

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
федерального государственного бюджетного образовательного учреждения высшего образования
**«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(МИ ВлГУ)**

Отделение среднего профессионального образования

«УТВЕРЖДАЮ»
Заместитель директора по УР
_____ Д.Е. Андрианов
« 20 » 05 2025 г.

РАБОЧАЯ ПРОГРАММА ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Разработка модулей программного обеспечения для компьютерных систем

для специальности 09.02.07 Информационные системы и программирование

Муром, 2025 г.

Рабочая программа профессионального модуля разработана на основе Федерального государственного образовательного стандарта (далее - ФГОС) по специальности среднего профессионального образования (далее - СПО) 09.02.07 Информационные системы и программирование №1547 от 09 декабря 2016 года.

Кафедра-разработчик: информационных систем.

Рабочую программу составил: к.т.н., доцент Канунова Е.Е.

от «06» мая 2025 г.

(подпись)

Рабочая программа рассмотрена и одобрена на заседании кафедры ИС.

Протокол № 17

от «06» мая 2025 г.

Заведующий кафедрой ИС *Андреианов Д.Е.*

(подпись)

СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	4
2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	7
3. СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	8
4. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	17
5. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	19

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Разработка модулей программного обеспечения для компьютерных систем

1.1. Область применения примерной программы

Примерная программа учебной дисциплины является частью программы подготовки специалистов среднего звена в соответствии с ФГОС по специальности 09.02.07 Информационные системы и программирование в части освоения основного вида профессиональной деятельности (ВПД): **Результатом освоения профессионального модуля является овладение обучающимися видом профессиональной деятельности Программист, в том числе профессиональными (ПК) и общими (ОК) компетенциями** 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
- ПК 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств
- ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения

1.2. Цели и задачи профессионального модуля – требования к результатам освоения профессионального модуля

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

иметь практический опыт:

- владения современными программными средствами, предназначенными для разработки мобильных приложений (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- владение функциональным программированием (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.).

уметь:

- Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием. (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Оценка сложности алгоритма (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Создавать программу по разработанному алгоритму как отдельный модуль. (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ. (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);

- Выполнять отладку и тестирование программы на уровне модуля. (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Применять инструментальные средства отладки программного обеспечения. (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Выполнять отладку и тестирование программы на уровне модуля (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Выполнять оптимизацию и рефакторинг программного кода (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Осуществлять разработку кода программного модуля на современных языках программирования (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Использовать методы для получения кода с заданной функциональностью и степенью качества. (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Использовать инструментальные средства отладки программных продуктов. (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.).

знать:

- Основные этапы разработки программного обеспечения. (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Основные этапы разработки программного обеспечения. (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Основные принципы отладки и тестирования программных продуктов. (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Инструментарий отладки программных продуктов. (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Основные виды и принципы тестирования программных продуктов (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Инструментальные средства анализа алгоритма (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Методы организации рефакторинга и оптимизации кода (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Модели процесса разработки программного обеспечения (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Основные принципы процесса разработки программного обеспечения (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Основные подходы к интегрированию программных модулей (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);

- Основы верификации программного обеспечения (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Современные технологии и инструменты интеграции (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Основные методы отладки (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Методы и схемы обработки исключительных ситуаций (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Основные методы и виды тестирования программных продуктов (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.);
- Основные задачи сопровождения информационной системы (ОК 01., ОК 02., ОК 03., ОК 04., ОК 05., ОК 06., ОК 07., ОК 09., ОК 10., ОК 11., ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3., ПК 2.4.).

1.3. Рекомендуемое количество часов на освоение программы профессионального модуля:

всего - 674 часа, в том числе:

максимальной учебной нагрузки обучающегося - 512 часов, включая:

обязательной аудиторной нагрузки обучающегося - 392 часа;

самостоятельной нагрузки обучающегося - 120 часов;

учебной и производственной практики - 144 часа;

экзамен по модулю - 18 часов.

2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Результатом освоения профессионального модуля является овладение обучающимися видом профессиональной деятельности 09.02.07 Информационные системы и программирование, 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
ПК 2.3.	Выполнять отладку программного модуля с использованием специализированных программных средств
ПК 2.4.	Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения
ОК 01.	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
ОК 02.	Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности
ОК 03.	Планировать и реализовывать собственное профессиональное и личностное развитие
ОК 04.	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами
ОК 05.	Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста
ОК 06.	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей
ОК 07.	Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях
ОК 09.	Использовать информационные технологии в профессиональной деятельности
ОК 10.	Пользоваться профессиональной документацией на государственном и иностранном языке
ОК 11.	Планировать предпринимательскую деятельность в профессиональной сфере

3. СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

3.1. Тематический план профессионального модуля

Коды профессиональных компетенций	Наименования разделов профессионального модуля	Всего часов (макс. учебная нагрузка и практики)	Объем времени, отведенный на освоение междисциплинарного курса (курсов)					Практика	
			Обязательная аудиторная учебная нагрузка обучающегося			Самостоятельная работа обучающегося		Учебная, часов	Производственная (по профилю специальности), часов
			Всего, часов	в т. ч. лабораторные работы и практические занятия, часов	в т. ч. курсовая работа (проект), часов	Всего, часов	в т. ч. курсовая работа (проект), часов		
1	2	3	4	5	6	7	8	9	10
ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 2.3.	Раздел 1. МДК.01.01 Разработка программных модулей	162	130	80		32			
ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 2.3., ПК 2.4.	Раздел 2. МДК.01.02 Поддержка и тестирование программных модулей	84	66	32		18			
ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5., ПК 1.6., ПК 2.3.	Раздел 3. МДК.01.03 Разработка мобильных приложений	146	114	64	16	32	16		
ПК 1.1., ПК 1.2., ПК 1.3., ПК 1.4., ПК 1.5.	Раздел 4. МДК.01.04 Системное программирование	120	82	48		38			
	Учебная практика	108						108	
	Производственная практика	36							36
	Квалификационный экзамен	18							
	Всего:	674	392	224		120	16	108	36

3.2. Содержание обучения по профессиональному модулю

Наименование разделов профессионального модуля (ПМ), междисциплинарных курсов (МДК) и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся	Объем часов	Уровень освоения
1	2	3	4
МДК.01.01 Разработка программных модулей		162	
	3 семестр		
Раздел 1	Разработка программных модулей		
Тема 1.1 Программная инженерия	Содержание учебного материала		
	Лекционные занятия. Программная инженерия. Информатизация общества и технологии программирования. Процессы жизненного цикла программного	6	1

	обеспечения. Этапы трансляции текстов программы в исполняемый код. Стандарты языков C и C++.		
	Самостоятельная работа обучающихся. Сложные условия. Двойные циклы. Изучение алгоритмов сортировки и поиска.	10	3
Тема 1.2 Процедурно-ориентированное программирование в C++	Содержание учебного материала		
	Лекционные занятия. Операции в языке C++. Приоритеты операций. Приведение типов. Побитовые операции. Операции сдвига. Адресная арифметика. Указатели. Массивы. Статические и динамические массивы. Многомерные массивы.	10	1
	Практические занятия. Форматированный ввод-вывод. Арифметические операции. Программы с линейной структурой. Программы с разветвляющей структурой. Разработка программ с циклической структурой. Побитовые операции. Указатели. Одномерные массивы. Одномерные массивы. Операции над массивами. Многомерные массивы. Операции над массивами.	16	2
	4 семестр		
Тема 1.3 Программная инженерия	Содержание учебного материала		
	Практические занятия. Коллективная разработка приложений с использованием GitHub.	2	2
	Самостоятельная работа обучающихся. Создание проекта на makefile, make, qmake и premake.	10	3
Тема 1.4 Процедурно-ориентированное программирование в C++	Содержание учебного материала		
	Лекционные занятия. Время существования и область видимости переменных. Функции. Рекурсивные функции. Перегрузка функций. Шаблоны функций. Структуры и объединения. Директивы препроцессора. Обработка исключительных ситуаций. Основные методы программирования.	16	1
	Практические занятия. Строки в C++. Функции. Рекурсивные функции. Перегрузка функций. Шаблоны функций. Работа с векторами. Исключительные ситуации. Обработка исключений в C++. Объединения. Разработка динамически присоединяемых библиотек.	18	2
	Лабораторные работы. Структуры. Файлы и строки в C++.	8	3
Тема 1.5 Объектно-ориентированное программирование в C++	Содержание учебного материала		
	Лекционные занятия. Инкапсуляция. Классы и объекты. Конструкторы и деструкторы. Друзья класса. Наследование. Простое наследование. Множественное наследование. Виртуальные базовые классы. Полиморфизм. Перегрузка операций. Преобразования типов, определяемых пользователем.	18	1

	Чистый полиморфизм. Виртуальные функции класса. Абстрактные классы. Параметрический полиморфизм. Стандартная библиотека шаблонов (STL).		
	Практические занятия. Множества в C++. Словари в C++. Множественное наследование в C++. Дружественные функции в C++. Стеки в C++. Очереди в C++. Знакомство с потоками.	12	2
	Лабораторные работы. Классы и объекты в C++. Наследование и виртуальные функции в C++. Перегрузка операций в C++. Преобразование типов в C++. Стандартная библиотека шаблонов. Динамические структуры данных.	24	3
	Самостоятельная работа обучающихся. Синтаксический сахар. Написание классов для интерфейса парсера.	12	3
МДК.01.02 Поддержка и тестирование программных модулей		84	
	4 семестр		
Раздел 1	Тестирование программных модулей		
Тема 1.1 Тестирование на этапах жизненного цикла проекта	Содержание учебного материала		
	Лекционные занятия. Основные понятия и определения тестирования. Тестирование на этапах жизненного цикла проекта.	2	1
	Самостоятельная работа обучающихся. Стандарты, регламентирующие процесс тестирования.	7	3
Тема 1.2 Виды ошибок	Содержание учебного материала		
	Лекционные занятия. Виды ошибок. Методы отладки.	4	1
	Практические занятия. Основы тестирования и классификация ошибок. Интуитивное тестирование.	4	2
Тема 1.3 Отчет об ошибке	Содержание учебного материала		
	Лекционные занятия. Отчет об ошибке.	2	1
	Практические занятия. Составление отчета об ошибке.	4	2
Тема 1.4 Проектирование и разработка тестов	Содержание учебного материала		
	Лекционные занятия. Классификация тестирования по уровням. Характеристики хорошего теста. V-модель разработки ПО. Позитивные и негативные тесты. Тестирование производительности программного обеспечения. Методы разработки тестов. Тестирование "белым ящиком". Методы разработки тестов. Тестирование "черным ящиком". Модульное тестирование. Интеграционное тестирование. Регрессионное тестирование. Рефакторинг кода. Оптимизация программ. Инструментальные средства для отладки и оптимизации программного обеспечения.	26	1

	Практические занятия. Тестирование "белым ящиком". Тестирование "черным ящиком". Автоматизация процесса тестирования программного продукта. Создание unit тестов. Автоматическое тестирование Selenium. Проведение нагрузочного тестирования.	24	2
	Самостоятельная работа обучающихся. Методики разработки тестов. Автоматизация процесса тестирования программного продукта.	11	3
МДК.01.03 Разработка мобильных приложений		130	
	4 семестр		
Раздел 1	Обзор платформ (ОС) для мобильных устройств и средств разработки под различные платформы		
Тема 1.1 Роль мобильных устройств в современной информатике.	Содержание учебного материала		
	Лекционные занятия. Понятие мобильного устройства. Классификация мобильных устройств. Рынок мобильных устройств. Тенденции информатики и ИТ, связанные с развитием мобильных устройств. Обзор современных планшетных устройств. Мобильные операционные системы. Особенности и проблемы, связанные с разработкой приложений для мобильных устройств. История развития мобильных устройств. Обзор современных смартфонов.	6	1
	Практические занятия. Вызов Активности с помощью явного намерения и получение результатов работы. Использование неявных Намерений. Получение данных из Намерения.	6	2
	Лабораторные работы. Отслеживание состояний Активности. Использование значений строк и цветов.	8	3
	Самостоятельная работа обучающихся. Каково устройство платформы Android. Что представляет собой Android SDK. Назовите основные средства разработки под Android. Перечислите достоинства и недостатки эмуляторов Android.	8	3
Тема 1.2 Основы работы в ОС Android	Содержание учебного материала		
	Лекционные занятия. ОС Android. Архитектура Android. Уровень ядра. Уровень библиотек. Уровень каркаса приложений. Уровень приложений. Особенности языка Java. Простейшее приложение для Android. Структура Android-приложения. История ОС Android. Особенности различных версий Android. Работа в Android. Инструменты разработки приложений для Android. Android SDK. Настройка Eclipse для Android. Работа с виртуальными Android-устройствами.	6	1
	Практические занятия. Использование SharedPreferences для сохранения	6	2

	состояния. Использование SharedPreferences для сохранения настроек. Работа с файлами.		
	Лабораторные работы. Локализация приложения. Использование анимации.	8	3
	Самостоятельная работа обучающихся. Выясните объем продаж мобильных устройств с ОС Android. Какая версия платформы наиболее популярна в настоящее время. Перечислите основные виды Android-приложений. Перечислите четыре различных типа компонентов.	8	3
Тема 1.3 Разработка пользовательского интерфейса для мобильных приложений	Содержание учебного материала		
	Лекционные занятия. Особенности разработки пользовательского интерфейса для мобильных устройств. Сенсорные технологии. Концепция трех экранов. Понятие компоновки. Типы компоновок в Android. Использование XML для описания пользовательского интерфейса. Понятие виджета. Обзор базовых виджетов. Обработка событий. Виджеты списки и привязка данных. Текстовые поля. Полосы прокрутки. Виджеты для отображения графики. Кнопки и флажки. Индикаторы, слайдеры и компоненты для отображения времени. Всплывающие уведомления. Создание собственных всплывающих уведомлений. Диалоги. Создание диалоговых окон. Меню.	6	1
	Практические занятия. Дочерние и контекстные меню. Создание и использование меню. Работа с СУБД SQLite.	6	2
	Лабораторные работы. Использование LinearLayout. Использование RelativeLayout.	8	3
	Самостоятельная работа обучающихся. Опишите иерархию классов Android SDK. Опишите иерархию компонентов, определяющая компоновку интерфейса пользователя. Опишите механизм передачи намерений. Опишите жизненный цикл активности.	14	3
Тема 1.4 Компоненты мобильных приложений	Содержание учебного материала		
	Лекционные занятия. Основные компоненты Android-приложения. Компонент Activity. Процессы в Android. Состояния Activity. Запуск собственных и системных Activity. Объекты Intent. Обмен данными между Activity. Компонент Service. Работа служб в Android. Создание службы. Вызов системной службы. Компонент Broadcast Receiver. Передача и прослушивание событий. Отслеживание системных событий. Компонент Content Provider. База данных SQLite. Инструменты для работы с БД в Android. Создание БД. Управление БД из приложения. Запросы к Content Provider.	6	1
	Практические занятия. Работа с SQLite с классом-адаптером. Работа с SQLite с	16	2

	классом-адаптером. Получение списка контактов.		
	Лабораторные работы. Использование TabWidget. Использование WebView.	4	3
	Самостоятельная работа обучающихся. Опишите жизненный цикл сервиса.	2	3
Тема 1.5 Дополнительные возможности Android.	Содержание учебного материала		
	Лекционные занятия. Файловый ввод-вывод. Пользовательские настройки. Ресурсы, используемые в Android-приложениях. Активы. Стили и темы. Локализация приложений. Работа с графикой и анимацией. Работа с системными компонентами и сетевыми сервисами Android. Получение информации о телефоне. Обработка телефонных вызовов. Работа с SMS. Мобильный интернет. Средства геолокации. Работа с оборудованием мобильного устройства.	10	1
	Практические занятия. Создание контент-провайдера. Использование сетевых сервисов.	14	2
	Лабораторные работы. Использование ListView. Использование управляющих элементов в пользовательском интерфейсе.	4	3
МДК.01.04 Системное программирование		120	
	4 семестр		
Раздел 1	Системное программирование		
Тема 1.1 Методология проектирования программных продуктов	Содержание учебного материала		
	Лекционные занятия. Знакомство с Qt. Модули Qt. Структура проекта. Основные типы.	4	1
	Практические занятия. Знакомство со средой Qt Creator и методами создания приложений. Знакомство со средой Qt Creator и методами создания приложений с графическим интерфейсом.	4	2
Тема 1.2 Системное программирование для различных ОС	Содержание учебного материала		
	Лекционные занятия. Работа с текстовыми строками в Qt. Класс QString. Списки строк QStringList. Контейнерные классы в Qt. Работа с файлами. Создание графического интерфейса средствами Qt. Виджеты (Widgets). Классы компоновки (Layouts). Сигнально-слотовые соединения.	12	1
	Практические занятия. Работа с контейнерами в среде Qt Creator: QVector. Использование контейнера QVector для решения задач обработки одномерных массивов. Работа с контейнерами в среде Qt Creator: QVector для обработки многомерных массивов. Использование контейнера QVector для решения задач обработки многомерных массивов. Работа с контейнерами в среде Qt Creator: QList. Работа с контейнерами в среде Qt Creator: QLinkedList. Работа с	20	2

	контейнерами в среде Qt Creator: QStack, QQueue. Работа с контейнерами в среде Qt Creator: QStack, QQueue. Решение задач. Работа с контейнерами в среде Qt Creator: QMap. Работа с контейнерами в среде Qt Creator: QSet.		
Тема 1.3 Объектно-ориентированное программирование	Содержание учебного материала		
	Лекционные занятия. Элементы графического интерфейса и их использование. Собственные классы в Qt. Создание элементов графического интерфейса Класс QObject. Управление памятью. Иерархии объектов. События (Events). Обработка событий (Event handling). Создание собственного элемента интерфейса. Создание свойств. Разработка приложений с графическим интерфейсом Окона. Класс QMainWindow.	12	1
	Практические занятия. Виджеты и компоновки. Разработка приложения «Текстовый редактор». Обработка событий в Qt. Работа с файлом ресурсов в Qt. Сигналы и слоты.	8	2
	Лабораторные работы. Создание простейшего графического приложения. Табулирование функции. Разработка приложения с графическим интерфейсом для обработки одномерных массивов. Разработка приложения с графическим интерфейсом для обработки строковых данных. Разработка приложения с графическим интерфейсом для обработки файлов.	16	3
	Самостоятельная работа обучающихся. Концепции ООП.	25	3
Тема 1.4 Стандарты языка C++	Содержание учебного материала		
	Лекционные занятия. Стандартные диалоги. Ресурсы программы. Создание собственных диалогов. Сохранение настроек. Многопоточность в приложении.	6	1
	Самостоятельная работа обучающихся. Стандарты C++ 17 и C++20.	13	3
Учебная практика Виды работ: работа с документацией, анализ предметной области. составление плана управления проектом, оценка трудозатрат проекта ознакомление с инструментальными средствами разработки ПО оформление технических документов		108	
Производственная практика Виды работ: разработка спецификаций классов, функций, алгоритмов, разработка программных модулей разработка информационно-логических моделей и алгоритмов программного проекта анализ и обоснование выбора виртуальной среды реализации программ		36	

<p>работа в средах визуального программирования согласно индивидуальному заданию</p> <p>разработка руководства пользователя, справочной системы</p> <p>оформление отчета</p>		
<p>Примерная тематика курсовых работ / индивидуальных проектов:</p> <ol style="list-style-type: none"> 1. Программа сравнения методов сортировки 2. Решение систем уравнений матричным методом 3. Калькулятор комплексных чисел 4. Перевод из римской системы счисления в десятичную 5. Решение систем уравнений методом Крамера 6. Шифрование текста методом замены, шифром Цезаря 7. Инженерный калькулятор 8. Калькулятор над числами в различных системах счисления 9. Построитель графиков функций 10. Логический калькулятор 11. Построение частотного словаря по тексту 12. Цифровые часы с будильником 13. Стрелочные часы с будильником 14. Словарь терминов 15. Морской бой 16. Поиск экстремумов функции 17. Игра Пятнашки 18. Автоматическое составление характеристик на студентов 19. Подсчет материалов для ремонта 20. Формирование открыток 21. Графический редактор 22. Поиск синонимов слов 23. Калькулятор комплексных чисел 24. Игра крестики-нолики 25. Матричный калькулятор 26. Проверка орфографии 27. Игра Память 28. Клавиатурный тренажер 29. Система анкетирования студентов 30. Расчет интегралов функции 31. Разработка мобильного приложения «Расписание студента» 	16	

32. Разработка мобильного приложения для парикмахерской			
33. Разработка мобильного приложения для кредитной организации			
34. Разработка мобильного приложения для управляющей компании ЖКХ			
35. Разработка мобильного приложения для автосалона			
36. Разработка мобильного приложения для автомастерской			
37. Разработка мобильного приложения для компании, занимающейся ИТ для службы поддержки			
38. Разработка мобильного приложения для компании по оказанию бытовых услуг			
39. Разработка мобильного приложения для социального работника			
40. Разработка мобильного приложения «Прогноз погоды»			
41. Разработка мобильного приложения «Курсы валют»			
42. Разработка мобильного приложения для автозаправки			
43. Разработка мобильного приложения аутсорсингом			
44. Разработка мобильного приложения для мебельного магазина			
45. Разработка мобильного приложения для агентства недвижимости			
46. Разработка мобильного приложения для книжного магазина			
47. Разработка мобильного приложения для интернет-магазина			
48. Разработка мобильного приложения «Студенческая группа»			
49. Разработка мобильного приложения «Зачетная книжка студента»			
50. Разработка мобильного приложения «Котировки на бирже»			
51. Разработка мобильного приложения для библиотеки			
52. Разработка мобильного приложения «Органайзер»			
53. Разработка мобильного приложения для автовокзала			
54. Разработка мобильного приложения для заказа товаров в сети Интернет			
55. Разработка мобильного приложения для учета личных расходов			
Квалификационный экзамен		18	
	Всего:	674	

4. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

4.1. Требования к минимальному материально–техническому обеспечению

Лаборатория разработки информационных систем

12 персональных компьютеров; проектор View Sonic PG603X DLP; экран настенный Lumien

Лекционная аудитория

Проектор ViewSonic PG603X DLP Экран Cactus Wallscreen

Компьютерный класс

Проектор ViewSonic PG603X DLP Экран Lumien Персональный компьютер RUSCO – 19 шт. Коммутатор D-Link Маршрутизатор беспроводной N ASUS RT-AC66U

Программное обеспечение:

LibreOffice (Mozilla Public License v2.0)

7-Zip (GNU LGPL)

Google Chrome (Лицензионное соглашение Google)

Mozilla Firefox (MPL)

РЕД ОС (Соглашение №140/05-21У от 18.05.2021 года о сотрудничестве в области науки, развития инновационной деятельности)

Microsoft Visual Studio (Программа Microsoft Azure Dev Tools for Teaching (Order Number: IM126433))

Pycharm Community Edition (проприетарная лицензия и Apache License 2.0)

Android Studio (Apache License 2.0)

Python 3.9.4 (Python Software Foundation License)

РЕД ОС (Соглашение №140/05-21У от 18.05.2021 года о сотрудничестве в области науки, развития инновационной деятельности)

Foxit Reader (Foxit EULA)

QT Creator ((L)GPL)

Python 3 (PSF License Agreement)

Double Commander (GNU GPL 2+)

4.2. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, интернет–ресурсов, дополнительной литературы.

Основные источники:

1. Косовская Т.М. Алгоритмы и анализ их сложности : учебное пособие / Косовская Т.М.. — Москва : Ай Пи Ар Медиа, 2023. — 111 с..
<https://www.iprbookshop.ru/125589.html>
2. Ю. В. Березовская, О. А. Юфрякова, В. Г. Вологодина [и др.]. Введение в разработку приложений для ОС Android : учебное пособие / — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 427 с. . <https://www.iprbookshop.ru/102000.html>
3. Пирская Л.В. Разработка мобильных приложений в среде Android Studio [Электронный ресурс]: учебное пособие/ Пирская Л.В.— Электрон. текстовые данные.— Ростов-на-Дону, Таганрог: Издательство Южного федерального университета, 2019.— 123 с. <http://www.iprbookshop.ru/100196.html>
4. Нужный, А. М. Разработка мобильных приложений на языке Java с использованием Android Studio : учебное пособие / А. М. Нужный, Н. И. Гребенникова, В. В. Сафронов. — Воронеж : Воронежский государственный технический университет, ЭБС АСВ, 2020. — 93 с. .
<https://www.iprbookshop.ru/111479.html>

Дополнительные источники:

1. Пирская Л.В. Разработка мобильных приложений в среде Android Studio [Электронный ресурс]: учебное пособие/ Пирская Л.В.— Электрон. текстовые данные.— Ростов-на-Дону, Таганрог: Издательство Южного федерального университета, 2019.— 123 с.. <http://www.iprbookshop.ru/100196.html>.— ЭБС «IPRbooks»

Интернет-ресурсы:

1. Справочник по Android Studio - <http://easyandroid.ru/>
2. Документация по C++ - <https://learn.microsoft.com/ru-ru/cpp/cpp/?view=msvc-170>
3. Документация по Qt Creator - <https://doc.qt.io/qtcreator/index.html>

4.3. Общие требования к организации образовательного процесса

Освоению модуля «Разработка модулей программного обеспечения для компьютерных систем» должно предшествовать изучение дисциплин «Информатика», «Введение в специальность», «Теория алгоритмов», «Основы программирования». При подготовке к квалификационному экзамену обучающимся оказываются консультации. Во время самостоятельной подготовки учащимся должен быть предоставлен доступ в Интернет. Требования к учебно-методической документации: наличие рекомендаций к выполнению лабораторных, практических и самостоятельных работ.

4.4. Кадровое обеспечение образовательного процесса

Требования к квалификации педагогических (инженерно - педагогических) кадров, обеспечивающих обучение междисциплинарному курсу (курсам): наличие высшего профессионального образования, соответствующего профилю модуля «Разработка модулей программного обеспечения для компьютерных систем».

5. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

[illegible]

[illegible]

	<p>Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием;</p>	
ПК 1.3. Выполнять отладку программных модулей с	<p>знать: Основные этапы разработки программного обеспечения.; знать: Основные этапы разработки</p>	<p>тест, защита курсовой работы; тест, защита курсовой работы; тест,</p>

<p>использованием специализированных программных средств</p>	<p>программного обеспечения.; знать: Основные принципы отладки и тестирования программных продуктов. ; знать: Инструментарий отладки программных продуктов.; знать: Основные виды и принципы тестирования программных продуктов; знать: Инструментальные средства анализа алгоритма; знать: Методы организации рефакторинга и оптимизации кода; знать: Модели процесса разработки программного обеспечения; знать: Основные принципы процесса разработки программного обеспечения; знать: Основные подходы к интегрированию программных модулей; знать: Основы верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и</p>	<p>защита курсовой работы; тест; тест; тест; тест; тест; тест, защита курсовой работы; тест; тест; тест; тест; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест;</p>
--	--	--

[illegible]

	<p>алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием;</p>	
<p>ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода</p>	<p>знать: Основные этапы разработки программного обеспечения.; знать: Основные этапы разработки программного обеспечения.; знать: Основные принципы отладки и тестирования программных продуктов. ; знать: Инструментарий отладки программных продуктов.; знать: Основные виды и принципы тестирования программных продуктов; знать: Инструментальные средства</p>	<p>тест, защита курсовой работы; тест, защита курсовой работы; тест, защита курсовой работы; тест; тест; тест; тест; тест; тест, защита курсовой работы; тест; тест; тест; тест; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест, защита курсовой работы; тест;</p>

	<p>анализа алгоритма; знать: Методы организации рефакторинга и оптимизации кода; знать: Модели процесса разработки программного обеспечения; знать: Основные принципы процесса разработки программного обеспечения; знать: Основные подходы к интегрированию программных модулей; знать: Основы верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства</p>	<p>тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест;</p>
--	---	--

[illegible]

[illegible]

	<p>Основы верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием;</p>	
ПК 2.4. Осуществлять	знать: Основные этапы разработки	тест, защита курсовой

<p>разработку тестовых наборов и тестовых сценариев для программного обеспечения</p>	<p>программного обеспечения.; знать: Основные этапы разработки программного обеспечения.; знать: Основные принципы отладки и тестирования программных продуктов. ; знать: Инструментарий отладки программных продуктов.; знать: Основные виды и принципы тестирования программных продуктов; знать: Инструментальные средства анализа алгоритма; знать: Методы организации рефакторинга и оптимизации кода; знать: Модели процесса разработки программного обеспечения; знать: Основные принципы процесса разработки программного обеспечения; знать: Основные подходы к интегрированию программных модулей; знать: Основы верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование</p>	<p>работы; тест, защита курсовой работы; тест, защита курсовой работы; тест; тест; тест; тест; тест; тест, защита курсовой работы; тест; тест; тест; тест; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; защита курсовой работы; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; защита курсовой работы; тест;</p>
--	---	--

	<p>программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием;</p>	
--	--	--

Формы и методы контроля и оценки результатов обучения должны позволять проверять у обучающихся не только сформированность профессиональных компетенций, но и развитие общих компетенций и обеспечивающих их умений.

Результаты обучения (освоенные общие компетенции)	Основные показатели оценки результата	Формы и методы контроля и оценки
ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам	<p>знать: Основные этапы разработки программного обеспечения.; знать: Основные этапы разработки программного обеспечения.; знать: Основные принципы отладки и тестирования программных продуктов. ; знать: Инструментарий отладки программных продуктов.; знать: Основные виды и принципы тестирования программных продуктов; знать: Инструментальные средства анализа алгоритма; знать: Методы организации рефакторинга и оптимизации кода; знать: Модели процесса разработки программного обеспечения; знать: Основные принципы процесса разработки программного обеспечения; знать: Основные подходы к интегрированию программных модулей; знать: Основы</p>	<p>тест, защита курсовой работы; тест, защита курсовой работы; тест, защита курсовой работы; тест; тест; тест; тест; тест; тест, защита курсовой работы; тест; тест; тест; тест; тест; тест; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест;</p>

	<p> верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием; </p>	
--	---	--

<p>ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности</p>	<p>знать: Основные этапы разработки программного обеспечения.; знать: Основные этапы разработки программного обеспечения.; знать: Основные принципы отладки и тестирования программных продуктов. ; знать: Инструментарий отладки программных продуктов.; знать: Основные виды и принципы тестирования программных продуктов; знать: Инструментальные средства анализа алгоритма; знать: Методы организации рефакторинга и оптимизации кода; знать: Модели процесса разработки программного обеспечения; знать: Основные принципы процесса разработки программного обеспечения; знать: Основные подходы к интегрированию программных модулей; знать: Основы верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные</p>	<p>тест, защита курсовой работы; тест, защита курсовой работы; тест, защита курсовой работы; тест; тест; тест; тест; тест, защита курсовой работы; тест; тест; тест; тест; защита курсовой работы; тест; тест; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест;</p>
---	--	--

	<p>средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием;</p>	
ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие	<p>знать: Основные этапы разработки программного обеспечения.; знать: Основные этапы разработки программного обеспечения.; знать: Основные принципы отладки и тестирования программных продуктов. ; знать: Инструментарий отладки программных продуктов.; знать: Основные виды и принципы тестирования программных продуктов; знать: Инструментальные средства анализа алгоритма; знать: Методы организации рефакторинга и оптимизации кода; знать: Модели процесса разработки программного обеспечения; знать: Основные принципы процесса разработки программного обеспечения; знать: Основные подходы к интегрированию программных модулей; знать: Основы верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные</p>	<p>тест, защита курсовой работы; тест, защита курсовой работы; тест, защита курсовой работы; тест; тест; тест; тест; тест; защита курсовой работы; тест; тест; тест; тест; тест; защита курсовой работы; тест; тест; защита курсовой работы; тест; защита курсовой работы; тест; тест; защита курсовой работы; тест; защита курсовой работы; тест; тест; защита курсовой работы; тест; тест; защита курсовой работы; тест; тест; защита курсовой работы; тест;</p>

	<p>методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием;</p>	
ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с	<p>знать: Основные этапы разработки программного обеспечения.; знать: Основные этапы разработки программного</p>	<p>тест, защита курсовой работы; тест, защита курсовой работы; тест, защита курсовой</p>

<p>коллегами, руководством, клиентами</p>	<p>обеспечения.; знать: Основные принципы отладки и тестирования программных продуктов. ; знать: Инструментарий отладки программных продуктов.; знать: Основные виды и принципы тестирования программных продуктов; знать: Инструментальные средства анализа алгоритма; знать: Методы организации рефакторинга и оптимизации кода; знать: Модели процесса разработки программного обеспечения; знать: Основные принципы процесса разработки программного обеспечения; знать: Основные подходы к интегрированию программных модулей; знать: Основы верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля;</p>	<p>работы; тест; тест; тест; тест; тест; защита курсовой работы; тест; тест; тест; тест; тест; защита курсовой работы; тест; защита курсовой работы; тест; защита курсовой работы; тест; защита курсовой работы; тест; тест; защита курсовой работы; тест; тест; защита курсовой работы; тест;</p>
---	--	--

[illegible]

	<p>тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием;</p>	
ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей	<p>знать: Основные этапы разработки программного обеспечения.; знать: Основные этапы разработки программного обеспечения.; знать: Основные принципы отладки и тестирования программных продуктов. ; знать:</p>	<p>тест, защита курсовой работы; тест, защита курсовой работы; тест, защита курсовой работы; тест; тест; тест; тест; тест; тест; тест; защита курсовой работы; тест; тест;</p>

	<p>Инструментарий отладки программных продуктов.; знать: Основные виды и принципы тестирования программных продуктов; знать: Инструментальные средства анализа алгоритма; знать: Методы организации рефакторинга и оптимизации кода; знать: Модели процесса разработки программного обеспечения; знать: Основные принципы процесса разработки программного обеспечения; знать: Основные подходы к интегрированию программных модулей; знать: Основы верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного</p>	<p>тест; тест; тест; тест; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест;</p>
--	---	--

[illegible]

	<p>уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием;</p>	
ОК 09. Использовать информационные технологии в профессиональной деятельности	<p>знать: Основные этапы разработки программного обеспечения.; знать: Основные этапы разработки программного обеспечения.; знать: Основные принципы отладки и тестирования программных продуктов. ; знать: Инструментарий отладки программных продуктов.; знать: Основные виды и принципы тестирования программных</p>	<p>тест, защита курсовой работы; тест, защита курсовой работы; тест, защита курсовой работы; тест; тест; тест; тест; защита курсовой работы; тест; тест; тест; тест; защита курсовой работы; тест; тест; тест; тест; защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой</p>

	<p>продуктов; знать: Инструментальные средства анализа алгоритма; знать: Методы организации рефакторинга и оптимизации кода; знать: Модели процесса разработки программного обеспечения; знать: Основные принципы процесса разработки программного обеспечения; знать: Основные подходы к интегрированию программных модулей; знать: Основы верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной</p>	<p>работы; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест;</p>
--	---	--

	функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием;	
ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке	знать: Основные этапы разработки программного обеспечения.; знать: Основные этапы разработки программного обеспечения.; знать: Основные принципы отладки и тестирования программных продуктов. ; знать: Инструментарий отладки программных продуктов.; знать: Основные виды и принципы тестирования программных продуктов; знать: Инструментальные средства анализа алгоритма; знать: Методы организации рефакторинга и оптимизации кода; знать: Модели процесса разработки программного обеспечения; знать: Основные принципы процесса разработки программного обеспечения; знать: Основные подходы к интегрированию программных модулей; знать: Основы верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь:	тест, защита курсовой работы; тест, защита курсовой работы; тест, защита курсовой работы; тест; тест; тест; тест; защита курсовой работы; тест; тест; тест; тест; тест; защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест;

	<p>Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием;</p>	
ОК 11. Планировать предпринимательскую деятельность в профессиональной сфере	<p>знать: Основные этапы разработки программного обеспечения.; знать: Основные этапы разработки программного обеспечения.; знать: Основные принципы отладки и тестирования программных продуктов. ; знать: Инструментарий отладки программных продуктов.; знать: Основные виды и принципы тестирования программных продуктов; знать: Инструментальные средства анализа алгоритма; знать: Методы организации</p>	<p>тест, защита курсовой работы; тест, защита курсовой работы; тест, защита курсовой работы; тест; тест; тест; тест; защита курсовой работы; тест; тест; тест; тест; защита курсовой работы; тест; тест; тест; тест, защита курсовой работы; тест; тест, защита курсовой работы; тест; защита курсовой работы; тест; тест, защита курсовой работы; тест;</p>

	<p>рефакторинга и оптимизации кода; знать: Модели процесса разработки программного обеспечения; знать: Основные принципы процесса разработки программного обеспечения; знать: Основные подходы к интегрированию программных модулей; знать: Основы верификации программного обеспечения; знать: Современные технологии и инструменты интеграции; знать: Основные методы отладки; знать: Методы и схемы обработки исключительных ситуаций; знать: Основные методы и виды тестирования программных продуктов; знать: Основные задачи сопровождения информационной системы; уметь: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.; уметь: Оценка сложности алгоритма; уметь: Создавать программу по разработанному алгоритму как отдельный модуль.; уметь: Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.; уметь: Выполнять отладку и тестирование программы на уровне модуля.; уметь: Применять инструментальные средства отладки программного обеспечения.; уметь: Выполнять отладку и тестирование программы на уровне модуля; уметь: Выполнять оптимизацию и рефакторинг программного кода; уметь: Осуществлять разработку кода программного модуля на современных языках программирования; уметь: Использовать методы для получения кода с заданной функциональностью и степенью качества.; уметь: Использовать инструментальные средства отладки программных</p>	<p>тест, защита курсовой работы; тест; тест; тест, защита курсовой работы; тест;</p>
--	--	--

	продуктов.; иметь практический опыт: владения современными программными средствами, предназначенными для разработки мобильных приложений ; иметь практический опыт: владение функциональным программированием;	
--	--	--

Фонд оценочных материалов (средств) по дисциплине
Разработка модулей программного обеспечения для компьютерных систем

1. Оценочные материалы для проведения текущего контроля успеваемости по дисциплине

Дисциплина Разработка мобильных приложений:

Задание 1. Скачать Android SDK + Eclipse (Eclipse ADT Bundle)

<http://developer.android.com/sdk/index.html>, а также последнюю версию java (JDK) <http://www.oracle.com/technetwork/java/index-jsp-138363.html> (или просто набрать в yandex "JDK" и пройти по ссылке на oracle.com). Установить всё это. Создать новый проект, зайти в Android SDK и скачать какую-нибудь версию Android SDK (например, lollipop - 5.0.1), выделив все инструменты для работы с ней; сконфигурировать эмулятор

(желательно эмулировать своё собственное мобильное устройство, включая версию андроид на нём в качестве Target SDK). Изменить TextView с надписью "Hello world" на какую-нибудь другую надпись по желанию. Запустить проект на эмуляторе и убедиться, что всё работает.

Задание 2. Создать новый проект, написать программу, которая выводит в элемент TextView надпись, введенную пользователем в текстовом поле EditText после нажатия на кнопку Button. Помимо этого в Activity должен быть TextView с ФИО студента и группой. Запустить на эмуляторе и убедиться, что всё работает.

Задание 3. Создать приложение, которое состоит из нескольких activities. Первое activity содержит элемент TextView с названием или номером activity, текстовое поле EditText для ввода какой-то информации, кнопку Button с названием "Next" или "Перейти на 2 activity/экран/окно" или просто "2". Помимо этого в 1 activity должен быть TextView с ФИО студента и группой. После нажатия на эту кнопку происходит переход на второе activity, где содержится TextView с названием или номером activity, TextView

с надписью что-то вроде "В первом окне вы напечатали:" и под ним - ещё один TextView

с содержимым EditText с первого activity, и, разумеется, кнопка "1" или "Вернуться на 1 экран" или "Вернуться к вводу текста", нажав на которую пользователь может перейти обратно к 1 activity. Запустить на эмуляторе и убедиться, что всё работает.

Задание 4. В новом проекте написать приложение, работающее с разными темами/стилями. Сначала создать свой стиль и применить его к какому-нибудь интерфейсному элементу, затем - свою тему, которая применяется ко всем интерфейсным элементам. Приложение при этом должно выглядеть нестандартно. Запустить на эмуляторе и убедиться, что всё работает. При возникновении ошибок открыть лог (CatLog) внизу, найти первую красную надпись и породить свою тему от той, которая требуется в этом красном сообщении.

Задание 5. Создать пользовательский (свой) список. Например, получить доступ в приложении к контактам (Permissions-закладка в AndroidManifest.xml) и скопировать контакты телефона в свой список, который отобразить после запуска приложения. Или создать свой список в виде твиттера (картинка+текст), элементы которого просто статически задать в массиве (как и картинки).

Задание 6. Создать приложение, содержащее анимированные интерфейсные элементы (например, увеличивающиеся при клике на них кнопки, вращающиеся TextView и т.д.).

Задание 7. Создать приложение, отображающее после запуска карты Google или какие-нибудь другие карты.

Задание 8. Создать собственный виджет с настройками. Например, виджет, который открывает какой-то сайт (по лекции), адрес которого можно поменять в настройках.

Задание 9. Создать приложение, использующее опциональное меню (меню настроек) и контекстное меню для какого-нибудь интерфейсного элемента. Естественно, выбор

пунктов меню должен что-то менять в интерфейсных элементах или их отображении!
Например, очистить поле ввода через контекстное меню, или отобразить невидимые интерфейсные элементы через установку галок в опциональном меню.

Задание 10. Создать приложение, отображающее после некоторых действий (нажатия на кнопку, например, или проверки корректности ввода текста в EditText) диалоговое окно, свидетельствующее об ошибке/информирующее/предупреждающее пользователя о чём то.

Задание 11. Создать приложение, помещающее по нажатию на кнопку какое-то сообщение со звуком в панель уведомлений/статус-панель на эмуляторе.

Задание 12. Создать приложение, по нажатию кнопки в котором проигрывается какой-то звук.

Задание 13. Создать приложение, при запуске которого проигрывается какое-то видео.

Задание 14. Создать приложение, при запуске которого активируется фотокамера телефона, производится снимок, и этот снимок помещается в ImageView интерфейса приложения.

Задание 15. Создать приложение, работающее с SharedPreferences и сохраняющее настройки, а также работающее с БД SQLite - заполняющее БД по нажатию кнопки 1 с помощью EditText, и выводящее все записи этой БД в какой-нибудь интерфейсный элемент ниже с помощью кнопки 2 (в виде списка, datagrid или просто правильно настроенного TextView).

Регламент проведения защиты и оценивание курсовой работы (проекта)

Защита курсовой работы проводится на заседании Комиссии. На защите кроме руководителя курсовой работы присутствует один или два члена комиссии.

может присутствовать научный руководитель курсовой работы.

Защита курсовой работы осуществляется в устной форме с показом презентации на проекторе.

Продолжительность защиты, как правило, не превышает 20 минут.

Для доклада основных положений курсовой работы, обоснования выводов и предложений студенту предоставляется не более 5-7 минут.

После доклада студент должен ответить на замечания научного руководителя, а также на заданные членами Комиссии вопросы по теме курсовой работы.

По результатам защиты выставляется оценка, определяемый оценками: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Оценка курсовой работы даётся членами Комиссии по защите курсовых работ на её закрытом заседании. Комиссией принимается во внимание содержание работы, обоснованность выводов и предложений, содержание доклада студента, отзыв научного руководителя, уровень теоретической и практической подготовки

студента, а также соблюдение требований по порядку оформления работы. Отзыв научного руководителя является важным, но не доминирующим фактором в определении оценки.

Оценки курсовых работ объявляются в тот же день после оформления в установленном порядке протокола заседания Комиссии. После защиты

курсовая работа хранится на профильной кафедре в течение 3 (трех) лет. Если защита курсовой работы признана неудовлетворительной, то

Комиссия устанавливает, может ли студент представить к повторной защите ту же работу с соответствующей доработкой, или должен написать новую.

При оценке курсовой работы учитываются следующие факторы: степень самостоятельности выполнения работы, новизна выводов и конструктивность предложений, качество используемого материала, уровень грамотности (общий и специальный), а также порядок оформления.

Общими критериями оценки качества курсовой работы являются: соответствие содержания курсовой работы, по которой она выносится на защиту;

соответствие требованиям, предъявляемым к форме и содержанию; уровень защиты курсовой работы.

Оценка «отлично» выставляется за курсовую работу, которая имеет грамотно изложенный теоретический раздел, характеризуется логичным и последовательным изложением материала с соответствующими выводами и обоснованными предложениями по практическому применению результатов исследования. Такая работа должна иметь положительный отзыв научного руководителя. При её защите студент показывает глубокие знания вопросов темы, свободно оперирует данными исследования, вносит обоснованные предложения по практическому применению результатов исследования, четко отвечает на поставленные вопросы.

Оценка «хорошо» выставляется за работу, которая, имеет грамотно изложенный теоретический раздел, характеризуется логичным и последовательным изложением материала, однако имеет не вполне обоснованные выводы и не имеет предложений по практическому применению результатов исследования.

Работа имеет в целом положительный отзыв научного руководителя, но содержит ряд незначительных замечаний. При её защите студент показывает знание вопросов темы, оперирует данными исследования, без особых затруднений отвечает на поставленные вопросы.

Оценка «удовлетворительно» выставляется за работу, которая характеризуется непоследовательностью в изложении материала.

Представленные выводы автора необоснованны. В отзыве научного руководителя имеются серьёзные замечания по содержанию работы и

методике анализа. При её защите студент проявляет неуверенность, показывает слабое знание вопросов темы, не даёт полного аргументированного ответа на заданные вопросы.

Оценка «неудовлетворительно» выставляется за работу, которая не отвечает требованиям, изложенным в учебно-методических указаниях по выполнению курсовых работ. В курсовой работе нет выводов, либо они носят декларативный характер. В отзыве научного руководителя имеются серьезные замечания. При защите курсовой работы студент затрудняется отвечать на поставленные вопросы по теме, не знает теории вопроса, при ответе допускает существенные ошибки. К защите не подготовлена презентация.

Оценка «неудовлетворительно» может быть также выставлена студенту, представившему на защиту чужую курсовую работу, написанную и уже защищенную в другом вузе или на другой кафедре. Подобные работы вообще не принимаются к рассмотрению членами комиссии, а студент обязан разработать новую тему, которая определяется профильной кафедрой.

Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	Лабораторные и практические работы	20
Рейтинг-контроль 2	Лабораторные и практические работы	20
Рейтинг-контроль 3	Лабораторные и практические работы	20
Посещение занятий студентом		10

Дополнительные баллы (бонусы)		10
Выполнение семестрового плана самостоятельной работы		20

2. Промежуточная аттестация по дисциплине

Перечень вопросов к экзамену / зачету / зачету с оценкой.

Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

Оценочные средства для промежуточной аттестации приведены в приложении 1.

Методические материалы, характеризующих процедуры оценивания

Раздел 1. МДК.01.01 Разработка мобильных приложений

1) Набор средств программирования, который содержит инструменты, необходимые для создания, компиляции и сборки мобильного приложения называется:

- а) Android SDK
- б) JDK
- в) плагин ADT
- г) Android NDK

2) С какой целью был создан Open Handset Alliance?

- А) писать историю развития ОС Android
- б) продавать смартфоны под управлением Android
- в) рекламировать смартфоны под управлением Android
- г) разрабатывать открытые стандарты для мобильных устройств

3) С какой целью инструмент Intel* Graphics Performance Analyzers (Intel* GPA) System Analyzer используется в среде разработки Intel* Beacon Mountain?

а) позволить разработчикам оптимизировать загрузженность системы при использовании процедур OpenGL

- б) для ускорения работы эмулятора в среде разработки
- в) для оптимизированной обработки данных и изображений
- г) позволить разработчикам эффективно распараллелить C++ мобильные приложения

3) Библиотеки, реализованные на базе PacketVideo OpenCORE:

- А) Media Framework
- Б) SQLite
- В) FreeType
- Г) 3D библиотеки

4) Какой движок баз данных используется в ОС Android?

- А) InnoDB
- Б) DBM
- В) MyISAM
- Г) SQLite

5) С какой целью инструмент Intel* Integrated Performance Primitives (Intel* IPP) используется в среде разработки Intel* Beacon Mountain?

А) для оптимизированной обработки данных и изображений
Б) позволить разработчикам оптимизировать загрузженность системы при использовании процедур OpenGL

В) для ускорения работы эмулятора в среде разработки

Г) позволить разработчикам эффективно распараллелить C++ мобильные приложения

б) Intel XDK поддерживает разработку под:

- A) JavaFX Mobile
 - Б) Apple iOS, BlackBerry OS
 - В) MtkOS, Symbian OS, Microsoft Windows 8
 - Г) Android, Apple iOS, Microsoft Windows 8, Tizen
- 7) Каждый приемник широковещательных сообщений является наследником класса ...
- A) ViewReceiver
 - Б) IntentReceiver
 - В) ContentProvider
 - Г) BroadcastReceiver
- 8) Какой класс является основным строительным блоком для компонентов пользовательского интерфейса (UI), определяет прямоугольную область экрана и отвечает за прорисовку и обработку событий?
- A) GUI
 - Б) View
 - В) UIComponent
 - Г) Widget
- 9) Какой слушатель используется для отслеживания события касания экрана устройства?
- A) OnPressListener
 - Б) onTouchListener
 - В) OnClickListener
 - Г) OnInputListener
- 10) В какой папке необходимо разместить XML файлы, которые определяют все меню приложения?
- A) res/value
 - Б) res/items
 - В) res/layout
 - Г) res/menu
- 11) Фоновые приложения ...
- A) после настройки не предполагают взаимодействия с пользователем, большую часть времени находятся и работают в скрытом состоянии
 - Б) выполняют свои функции и когда видимы на экране, и когда скрыты другими приложениями
 - В) небольшие приложения, отображаемые в виде графического объекта на рабочем столе
 - Г) большую часть времени работают в фоновом режиме, однако допускают взаимодействие с пользователем и после настройки
- 12) Полный иерархический список обязательных файлов и папок проекта можно увидеть на вкладке ...
- A) Package Explorer
 - Б) Internet Explorer
 - В) Navigator
 - Г) Project Explorer
- 13) Какой компонент управляет распределенным множеством данных приложения?
- A) сервис (Service)
 - Б) активность (Activity)
 - В) приемник широковещательных сообщений (Broadcast Receiver)
 - Г) контент-провайдер (Content Provider)
- 14) Какой язык разметки используется для описания иерархии компонентов графического пользовательского интерфейса Android-приложения?
- A) html
 - Б) xml

- В) gml
- Г) xhtml

15) Выберите верную последовательность действий, необходимых для создания в приложении контент-провайдера.

А) Создание класса наследника от класса ContentProvider; Определение способа организации данных; Заполнение контент-провайдера данными

Б) Проектирование способа хранения данных; Определение способа организации данных;

В) Создание класса наследника от класса ContentProvider; Заполнение контент-провайдера данными; Определение способа работы с данными

Г) Проектирование способа хранения данных; Создание класса-наследника от класса ContentProvider; Определение строки авторизации провайдера, URI для его строк и имен столбцов

16) Выберите верные утверждения относительно объекта-намерения (Intent).

А) представляет собой структуру данных, содержащую описание операции, которая должна быть выполнена, и обычно используется для запуска активности или сервиса

Б) используется для передачи сообщений пользователю

В) используется для получения инструкций от пользователя

Г) используются для передачи сообщений между основными компонентами приложений

17) Расположение элементов мобильного приложения:

А) полезно для передачи иерархии

Б) влияет на удобство использования

В) полезно для создания пространственных отношений между объектами на экране и объектами реального мира

Г) все варианты ответа верны

18) Какие элементы управления применяются для действий по настройке?

А) командные элементы управления

Б) элементы выбора

В) элементы ввода

Г) элементы отображения

19) Примерами комбо-элементов не являются:

А) комбо-список

Б) все вышеперечисленное

В) комбо-кнопка

Г) комбо-поле

20) Дизайн или проектирование интерфейса для графических дизайнеров:

А) все варианты ответа верны

Б) прозрачность и понятность информации

В) тон, стиль, композиция, которые являются атрибутами бренда

Г) передача информации о поведении посредством ожидаемого назначения

21) Более крупные элементы:

А) привлекают больше внимания

Б) все варианты ответа верны

В) размер не влияет на уровень внимания

Г) привлекают меньше внимания

22) К традиционным типографическим инструментам не относят

А) масштаб

Б) цвет

В) разреженность

Г) выравнивание по сетке

23) К элементам ввода относят:

А) ограничивающие элементы ввода

Б) ползунки

В) счетчики

Г) все вышеперечисленное

24) Выделяют следующие категории плотности экрана для Android-устройств:

А) HDPI, XHDPI, XXHDPI, и XXXHDPI

Б) правильный вариант ответа отсутствует

В) LDPI, MDPI, HDPI, XHDPI, XXHDPI, и XXXHDPI

Г) LDPI, MDPI, HDPI

25) Следующие утверждения не верны:

А) не используйте интерфейсные элементы

Б) картинки работают быстрее, чем слова

В) на любом шаге должна быть возможность вернуться назад

Г) если объекты похожи, они должны выполнять сходные действия

26) Следующие утверждения верны:

А) текстура бесполезна для передачи различий или привлечения внимания

Б) восприятие направления затруднено при больших размерах объектов

В) все варианты ответа верны

Г) люди легко воспринимают контрастность

27) Основные вкладки (FixedTabs) удобны при отображении

А) от четырех вкладок

Б) двух вкладок

В) трех и более вкладок

Г) трех и менее вкладок

28) Диалоговое окно, содержащее линейку процесса выполнения какого-то действия —

это

А) DatePickerDialog

Б) AlertDialog

В) ProgressDialog

Г) DialogFragment

29) Уведомления стоит использовать, когда

А) сообщение не требует ответа пользователя, но важно для продолжения его работы

Б) сообщение является важным и требует немедленного прочтения и ответа

В) сообщение является важным, однако требует немедленного прочтения, но не ответа

Г) сообщение является важным, однако не требует немедленного прочтения и ответа

30) Какой метод запускает новую активность?

А) startActivity()

Б) beginActivity()

В) intentActivity()

Г) newActivity()

31) ProgressDialog это:

А) контейнер для создания собственных диалоговых окон

Б) диалоговое окно с предопределенным интерфейсом, позволяющее выбрать дату или

время

В) диалоговое окно, содержащее линейку процесса выполнения какого-то действия

Г) диалоговое окно, которое может содержать заголовок, до трех кнопок, список выбираемых значений или настраиваемое содержимое

32) AlertDialog это:

А) контейнер для создания собственных диалоговых окон

Б) диалоговое окно, содержащее линейку процесса выполнения какого-то действия

В) диалоговое окно, которое может содержать заголовок, до трех кнопок, список выбираемых значений или настраиваемое содержимое

Г) диалоговое окно с предопределенным интерфейсом, позволяющее выбрать дату или время

33) Что необходимо сделать при добавлении в проект новой активности?

А) скачать и установить специальный инструмент MultiActivity SDK

- Б) прописать в манифесте информацию о новой активности
 - В) создать новый проект
 - Г) запустить эмулятор
- 34) Системы позиционирования смартфона могут включать
- А) все перечисленное
 - Б) систему GPS
 - В) систему ГЛОНАСС
 - Г) сигналы WiFi и Bluetooth
- 35) Какая константа не определена в классе MotionEvent, для обозначения сенсорных событий
- А) ACTION_DOWN
 - Б) ACTION_UP
 - В) ACTION_CLICK
 - Г) ACTION_MOVE
- 36) С какой целью используется метод release() в классах MediaPlayer и MediaRecorder?
- А) конец жизненного цикла объекта и освобождение ресурсов
 - Б) перевод объекта в ожидающее состояние
 - В) обновление объекта и запуск его работы
 - Г) создание объекта и запуск его работы
- 37) К датчикам окружающей среды, встроенным в мобильное устройство относят
- А) датчики вектора вращения
 - Б) датчики освещенности
 - В) акселерометры
 - Г) гироскопы
- 38) Библиотека Universal Image Loader for Android позволяет:
- А) парсить HTML-страницы
 - Б) строить графики и диаграммы
 - В) загружать, кешировать и отображать изображения
 - Г) использовать анимацию, доступную только с версии 3.x, на более ранних вариантах платформы Android
- 39) Facebook SDK for Android — это библиотека, позволяющая:
- А) получать доступ к информации любого пользователя
 - Б) отправлять рекламные сообщения от имени пользователя
 - В) писать сообщения на стену, читать и менять статусы, смотреть ленту друзей
 - Г) парсить страницы пользователей
- 40) Что из перечисленного не относится к правилам безопасности при подключении библиотек?
- А) с осторожностью использовать библиотеки из сомнительных источников
 - Б) ознакомиться с форумами и сайтами, где могут обсуждаться библиотеки
 - В) лично познакомиться с разработчиками библиотеки
 - Г) использовать скомпрометированные библиотеки
- 41) Библиотека MapNavigator предназначена для:
- А) работы с любыми картами
 - Б) работы с Яндекс.Картами
 - В) морской навигации
 - Г) работы с картами Google Maps
- 42) Библиотека jsoup не позволяет:
- А) находить и извлекать данные, используя DOM и селекторы CSS
 - Б) манипулировать HTML-элементами, атрибутами и текстом
 - В) писать сообщения на стену, читать и менять статусы, смотреть ленту друзей
 - Г) принимать в качестве параметра URL, файл или строку
- 43) При настройке обратной совместимости необходимо добавить в файл манифеста следующую информацию:
- А) только минимальную версию Android SDK

- Б) минимальную и основную (целевую) версии Android SDK
 - В) информацию о подключенной библиотеке
 - Г) только основную (целевую) версии Android SDK
- 44) Какая библиотека предназначена для упрощения загрузки изображений?
- А) Yandex.Metrica for Apps;
 - Б) Universal Image Loader for Android
 - В) ActionBarSherlock
 - Г) NineOldAndroids
- 45) Библиотеки совместимости предназначены для
- А) сбора статистики
 - Б) рисования графиков
 - В) использования возможностей, появившиеся в какой-то версии ОС Android, на более ранних версиях платформы
 - Г) подключения нестандартных элементов управления
- 46) Какая библиотека предназначена для использования анимации?
- А) Universal Image Loader for Android
 - Б) NineOldAndroids
 - В) Yandex.Metrica for Apps
 - Г) ActionBarSherlock
- 47) Для чего служит папка res/anim/ проекта?
- А) в этой папке находятся файлы, содержащие набор картинок, предназначенных для кадровой анимации
 - Б) в этой папке находятся файлы, содержащие анимированные ролики для воспроизведения в приложении
 - В) в этой папке находятся XML файлы, задающие реализацию анимации свойств
 - Г) в этой папке находятся XML файлы, задающие последовательность инструкций анимации преобразований
- 48) В какой файл обязательно добавляется информация при создании нового Activity в приложении?
- А) AndroidManifest.xml
 - Б) main.java
 - В) layout.xml
 - Г) activity.xml
- 49) Какой метод жизненного цикла активности вызывается системой непосредственно перед появлением активности на экране?
- А) onVisible()
 - Б) onOpen()
 - В) onResume()
 - Г) onCreate()
- 50) С какой целью используется метод SurfaceHolder.lockCanvas()?
- А) блокировка Canvas для перерисовки
 - Б) игнорирование дальнейшего взаимодействия с Canvas
 - В) сокрытие Canvas
 - Г) блокировка Canvas от сворачивания
- 51) Может ли мобильное приложение получить доступ к базе данных, созданной в другом приложении?
- А) не может ни при каких обстоятельствах
 - Б) может, но только с помощью контент-провайдеров
 - В) право на доступ открывает приложение-хозяин базы данных
 - Г) может обращаться напрямую
- 52) С помощью какого метода можно запретить смену ориентации устройства, при запущенном приложении?
- А) setRequestedOrientation
 - Б) setChangeOrientation

- В) disableChangeOrientation
Г) setOrientation
- 53) Какой из датчиков не используется для определения положения смартфона в пространстве?
- А) акселерометр
 - Б) gps
 - В) гироскоп
 - Г) магнитометр
- 54) К новым возможностям HTML5 относят (выберите все верные варианты ответа):
- А) возможность добавления аудио и видео без использования вспомогательных средств
 - Б) возможность рисования на холсте
 - В) возможность прямого доступа к оперативной памяти
 - Г) форматирование данных в режиме таблицы
- 55) Возможен ли перенос приложений iOS* в среду HTML5:
- А) нет, прямой перенос приложений невозможен
 - Б) да, используя средства Intel XDK
 - В) да, используя только сторонние средства
 - Г) да, только для iPhone, используя средства Intel XDK
- 56) Следующие утверждения верны:
- А) JavaScript не позволяет подключать другие внешние библиотеки, написанные на других языках
 - Б) приложения html5 выполняются быстрее и требуют меньше ресурсов, чем «нативные»
 - В) среда Intel XDK не работает с мультисенсорностью
 - Г) приложения html5 выполняются медленнее и требуют больших ресурсов, чем «нативные»
- 57) Разрабатывать приложения в среде Intel XDK можно:
- А) пользоваться заготовленными примерами
 - Б) все варианты ответа верны
 - В) «с нуля», прописывая все элементы
 - Г) использовать встроенный «дизайнер элементов» для отрисовки элементов
- 58) JavaScript не позволяет:
- А) получать прямой доступ к памяти
 - Б) работать с реестром
 - В) работать с картами
 - Г) одновременно использовать несколько подключаемых библиотек
- 59) В среде Intel XDK можно разрабатывать приложения для следующих платформ:
- А) Android
 - Б) все варианты ответа верны
 - В) Apple iOS
 - Г) Tizen
- 60) В заготовке любого приложения, разрабатываемого в среде Intel XDK прописана:
- А) все варианты ответа верны
 - Б) скрытие окна заставки Intel XDK
 - В) настройка размеров приложения под размеры устройства
 - Г) фиксация размеров приложения (запрет «скроллинга»)
- 61) Создавать и редактировать пользовательский интерфейс приложений в среде Intel XDK можно:
- А) используя встроенное приложение App Designer
 - Б) только изменяя готовые шаблоны с интерфейсом
 - В) все варианты ответа неверны
 - Г) только прописывая теги вручную
- 62) Удобное средство обмена между двумя NFC-устройствами:
- А) Wi-Fi Direct

- Б) AndroidBeam
 - В) Dalvik
 - Г) Bluetooth
- 63) Переключения между активностями осуществляются
- А) только при помощи кнопок
 - Б) только с использованием сенсорного экрана смартфона
 - В) только при помощи кнопок и других элементов управления
 - Г) все три варианта возможны

Раздел 2. МДК.01.02 Прикладное программирование

1. Язык программирования C++ разработал

- 1. Дональд Кнут
- 2. Кен Томпсон
- 3. Никлаус Вирт
- 4. Бьерн Страуструп

2. Какому зарезервированному слову программа передаёт управление в случае, если значение переменной или выражения оператора switch не совпадает ни с одним константным выражением?

- 1. all
- 2. contingency
- 3. default
- 4. other

3. Какие среды программирования (IDE) предназначены для разработки программных средств?

- 1. MVS, Code::Blocks, QT Creator, AutoCAD, Eclipse
- 2. MVS, Code::Blocks, QT Creator, RAD Studio, MathCAD
- 3. MVS, NetBeans, QT Creator, RAD Studio, Dev-C++

4. Укажите правильную форму записи цикла do while

- 1.


```
// форма записи оператора цикла do while:
do // начало цикла do while
{
/*блок операторов*/;
}
while (/*условие выполнения цикла*/) // конец цикла do while
```
- 2.

```
// форма записи оператора цикла do while:
do // начало цикла do while
{
/*блок операторов*/;
}
while (/*условие выполнения цикла*/); // конец цикла do while
```
- 3.

```
// форма записи оператора цикла do while:
do // начало цикла do while
{
/*блок операторов*/;
}
while {/*условие выполнения цикла*/} // конец цикла do while
```


5. Укажите объектно-ориентированный язык программирования

1. Eiffel
2. Java
3. Все варианты ответов
4. C++

6. Цикл с постусловием?

1. for
2. do while
3. while

7. Цикл с предусловием?

1. do while
2. for
3. while

8. Программа, переводящая входную программу на исходном языке в эквивалентную ей выходную программу на результирующем языке, называется:

1. компилятор
2. транслятор
3. сканер
4. интерпретатор

9. Простые типы данных в C++.

1. целые – int, вещественные – float или real, символьные – char
2. целые – bool, вещественные – float или double, символьные – string
3. целые – int, вещественные – float или double, символьные – string
4. целые – int, вещественные – float или double, символьные – char

10. Структура объявления переменных в C++

1. [=], идент. 2,...;
2. [=]; идент. 2,...;
3. [:=], идент. 2,...;
4. [==]; идент. 2,...;

11. Какая из следующих записей - правильный комментарий в C++?

1. /* комментарий */
2. {комментарий}
3. */ Комментарий */
4. ** Комментарий **

12. Какой оператор не допускает перехода от одного константного выражения к другому?

1. точка с запятой
2. Stop;
3. end;
4. break;

13. Какие служебные символы используются для обозначения начала и конца блока кода?

1. begin end
2. { }
3. '<' '>'

4. ()

14. Чтобы подключить заголовочный файл в программу на C++, например `iostream` необходимо написать:

1. `include #iostream,h;`
2. `"#include <>; с iostream.h внутри скобок"`
3. `"#include <> с iostream внутри скобок"`
4. `include (iostreamh)`

15. Тело любого цикла выполняется до тех пор, пока его условие ...

1. ложно
2. истинно
3. у цикла нет условия

16. Общий формат оператора множественного выбора - `switch`

1.

```
switch (switch_expression)
{
    case constant1: statement1; [break;]
    case constant2: statement2; [break;]
    case constantN: statementN; [break;]
    [default: statement N+1;]
}
```

2.

```
switch (switch_expression)
{
    case constant1: statement1; [break;]
    case constant2: statement2; [break;]
    case constantN: statementN; [break;]
    [else: statement N+1;]
}
```

3.

```
switch (switch_expression)
{
    case constant1, case constant2: statement1; [break;]
    case constantN: statementN; [break;]
    [default: statement N+1;]
}
```

17. Какую функцию должны содержать все программы на C++?

1. `start()`
2. `main()`
3. `system()`
4. `program()`

18. Какой из ниже перечисленных операторов, не является циклом в C++?

1. `repeat until`
2. `while`
3. `do while`
4. `for`

19. Какой из следующих операторов - оператор сравнения двух переменных?

1. `:=`
2. `=`

- 3. equal
- 4. ==

20. Чему будет равна переменная a, после выполнения этого кода `int a; for(a = 0; a < 10; a++) {}`?

- 1. 10
- 2. 9
- 3. 1

21. Каков результат работы следующего фрагмента кода?

```
int x = 0;
```

```
switch(x)
{
    case 1: cout << "Один";

    case 0: cout << "Нуль";

    case 2: cout << "Привет мир";

}
```

- 1. Один
- 2. Нуль
- 3. НульПривет мир
- 4. Привет мир

22. Какой служебный знак ставится после оператора case ?

- 1. .
- 2. :
- 3. ;
- 4. -

23. Выберите правильный вариант объявления константной переменной в C++, где type - тип данных в C++ variable- имя переменной value - константное значение

- 1. `const type variable = value;`
- 2. `const type variable := value;`
- 3. `const variable = value;`

24. Какими знаками заканчивается большинство строк кода в Си++?

- 1. , (запятая)
- 2. . (точка)
- 3. ; (точка с запятой)
- 4. : (двоеточие)

25. Какой из перечисленных типов данных не является типом данных в C++?

- 1. double
- 2. real
- 3. int
- 4. float

26. Какое значение, по умолчанию, возвращает программа операционной системе в случае успешного завершения?

- 1. 0

2. -1
3. Программа не возвращает значение.
4. 1

27. Название C++ предложил

1. Рик Масситти
2. Кэн Томпсон
3. Бьерн Страуструп
4. Дональд Кнут

28. До каких пор будут выполняться операторы в теле цикла while ($x < 100$)?

1. Пока x строго меньше ста
2. Пока x меньше или равен ста
3. Пока x равен ста
4. Пока x больше ста

29. Укажите правильное определение функции main в соответствии со спецификацией стандарта ANSI

1. `int main(void)`
2. `int main()`
3. `void main()`
4. `void main(void)`

30. Какие преобразования типов данных не возможны без потери данных?

1. `char to float`
2. все перечисленные преобразования не возможны
3. `float to int`
4. `int to float`

Раздел 3. МДК.01.03 Программирование на языках высокого уровня

1. Что не является обязательным элементом объектной модели:

- a) абстрагирование;
- b) инкапсуляция;
- c) устойчивость;
- d) модульность;

2. время исполнения проекта – это:

- a) это время, необходимое для создания рабочей версии программы, т.е. версии, готовой для работы в производственных условиях и выполняющей предусмотренные функции;
- b) это время, необходимое для создания базы данных, готовой для работы в производственных условиях и выполняющей предусмотренные функции;
- c) это время, необходимое для создания рабочей версии программы, т.е. версии, готовой для работы в производственных условиях, но не выполняющей предусмотренные функции;
- d) нет правильного варианта;

3. технология программирования не включает:

- a) указание последовательности выполнения технологических операций;
- b) перечисление условий, при которых выполняется та или иная операция;
- c) описания самих операций, где для каждой операции определены исходные данные, результаты, а также инструкции, нормативы, стандарты, критерии и методы оценки и т. п.;

d) время выполнения операций;

4. какой из элементов не является элементом объектной модели:

- a) модульность;
- b) параллелизм;
- c) последовательность;
- d) иерархия;

5. Программа-это:

- a) последовательность действий, направленных на выполнение их некоторым исполнителем.;
- b) это точно определенная последовательность простых действий, обеспечивающих решение любой задачи из некоторого класса;
- c) комплекс программ, которые обеспечивают управление компонентами компьютерной системы;
- d) указание последовательности выполнения технологических операций;

6. Что не включает в себя качество ПО:

- a) эффективность;
- b) мобильность;
- c) практичность;
- d) управление доступом;

7. Массив-это:

- a) упорядоченная последовательность переменных разных типов;
- b) упорядоченная последовательность констант одного типа;
- c) упорядоченная последовательность переменных одного типа;
- d) нет верного ответа;

8. К структурированным типам относятся

- 1) вещественный, 2) файл, 3) множество,
- 4) символьный, 5) диапазон, 6) целый

1. Идентификатор - это:

- a) последовательность латинских букв, цифр и символа «_», начинающаяся с буквы или символа «_»
- b) неизменяемые объекты языка (константы)
- c) последовательность латинских и русских букв
- d) способ кодирования, допустимые преобразования над значением данной переменной

2. Оператор INT в Си++ применяется для: 1: переопределения диапазона целых чисел 2: преобразования переменной к целому типу 3: описания переменных целого типа 4: прибавления единицы к коду символа

1. Проект – это:

- a) набор программ и установленные отношения между ними;
- b) набор компонентов и установленные отношения между ними;
- c) набор классов и установленные отношения между ними;
- d) нет верного ответа;

2. Задача компилятора -:

- a) трансляция исходного кода программы в бинарное представление;
- b) чтение исходного кода программы;
- c) создание копии программы;

d) нет верного ответа;

1. Укажите неправильно записанную операцию отношения:

- a) !=;
- b) <=;
- c) >=;
- d) все операторы записаны правильно.

2. Структуры - это:

- a) это составной тип данных, построенный с использованием разнообразных типов
- b) упорядоченная последовательность констант одного типа;
- c) набор программ и установленные отношения между ними;
- d) нет верного ответа;

3. Перебор элементов массива характеризуется:

- a) количеством одновременно обрабатываемых элементов;
- b) количеством индексов;
- c) удалением массива;
- d) нет верного ответа;

4. Операции, применяемые к одному операнду, называются:

- a) бинарными;
- b) комплексные;
- c) унарные;
- d) нет верного ответа

5. Операция, возвращающая остаток от деления:

- a) ||;
- b) &;
- c) //;
- d) %;

6. Логическое «или»:

- a) &&;
- b) ||;
- c) !;
- d) <<;

7. Логическое «и»:

- a) >>;
- b) !;
- c) &&;
- d) ++;

8. Операции сравнения в результате дают значение типа: a) int;

- b) bool;
- c) real;
- d) double.

9. Цикл с постусловием?

- 1. for
- 2. do while
- 3. while

10. Цикл с предусловием?

1. do while
2. for
3. while

11. Программа, переводящая входную программу на исходном языке в эквивалентную ей выходную программу на результирующем языке, называется:

1. компилятор
2. транслятор
3. сканер
4. интерпретатор

12. Простые типы данных в C++.

1. целые – int, вещественные – float или real, символьные – char
2. целые – bool, вещественные – float или double, символьные – string
3. целые – int, вещественные – float или double, символьные – string
4. целые – int, вещественные – float или double, символьные – char

13. Структура объявления переменных в C++

1. [=], <идент. 2>, ...;
2. [=]; <идент. 2>, ...;
3. [:=], <идент. 2>, ...;
4. [==]; <идент. 2>, ...;

14. Какая из следующих записей - правильный комментарий в C++?

1. /* комментарий */
2. {комментарий}
3. */ Комментарии */
4. ** Комментарий **

15. Какой оператор не допускает перехода от одного константного выражения к другому?

1. точка с запятой
2. Stop;
3. end;
4. break;

16. Какие служебные символы используются для обозначения начала и конца блока кода?

1. begin end
2. { }
3. < >
4. ()

17. Чтобы подключить заголовочный файл в программу на C++, например iostream необходимо написать:

1. include #iostream,h;
2. #include <>; с iostream.h внутри скобок
3. #include <> с iostream внутри скобок
4. include (iostreamh)

18. Тело любого цикла выполняется до тех пор, пока его условие ...

1. ложно
2. истинно
3. у цикла нет условия

20. Оператор if else позволяет определить действие ...

1. только для ложного условия
2. для истинного и ложного условий
3. только для истинного условия

21. В какой из следующих строк выполняется обращение к седьмому элементу массива, размер массива равен 10?

1. mas;
2. mas[7];
3. mas(7);
4. mas[6];

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	Уровень сформированности компетенций
Более 80	«Отлично»	Содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	Высокий уровень
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	Продвинутый уровень

50-65	«Удовлетворительно»	Содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки	<i>Пороговый уровень</i>
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	<i>Компетенции не сформированы</i>

3. Задания в тестовой форме по дисциплине

Примеры заданий:

Тест по дисциплине: МДК 01.03 Разработка мобильных приложений 1) Набор средств программирования, который содержит инструменты, необходимые для создания, компиляции и сборки мобильного приложения называется: а) Android SDK б) JDK в) плагин ADT г) Android NDK 2) С какой целью был создан Open Handset Alliance? А) писать историю развития ОС Android б) продавать смартфоны под управлением Android в) рекламировать смартфоны под управлением Android г) разрабатывать открытые стандарты для мобильных устройств 3) С какой целью инструмент Intel* Graphics Performance Analyzers (Intel* GPA) System Analyzer используется в среде разработки Intel* Beacon Mountain? а) позволить разработчикам оптимизировать загрузженность системы при использовании процедур OpenGL б) для ускорения работы эмулятора в среде разработки в) для оптимизированной обработки данных и изображений г) позволить разработчикам эффективно распараллелить C++ мобильные приложения 3) Библиотеки, реализованные на базе PacketVideo OpenCORE: А) Media Framework Б) SQLite В) FreeType Г) 3D библиотеки 4) Какой движок баз данных используется в ОС Android? А) InnoDB Б) DBM В) MyISAM Г) SQLite 5) С какой целью инструмент Intel* Integrated Performance Primitives (Intel* IPP) используется в среде разработки Intel* Beacon Mountain? А) для оптимизированной обработки данных и изображений Б) позволить разработчикам оптимизировать загрузженность системы при использовании процедур OpenGL В) для ускорения работы эмулятора в среде разработки Г) позволить разработчикам эффективно распараллелить C++ мобильные приложения 6) Intel XDK поддерживает разработку под: А) JavaFX Mobile Б) Apple iOS, BlackBerry OS В) MtkOS, Symbian OS, Microsoft Windows 8 Г) Android, Apple iOS, Microsoft Windows 8, Tizen 7) Каждый приемник широковещательных сообщений является наследником класса ... А) ViewReceiver Б) IntentReceiver В) ContentProvider Г) BroadcastReceiver 8) Какой класс является основным строительным блоком для компонентов пользовательского интерфейса (UI), определяет прямоугольную область экрана и отвечает за прорисовку и обработку событий? А) GUI Б) View В) UIComponent Г) Widget 9) Какой слушатель используется для отслеживания события касания экрана устройства? А) OnPressListener Б) OnTouchListener В) OnClickListener Г) OnInputListener 10) В какой папке необходимо разместить XML файлы, которые определяют все меню приложения? А) res/value Б) res/items В) res/layout Г) res/menu 11) Фоновые приложения ... А) после настройки не предполагают взаимодействия с пользователем, большую часть времени находятся и работают в скрытом состоянии Б) выполняют свои функции и когда видимы на экране, и когда скрыты другими приложениями В) небольшие приложения, отображаемые в виде графического объекта на рабочем столе Г) большую часть времени работают в фоновом режиме, однако допускают взаимодействие с пользователем и после настройки 12) Полный иерархический список обязательных файлов и

папок проекта можно увидеть на вкладке ... А) Package Explorer Б) Internet Explorer В) Navigator Г) Project Explorer 13) Какой компонент управляет распределенным множеством данных приложения? А) сервис (Service) Б) активность (Activity) В) приемник широковещательных сообщений (Broadcast Receiver) Г) контент-провайдер (Content Provider) 14) Какой язык разметки используется для описания иерархии компонентов графического пользовательского интерфейса Android-приложения? А) html Б) xml В) gml Г) xhtml 15) Выберите верную последовательность действий, необходимых для создания в приложении контент-провайдера. А) Создание класса наследника от класса ContentProvider; Определение способа организации данных; Заполнение контент-провайдера данными Б) Проектирование способа хранения данных; Определение способа организации данных; В) Создание класса наследника от класса ContentProvider; Заполнение контент-провайдера данными; Определение способа работы с данными Г) Проектирование способа хранения данных; Создание класса-наследника от класса ContentProvider; Определение строки авторизации провайдера, URI для его строк и имен столбцов 16) Выберите верные утверждения относительно объекта-намерения (Intent). А) представляет собой структуру данных, содержащую описание операции, которая должна быть выполнена, и обычно используется для запуска активности или сервиса Б) используется для передачи сообщений пользователю В) используется для получения инструкций от пользователя Г) используются для передачи сообщений между основными компонентами приложений 17) Расположение элементов мобильного приложения: А) полезно для передачи иерархии Б) влияет на удобство использования В) полезно для создания пространственных отношений между объектами на экране и объектами реального мира Г) все варианты ответа верны 18) Какие элементы управления применяются для действий по настройке? А) командные элементы управления Б) элементы выбора В) элементы ввода Г) элементы отображения 19) Примерами комбо-элементов не являются: А) комбо-список Б) все вышеперечисленное В) комбо-кнопка Г) комбо-поле 20) Дизайн или проектирование интерфейса для графических дизайнеров: А) все варианты ответа верны Б) прозрачность и понятность информации В) тон, стиль, композиция, которые являются атрибутами бренда Г) передача информации о поведении посредством ожидаемого назначения 21) Более крупные элементы: А) привлекают больше внимания Б) все варианты ответа верны В) размер не влияет на уровень внимания Г) привлекают меньше внимания 22) К традиционным типографическим инструментам не относят А) масштаб Б) цвет В) разреженность Г) выравнивание по сетке 23) К элементам ввода относят: А) ограничивающие элементы ввода Б) ползунки В) счетчики Г) все вышеперечисленное 24) Выделяют следующие категории плотности экрана для Android-устройств: А) HDPI, XHDPI, XXHDPI, и XXXHDPI Б) правильный вариант ответа отсутствует В) LDPI, MDPI, HDPI, XHDPI, XXHDPI, и XXXHDPI Г) LDPI, MDPI, HDPI 25) Следующие утверждения не верны: А) не используйте интерфейсные элементы Б) картинки работают быстрее, чем слова В) на любом шаге должна быть возможность вернуться назад Г) если объекты похожи, они должны выполнять сходные действия 26) Следующие утверждения верны: А) текстура бесполезна для передачи различий или привлечения внимания Б) восприятие направления затруднено при больших размерах объектов В) все варианты ответа верны Г) люди легко воспринимают контрастность 27) Основные вкладки (FixedTabs) удобны при отображении А) от четырех вкладок Б) двух вкладок В) трех и более вкладок Г) трех и менее вкладок 28) Диалоговое окно, содержащее линейку процесса выполнения какого-то действия — это А) DatePickerDialog Б) AlertDialog В) ProgressDialog Г) DialogFragment 29) Уведомления стоит использовать, когда А) сообщение не требует ответа пользователя, но важно для продолжения его работы Б) сообщение является важным и требует немедленного прочтения и ответа В) сообщение является важным, однако требует немедленного прочтения, но не ответа Г) сообщение является важным, однако не требует немедленного прочтения и ответа 30) Какой метод запускает новую активность? А) startActivity() Б) beginActivity() В) intentActivity() Г) newActivity() 31) ProgressDialog это: А) контейнер для создания собственных диалоговых окон Б) диалоговое окно с предопределенным интерфейсом, позволяющее выбрать дату или время В) диалоговое окно, содержащее линейку процесса выполнения какого-то действия Г) диалоговое окно, которое может содержать заголовок, до трех кнопок,

список выбираемых значений или настраиваемое содержимое 32) AlertDialog это: А) контейнер для создания собственных диалоговых окон Б) диалоговое окно, содержащее линейку процесса выполнения какого-то действия В) диалоговое окно, которое может содержать заголовок, до трех кнопок, список выбираемых значений или настраиваемое содержимое Г) диалоговое окно с предопределенным интерфейсом, позволяющее выбрать дату или время 33) Что необходимо сделать при добавлении в проект новой активности? А) скачать и установить специальный инструмент MultiActivity SDK Б) прописать в манифесте информацию о новой активности В) создать новый проект Г) запустить эмулятор 34) Системы позиционирования смартфона могут включать А) все перечисленное Б) систему GPS В) систему ГЛОНАСС Г) сигналы WiFi и Bluetooth 35) Какая константа не определена в классе MotionEvent, для обозначения сенсорных событий А) ACTION_DOWN Б) ACTION_UP В) ACTION_CLICK Г) ACTION_MOVE 36) С какой целью используется метод release() в классах MediaPlayer и MediaRecorder? А) конец жизненного цикла объекта и освобождение ресурсов Б) перевод объекта в ожидающее состояние В) обновление объекта и запуск его работы Г) создание объекта и запуск его работы 37) К датчикам окружающей среды, встроенным в мобильное устройство относят А) датчики вектора вращения Б) датчики освещенности В) акселерометры Г) гироскопы 38) Библиотека Universal Image Loader for Android позволяет: А) парсить HTML-страницы Б) строить графики и диаграммы В) загружать, кешировать и отображать изображения Г) использовать анимацию, доступную только с версии 3.x, на более ранних вариантах платформы Android 39) Facebook SDK for Android — это библиотека, позволяющая: А) получать доступ к информации любого пользователя Б) отправлять рекламные сообщения от имени пользователя В) писать сообщения на стену, читать и менять статусы, смотреть ленту друзей Г) парсить страницы пользователей 40) Что из перечисленного не относится к правилам безопасности при подключении библиотек? А) с осторожностью использовать библиотеки из сомнительных источников Б) ознакомиться с форумами и сайтами, где могут обсуждаться библиотеки В) лично познакомиться с разработчиками библиотеки Г) использовать скомпрометированные библиотеки 41) Библиотека MapNavigator предназначена для: А) работы с любыми картами Б) работы с Яндекс.Картами В) морской навигации Г) работы с картами Google Maps 42) Библиотека jsoup не позволяет: А) находить и извлекать данные, используя DOM и селекторы CSS Б) манипулировать HTML-элементами, атрибутами и текстом В) писать сообщения на стену, читать и менять статусы, смотреть ленту друзей Г) принимать в качестве параметра URL, файл или строку 43) При настройке обратной совместимости необходимо добавить в файл манифеста следующую информацию: А) только минимальную версию Android SDK Б) минимальную и основную (целевую) версии Android SDK В) информацию о подключенной библиотеке Г) только основную (целевую) версии Android SDK 44) Какая библиотека предназначена для упрощения загрузки изображений? А) Yandex.Metrica for Apps; Б) Universal Image Loader for Android В) ActionBarSherlock Г) NineOldAndroids 45) Библиотеки совместимости предназначены для А) сбора статистики Б) рисования графиков В) использования возможностей, появившиеся в какой-то версии ОС Android, на более ранних версиях платформы Г) подключения нестандартных элементов управления 46) Какая библиотека предназначена для использования анимации? А) Universal Image Loader for Android Б) NineOldAndroids В) Yandex.Metrica for Apps Г) ActionBarSherlock 47) Для чего служит папка res/anim/ проекта? А) в этой папке находятся файлы, содержащие набор картинок, предназначенных для кадровой анимации Б) в этой папке находятся файлы, содержащие анимированные ролики для воспроизведения в приложении В) в этой папке находятся XML файлы, задающие реализацию анимации свойств Г) в этой папке находятся XML файлы, задающие последовательность инструкций анимации преобразований 48) В какой файл обязательно добавляется информация при создании нового Activity в приложении? А) AndroidManifest.xml Б) main.java В) layout.xml Г) activity.xml 49) Какой метод жизненного цикла активности вызывается системой непосредственно перед появлением активности на экране? А) onVisible() Б) onOpen() В) onResume() Г) onCreate() 50) С какой целью используется метод SurfaceHolder.lockCanvas()? А) блокировка Canvas для перерисовки Б) игнорирование дальнейшего взаимодействия с Canvas В) сокрытие Canvas Г)

блокировка Canvas от сворачивания 51) Может ли мобильное приложение получить доступ к базе данных, созданной в другом приложении? А) не может ни при каких обстоятельствах Б) может, но только с помощью контент-провайдеров В) право на доступ открывает приложение-хозяин базы данных Г) может обращаться напрямую 52) С помощью какого метода можно запретить смену ориентации устройства, при запущенном приложении? А) setRequestedOrientation Б) setChangeOrientation В) disableChangeOrientation Г) setOrientation 53) Какой из датчиков не используется для определения положения смартфона в пространстве? А) акселерометр Б) gps В) гироскоп Г) магнитометр 54) К новым возможностям HTML5 относят (выберите все верные варианты ответа): А) возможность добавления аудио и видео без использования вспомогательных средств Б) возможность рисования на холсте В) возможность прямого доступа к оперативной памяти Г) форматирование данных в режиме таблицы 55) Возможен ли перенос приложений iOS* в среду HTML5: А) нет, прямой перенос приложений невозможен Б) да, используя средства Intel XDK В) да, используя только сторонние средства Г) да, только для iPhone, используя средства Intel XDK 56) Следующие утверждения верны: А) JavaScript не позволяет подключать другие внешние библиотеки, написанные на других языках Б) приложения html5 исполняются быстрее и требуют меньше ресурсов, чем «нативные» В) среда Intel XDK не работает с мультисенсорностью Г) приложения html5 исполняются медленнее и требуют больших ресурсов, чем «нативные» 57) Разрабатывать приложения в среде Intel XDK можно: А) пользоваться готовыми примерами Б) все варианты ответа верны В) «с нуля», прописывая все элементы Г) использовать встроенный «дизайнер элементов» для отрисовки элементов 58) JavaScript не позволяет: А) получать прямой доступ к памяти Б) работать с реестром В) работать с картами Г) одновременно использовать несколько подключаемых библиотек 59) В среде Intel XDK можно разрабатывать приложения для следующих платформ: А) Android Б) все варианты ответа верны В) Apple iOS Г) Tizen 60) В заготовке любого приложения, разрабатываемого в среде Intel XDK прописана: А) все варианты ответа верны Б) скрытие окна заставки Intel XDK В) настройка размеров приложения под размеры устройства Г) фиксация размеров приложения (запрет «скроллинга») 61) Создавать и редактировать пользовательский интерфейс приложений в среде Intel XDK можно: А) используя встроенное приложение App Designer Б) только изменяя готовые шаблоны с интерфейсом В) все варианты ответа неверны Г) только прописывая теги вручную 62) Удобное средство обмена между двумя NFC-устройствами: А) Wi-Fi Direct Б) AndroidBeam В) Dalvik Г) Bluetooth 63) Переключения между активностями осуществляются А) только при помощи кнопок Б) только с использованием сенсорного экрана смартфона В) только при помощи кнопок и других элементов управления Г) все три варианта возможны

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке <https://www.mivlgu.ru/iop/course/view.php?id=3715>

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.