достаточности тестового пакета.</p> <p>Оценка <b>«хорошо»</b> - выполнено тестирование модуля, в том числе с помощью инструментальных средств, и оформлены результаты тестирования: выполнено функциональное тестирование, выполнена и представлена оценка тестового покрытия.</p> <p>Оценка <b>«удовлетворительно»</b> - выполнено тестирование модуля и оформлены результаты тестирования: выполнено функциональное тестирование, выполнена и представлена оценка тестового покрытия с некоторыми погрешностями.</p>	<p>Экзамен / зачет в форме собеседования по практическому заданию</p> <p>Защита портфолио по практическим работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p>
ПК 1.5 Осуществлять	Оценка <b>«отлично»</b> - определены	Экзамен / зачет в форме

<p>рефакторинг и оптимизацию программного кода</p>	<p>качественные характеристики программного кода с помощью инструментальных средств; выявлены фрагменты некачественного кода; выполнен рефакторинг на уровнях переменных, функций, классов, алгоритмических структур; проведена оптимизация и подтверждено повышение качества программного кода.</p> <p>Оценка «<b>хорошо</b>» - определены качественные характеристики программного кода с помощью инструментальных средств; выявлены фрагменты некачественного кода; выполнен рефакторинг на нескольких уровнях; проведена оптимизация и выполнена оценка качества полученного программного кода.</p> <p>Оценка «<b>удовлетворительно</b>» - определены качественные характеристики программного кода частично с помощью инструментальных средств; выявлено несколько фрагментов некачественного кода; выполнен рефакторинг на нескольких уровнях; проведена оптимизация и выполнена оценка качества полученного программного кода.</p>	<p>собеседования по практическому заданию Защита портфолио по практическим работам Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p>
<p><b>Раздел 3. Разработка мобильных приложений</b></p>		
<p>ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием</p>	<p>Оценка «<b>отлично</b>» - программный модуль разработан по имеющемуся алгоритму в среде разработки методами объектно- ориентированного/ структурного программирования и полностью соответствует техническому заданию, соблюдены и пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p> <p>Оценка «<b>хорошо</b>» - программный модуль разработан по имеющемуся алгоритму в среде разработки методами объектно- ориентированного/ структурного программирования и практически соответствует техническому заданию с незначительными отклонениями, пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p>	<p>Экзамен / зачет в форме собеседования по практическому заданию Защита портфолио по практическим работам Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p>

	<p>Оценка <b>«удовлетворительно»</b> - программный модуль разработан по имеющемуся алгоритму в среде методами объектно-ориентированного / структурного программирования и соответствует техническому заданию; документация на модуль оформлена без существенных отклонений от стандартов</p>	
<p>ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ.</p>	<p>Оценка <b>«отлично»</b> - разработан модуль для заданного мобильного устройства с соблюдением основных этапов разработки на одном из современных языков программирования; при проверке работоспособности модуля на устройстве или эмуляторе установлено его соответствие спецификации.</p> <p>Оценка <b>«хорошо»</b> - разработан модуль для заданного мобильного устройства с учетом основных этапов разработки на одном из современных языков программирования; при проверке работоспособности модуля на устройстве или эмуляторе установлено соответствие выполняемых функций спецификации с незначительными отклонениями.</p> <p>Оценка <b>«удовлетворительно»</b> - разработан модуль для заданного мобильного устройства на одном из современных языков программирования; при проверке работоспособности модуля на устройстве или эмуляторе установлено соответствие основных выполняемых функций спецификации.</p>	<p>Экзамен / зачет в форме собеседования по практическому заданию Защита портфолио по практическим работам Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p>
<p><b>Раздел модуля 4. Системное программирование</b></p>		
<p>ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием</p>	<p>Оценка <b>«отлично»</b> - программный модуль разработан по имеющемуся алгоритму в среде разработки методами объектно-ориентированного/структурного программирования и полностью соответствует техническому заданию, соблюдены и пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p> <p>Оценка <b>«хорошо»</b> - программный модуль разработан по имеющемуся алгоритму в среде разработки методами объектно-ориентированного/структурного программирования и практически</p>	<p>Экзамен / зачет в форме собеседования по практическому заданию Защита портфолио по практическим работам Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p>

	<p>соответствует техническому заданию с незначительными отклонениями, пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p> <p>Оценка «удовлетворительно» - программный модуль разработан по имеющемуся алгоритму в среде разработки методами объектно-ориентированного/ структурного программирования и соответствует техническому заданию; документация на модуль оформлена без существенных отклонений от стандартов.</p>	
<p>ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств</p>	<p>Оценка «<b>отлично</b>» - выполнена отладка модуля с пояснением особенностей отладочных классов; сохранены и представлены результаты отладки.</p> <p>Оценка «<b>хорошо</b>» - выполнена отладка модуля сохранены и представлены результаты отладки.</p> <p>Оценка «<b>удовлетворительно</b>» - выполнена отладка модуля, пояснены ее результаты.</p>	<p>Экзамен / зачет в форме собеседования по практическому заданию</p> <p>Защита портфолио по практическим работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p>
<p>ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.</p>	<ul style="list-style-type: none"> <li>- обоснованность постановки цели, выбора и применения методов и способов решения профессиональных задач;</li> <li>- адекватная оценка и самооценка эффективности и качества выполнения профессиональных задач</li> </ul>	<p>Экспертное наблюдение за выполнением работ</p>
<p>ОП 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.</p>	<ul style="list-style-type: none"> <li>- использование различных источников, включая электронные ресурсы, медиаресурсы, Интернет-ресурсы, периодические издания по специальности для решения профессиональных задач</li> </ul>	
<p>ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.</p>	<ul style="list-style-type: none"> <li>- демонстрация ответственности за принятые решения</li> <li>- обоснованность самоанализа и коррекция результатов собственной работы;</li> </ul>	
<p>ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.</p>	<ul style="list-style-type: none"> <li>- взаимодействовать с обучающимися, преподавателями и мастерами в ходе обучения, с руководителями учебной и производственной практик;</li> <li>- обоснованность анализа работы членов команды (подчиненных)</li> </ul>	

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.	- демонстрировать грамотность устной и письменной речи, - ясность формулирования и изложения мыслей	
ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей.	- соблюдение норм поведения во время учебных занятий и прохождения учебной и производственной практик,	
ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.	- эффективное выполнение правил ТБ во время учебных занятий, при прохождении учебной и производственной практик; - демонстрация знаний и использование ресурсосберегающих технологий в профессиональной деятельности	
ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.	- эффективность использовать средств физической культуры для сохранения и укрепления здоровья при выполнении профессиональной деятельности.	
ОК 09. Использовать информационные технологии в профессиональной деятельности.	- эффективность использования информационно-коммуникационных технологий в профессиональной деятельности согласно формируемым умениям и получаемому практическому опыту;	
ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языках.	- эффективность использования в профессиональной деятельности необходимой технической документации, в том числе на английском языке.	

#### **Разработчики:**

- Шимбирёв Андрей Андреевич, преподаватель ФГБОУ ВО "РЭУ им. Г.В.Плеханова"
- Волкова Галина Юрьевна, преподаватель ФГБОУ ВО "РЭУ им. Г.В.Плеханова"
- Комаров Андрей Алексеевич, преподаватель ФГБОУ ВО "РЭУ им. Г.В.Плеханова"
- Митасов Никита Сергеевич, преподаватель ФГБОУ ВО "РЭУ им. Г.В.Плеханова"

#### **Эксперты:**

---

(место работы)	(занимаемая должность)	(инициалы, фамилия)
----------------	------------------------	---------------------

---

(место работы)	(занимаемая должность)	(инициалы, фамилия)
----------------	------------------------	---------------------