

Подписано электронной подписью:
Вержицкий Данил Григорьевич
Должность: Директор КГПИ ФГБОУ ВО «КемГУ»
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Дата и время: 2024-02-21 00:00:00
Кузбасский государственный технический университет
федерального государственного бюджетного образовательного учреждения
высшего образования
«Кемеровский государственный университет»
Факультет информатики, математики и экономики

УТВЕРЖДАЮ

Декан А.В. Фомина
«09» февраля 2023 г.

Рабочая программа дисциплины
К.М.05.10 Проектирование и разработка мобильных приложений

Направление подготовки

09.03.01 Информатика и вычислительная техника

Направленность (профиль) подготовки
«Автоматизированные системы обработки информации и управления»
Программа бакалавриата

Квалификация выпускника
бакалавр

Форма обучения
Очная

Год набора 2022

Новокузнецк 2023

Оглавление

1 Цель дисциплины.....	3
1.1 Формируемые компетенции	3
1.2 Индикаторы достижения компетенций.....	3
1.3 Знания, умения, навыки (ЗУВ) по дисциплине.....	4
2 Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации.	6
3. Учебно-тематический план и содержание дисциплины.....	6
3.1 Учебно-тематический план.....	6
3.2. Содержание занятий по видам учебной работы	7
4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации.	8
5 Учебно-методическое обеспечение дисциплины.	10
5.1 Учебная литература	10
5.2 Программное и информационное обеспечение освоения дисциплины.	10
5.2.1 Программное обеспечение.....	10
5.2.2 Современные профессиональные базы данных и информационные справочные системы.	11
6 Иные сведения и (или) материалы.	11
6.1.Примерные темы и варианты письменных учебных работ	11
6.2. Примерные вопросы и задания / задачи для промежуточной аттестации	12
Сведения о разработке и утверждении рабочей программы дисциплины	17

1 Цель дисциплины.

В результате освоения данной дисциплины у обучающегося должны быть сформированы компетенции основной профессиональной образовательной программы бакалавриата (далее - ОПОП): ОПК–2, ОПК-9.

Содержание компетенций как планируемых результатов обучения по дисциплине см. таблицы 1.1, 1.2 и 1.3.

1.1 Формируемые компетенции

Таблица 1.1 - Формируемые дисциплиной компетенции

Наименование вида компетенции	Наименование категории (группы) компетенций	Код и название компетенции
Общепрофессиональная		ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности
Общепрофессиональная		ОПК-9 Способен осваивать методики использования программных средств для решения практических задач.

1.2 Индикаторы достижения компетенций

Таблица 1.2 – Индикаторы достижения компетенций, формируемые дисциплиной

Код и название компетенции	Индикаторы достижения компетенции по ОПОП	Дисциплины и практики, формирующие компетенцию ОПОП
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.1. Выбирает современные информационные технологии, в том числе отечественного производства, и программные средства для решения поставленной задачи профессиональной деятельности. ОПК-2.2. Применяет технологии параллельного программирования в соответствии с заданием. ОПК-2.3. Применяет технологии web-программирования в соответствии с заданием. ОПК-2.4. Разрабатывает инфологические и даталогические схемы баз данных в соответствии с заданием. ОПК-2.5. Работает с	К.М.05 Современные информационные технологии и информационные системы К.М.05.09 Базы данных К.М.05.10 Проектирование и разработка web-приложений К.М.05.11 Проектирование и разработка мобильных приложений К.М.05.12 Параллельные и распределенные вычислительные системы К.М.05.13 Системы искусственного интеллекта К.М.06 Обеспечение проектной деятельности К.М.06.02 Технологии программирования К.М.06.03 Автоматизация процесса разработки проектной документации К.М.08 Практики К.М.08.01(П) Технологическая (проектно-технологическая) практика К.М.09 Государственная итоговая аттестация

Код и название компетенции	Индикаторы достижения компетенции по ОПОП	Дисциплины и практики, формирующие компетенцию ОПОП
	современными системами управления базами данных. ОПК-2.6. Применяет в соответствии с заданием современные методы, средства и технологии на различных фазах создания и эксплуатации систем искусственного интеллекта	К.М.09.01(Д) Выполнение и защита выпускной квалификационной работы
ОПК-9 Способен осваивать методики использования программных средств для решения практических задач.	ОПК-9.1. Определяет категорию программных продуктов, с помощью которой может быть решена поставленная задача. ОПК-9.2. Применяет справочные средства для освоения программных продуктов, необходимых для решения поставленной задачи. ОПК-9.3. Осуществляет поиск информации, необходимой для самостоятельного освоения типовых и специализированных программных средств в соответствии с заданием. ОПК-9.4. Самостоятельно осваивает и применяет типовые и специализированные программные	К.М.05 Современные информационные технологии и информационные системы К.М.05.07 Пакеты прикладных программ компьютерного моделирования К.М.05.10 Проектирование и разработка web-приложений К.М.05.11 Проектирование и разработка мобильных приложений К.М.05.12 Параллельные и распределенные вычислительные системы К.М.08 Практики К.М.08.02(П) Эксплуатационная практика К.М.09 Государственная итоговая аттестация К.М.09.01(Д) Выполнение и защита выпускной квалификационной работы

1.3 Знания, умения, навыки (ЗУВ) по дисциплине

Таблица 1.3 – Знания, умения, навыки, формируемые дисциплиной

Код и название компетенции	Индикаторы достижения компетенции, закрепленные за дисциплиной	Знания, умения, навыки (ЗУВ), формируемые дисциплиной
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и	ПК 2.3 Разрабатывает код для реализации компонент информационных систем	Знать: - теоретические основы функционирования Web-сети; - основные стандарты Web-сети (HTTP, HTML, CSS, Javascript); - понятие web-приложений и web-сервисов; - основные подходы к разработке web-приложений; - технологию разработки web-приложений Microsoft ASP.Net Web Forms; - способы проектирования web-приложений.

Код и название компетенции	Индикаторы достижения компетенции, закрепленные за дисциплиной	Знания, умения, навыки (ЗУВ), формируемые дисциплиной
использовать их при решении задач профессиональной деятельности		<p>Уметь:</p> <ul style="list-style-type: none"> - разрабатывать Web-приложения с использованием технологии разработки Web-приложений Microsoft ASP.Net Web Forms; - разработка Web-сервисов с использованием технологии разработки Web-приложений Microsoft ASP.Net; - проектировать web-приложения. <p>Владеть:</p> <ul style="list-style-type: none"> - интегрированной средой разработки программных систем с помощью среды разработки Microsoft Visual Studio; - навыками разработки web-приложений.
ОПК-9 Способен осваивать методики использования программных средств для решения практических задач.	ОПК-9.4. Самостоятельно осваивает и применяет типовые и специализированные программные	<p>Уметь:</p> <p>самостоятельно осваивать и применять средства эффективной разработки Web-приложений.</p>

2 Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации.

Таблица 2 – Объём и трудоёмкость дисциплины по видам учебных занятий

Общая трудоёмкость и виды учебной работы по дисциплине, проводимые в разных формах	Объём часов по формам обучения		
	ОФО	ОЗФО	ЗФО
1 Общая трудоёмкость дисциплины	144		
2 Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)	36		
Аудиторная работа (всего):	36		
в том числе:			
лекции	18		
практические занятия, семинары			
практикумы			
лабораторные работы	18		
в интерактивной форме			
в электронной форме			
Внеаудиторная работа (всего):			
в том числе, индивидуальная работа обучающихся с преподавателем			
подготовка курсовой работы /контактная работа ¹			
групповая, индивидуальная консультация и иные виды учебной деятельности, предусматривающие групповую или индивидуальную работу обучающихся с преподавателем)			
творческая работа (эссе)			
3 Самостоятельная работа обучающихся (всего)	108		
4 Промежуточная аттестация обучающегося – зачет с оценкой 7 семестр			

3. Учебно-тематический план и содержание дисциплины. 3.1 Учебно-тематический план

Таблица 3.1 - Учебно-тематический план

№ п/п	Разделы и темы дисциплины по занятиям	Общая трудоёмкость (всего час.)	Трудоёмкость занятий (час.)				Формы текущего контроля и промежуточной аттестации успеваемости
			ОФО			СРС	
			Аудиторн. занятия				
			лекц.	практ.	лаб.		
1-2	Классификация мобильных устройств. Архитектура мобильных устройств и их компонентов.	36	2		2	32	Устный опрос, решение учебных задач
3-8	Java для мобильных устройств.	36	6		6	24	Устный опрос, решение учебных задач
9-16	Программирование для ОС Android.	36	8		8	20	Устный опрос, решение учебных задач
17-18	Особенности экосистемы iOS. Особенности разработки приложений под WindowsPhone.	36	2		2	32	Устный опрос, решение учебных задач

¹ Часы, выделенные в УП на курсовое проектирование в контактной форме (3 часа)

№ п/п	Разделы и темы дисциплины по занятиям	Общая трудоёмкость (всего час.)	Трудоемкость занятий (час.)				Формы текущего контроля и промежуточной аттестации успеваемости
			ОФО			СРС	
			Аудиторн. занятия				
			лекц.	практ.	лаб.		
	Промежуточная аттестация- зачет с оценкой						
	Всего:	144	18		18	108	

3.2. Содержание занятий по видам учебной работы

Таблица 3.2 – Содержание дисциплины

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
<i>Содержание лекционного курса</i>		
1.1	Классификация мобильных устройств. Архитектура мобильных устройств и их компонентов.	Особенности и назначение мобильных устройств. История появления мобильных устройств, архитектура мобильных устройств, операционные системы для мобильных устройств (обзор), современные мобильные устройства на примере устройств для ОС iOS, Android и WindowsMobile. Операционные системы для мобильных устройств. Возможности современных ОС для мобильных устройств, API ОС мобильных устройств, Java для мобильных устройств и библиотеки ОС iOS и Android. Программирование мобильных устройств (обзор). Java ME: архитектура и возможности, применимость, недостатки и преимущества; программирование на C++ и Objective-C для OS iOS, программирование на Java и C++ для ОС Android, сторонние средства разработки мобильных приложений. Создание Android-приложений на языке C++. Недостатки и преимущества использования Java и C++ в мобильных устройствах; компиляция C++ приложений в Android: Android NDK; типовые задачи для реализации на C++; отладка C++ приложений.
1.2	Java для мобильных устройств.	Конфигурации и профили в Java ME. Среды разработки, версии Java ME, профили MIDL 1 и 2, их отличительные особенности, требования к аппаратной среде. Мидлеты. Жизненный цикл мидлета. События высокого и низкого уровня; компоненты пользовательского интерфейса. Сборка и запуск сложных мидлетов. Компиляция, верификация, загрузка мидлета, отладка; сетевое взаимодействие, протоколы; модель клиент-сервер, тонкие клиенты; взаимодействие мидлетов с серверным ПО через сеть.
1.3	Программирование для ОС Android.	Архитектура программ в ОС Android. Виртуальная машина Java в Android; создание приложений под ОС Android; Android SDK и сторонние разработки; установка инструментария, компиляция и установка Android-приложений. Пользовательский интерфейс и обработка событий в ОС Android. Принципы работы с Android: Activity (Активность, Деятельность), Intents (Намерения), Views (Представление), Services (Службы), ContentProvider (Контент-провайдер), BroadcastReceiver (Приемник широковещательных сообщений/запросов); элементы управления и работа с ними, обработка событий, модель документ/представление в мобильном программировании, работа с API ОС Android. Текстовые элементы управления, кнопки, списки, таблицы, управление датой и временем, MapView, галерея, счетчик, диспетчеры шаблонов, адаптеры, создание меню, расширенные меню, загрузка меню при помощи XML-файлов, создание диалоговых окон, диалоговые окна с подсказками и предупреждениями. Доступ к оборудованию из Android-приложения. Принципы работы с оборудованием: типовой набор оборудования мобильного устройства Android; принципы работы с камерой; распознавание пользовательских жестов; работы с модулями определения геоположения; работа с GSM сервисами; акселерометры и гироскопы. Введение в разработку Android-приложений. Инструменты

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
		разработки Android-приложений. Пример простейших программ Android-приложения. Запуск приложения на эмуляторе. Управление ресурсами. Тестирование приложения. Среда разработки AndroidStudio, Eclipse. Работа с AndroidMarket. Подготовка AndroidManifest.xml для загрузки, локализация приложения, подготовка ярлыка приложения, подготовка APK-файла для загрузки, работа пользователя с AndroidMarket.
1.4	Особенности экосистемы iOS. Особенности разработки приложений под WindowsPhone.	Экосистема Apple. Концепция закрытой экосистемы; принципы AppleiTunesStore; статус AppleDeveloper; требования Apple к программам; процесс апробации и одобрения приложений; платные приложения и встроенные покупки. Принципы программирования для iOS. Разные версии iOS и их возможности; концепция пользовательского интерфейса; программирование на Objective-C: основные особенности, обработка событий, обращения к методам, обработка событий. Среда разработки приложений для iOS. Среда разработки XCode, provision профили устройств, сборка приложений, отладка и тестирование приложений, отправка приложений на апробацию и одобрение; сторонний инструментальный для разработки. Принципы программирования для WindowsPhone 7. Мобильное программирование, платформы для разработки. Система Windows Phone 7. Microsoft Visual Studio Express for Windows Phone. Аппаратные средства устройств, поддерживающих WindowsPhone 7. История появления, аспекты применения MicrosoftSilverlight. Основы работы с сенсорным вводом. Обработка нескольких касаний. Использование изображений. Возможности сбора сведений об окружающем мире в WindowsPhone 7. Акселерометр и служба определения местоположения, вторичные потоки выполнения, обработка асинхронных операций и доступ к Веб-сервисам. Особенности использования pivot и panorama. Принципы интерфейса системы и приложений Metro. Типографика.
<i>Содержание практических занятий</i>		
1.1	Классификация мобильных устройств. Архитектура мобильных устройств и их компонентов.	Программирование на C++ для ОС Android. Типовые задачи для реализации на C++. Отладка C++ приложений.
1.2	Java для мобильных устройств.	Программирование на Java ОС Android. Компиляция, верификация, загрузка мидллета, отладка.
1.3	Программирование для ОС Android.	Запуск приложения на эмуляторе. Управление ресурсами. Тестирование приложения. Среда разработки Eclipse. Создание приложений под ОС Android. Установка инструментария, компиляция и установка Android-приложений. Пользовательский интерфейс и обработка событий в ОС Android. Подготовка AndroidManifest.xml для загрузки, локализация приложения, подготовка ярлыка приложения, подготовка APK-файла для загрузки, работа пользователя с AndroidMarket.
1.4	Особенности экосистемы iOS. Особенности разработки приложений под WindowsPhone.	Принципы программирования для iOS. Принципы программирования для WindowsPhone 7.

4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации.

Для положительной оценки по результатам освоения дисциплины обучающемуся необходимо выполнить все установленные виды учебной работы. Оценка результатов работы обучающегося в баллах (по видам) приведена в таблице 4.

Таблица 4.1 - Балльно-рейтинговая оценка результатов учебной работы обучающихся по видам (БРС)

Учебная работа (виды)	Сумма баллов	Виды и результаты учебной работы	Оценка в аттестации	Баллы (18 недель)
-----------------------	--------------	----------------------------------	---------------------	-------------------

Текущая учебная работа ОФО (2 семестр)				
Текущая учебная работа в семестре (посещение занятий по расписанию и выполнение заданий)	60 (100% /баллов приведенной шкалы)	Лекционные занятия (9 занятий)	1 балл – посещение 1 лекционного занятия	7 - 9
		Практические занятия (36 занятий)	0.75 балла – посещение 1 занятия и выполнение задания на 51-85% 1 балл – посещение 1 занятия и существенный вклад на занятии в работу всей группы, самостоятельность и выполнение работы на 85.1-100%	33 - 36
		Написание реферата по выбранной теме	11 баллов – реферат написан на «удовлетворительно» 13 баллов – реферат написан на «хорошо» 15 баллов – реферат написан на «отлично»	11 - 15
Итого по текущей работе в семестре				51-60
Промежуточная аттестация				
Промежуточная аттестация (экзамен)	40 (100% /баллов приведенной шкалы)	Вопрос 1.	10 баллов (пороговое значение) 20 баллов (максимальное значение)	10 - 20
		Решение задачи 1.	10 баллов (пороговое значение) 20 баллов (максимальное значение)	10 – 20
Итого по промежуточной аттестации (экзамен)				20-40
Суммарная оценка по дисциплине: Сумма баллов текущей и промежуточной аттестации 51 – 100 баллов.				

Итоговая оценка выставляется в ведомость согласно следующему правилу (таблица 4.2):

Таблица 4.2. Оценка уровня сформированности компетенций в промежуточной аттестации

Критерии оценивания компетенции	Уровень сформированности компетенции	Итоговая оценка	Оценка по 100-балльной шкале
Обучающийся не владеет теоретическими основами дисциплины и научной терминологией, демонстрирует отрывочные знания, не способен решать практические профессиональные задачи, допускает множественные существенные ошибки в ответах, не умеет интерпретировать результаты и делать выводы.	недопустимый	неудовлетворительно	Менее 51 балла
Обучающийся владеет частично теоретическими основами дисциплины и научной терминологией, фрагментарно способен решать практические профессиональные задачи, допускает несколько существенных ошибок решениях, может частично интерпретировать полученные результаты, допускает ошибки в выводах.	пороговый	удовлетворительно	51-65
Обучающийся владеет теоретическими основами дисциплины и научной терминологией, грамотно излагает материал, способен решать практические профессиональные задачи, но допускает отдельные несущественные ошибки в интерпретации результатов и выводах.	повышенный	хорошо	66-85
Обучающийся в полной мере владеет	продвинутый	отлично	86-100

теоретическими основами дисциплины и научной терминологией, грамотно излагает материал, способен иллюстрировать ответ примерами, фактами, данными научных исследований, применять теоретические знания для решения практических профессиональных задач. Правильно интерпретирует полученные результаты и делает обоснованные выводы.			
--	--	--	--

5 Учебно-методическое обеспечение дисциплины.

5.1 Учебная литература

Основная учебная литература

1. Соколова, В. В. Разработка мобильных приложений : учебное пособие / В.В. Соколова. – Томск : Издательство Томского политехнического университета, 2015. – 176 с. – ISBN 978-5-4387-0369-3. – URL: <http://biblioclub.ru/index.php?page=book&id=442808>. – (дата обращения 31.08.2019). – Текст: электронный.
2. Сильвен, Р. Android NDK. Разработка приложений под Android на C/C++ / Р. Сильвен. – Москва : ДМК Пресс, 2012. – 496 с. – ISBN 978-5-94074-657-7. – URL: <https://e.lanbook.com/book/9126>. – (дата обращения 31.08.2019). – Текст: электронный.

Дополнительная литература

1. Куркин, А. В. Программирование под платформу Andriod : учебное пособие / А.В. Куркин. – Санкт-Петербург : НИУ ИТМО, 2015. – 35 с. – URL: <https://e.lanbook.com/book/91569>. – (дата обращения 31.08.2019). – Текст: электронный.
2. Кузин, А. В. Основы программирования на языке Objective-C для iOS : учеб. пособие / А.В. Кузин, Е.В. Чумакова. – Москва. : ИНФРА-М, 2019. – 118 с. – ISBN 978-5-16-005042-3– URL: <http://znanium.com/catalog/product/1003260>. – (дата обращения 31.08.2019). – Текст: электронный.
3. Пруцков, А. В. Программирование на языке Java. Введение в курс с примерами и практическими заданиями : учебник / А.В. Пруцков. – Москва : КУРС, 2018. – 208 с. – ISBN 978-5-906923-51-6. – URL: <http://znanium.com/catalog/product/1017180>. – (дата обращения 31.08.2019). – Текст: электронный.

5.2 Программное и информационное обеспечение освоения дисциплины.

5.2.1 Программное обеспечение

В обучении используются информационные технологии на базе компьютерных классов учебного корпуса №4 (пр. Metallurgov 19):

- лекционные занятия ведутся с использованием презентаций и программного обеспечения мультимедиа демонстраций;

Для проведение практических занятий, самостоятельной работы студентов используется программное обеспечение, приведенные в таблице 5.

Таблица 5 – Информационные технологии и программное обеспечение аудиторных занятий и самостоятельной работы

Программа / система	Сведения о праве использования (лицензия,	№ комп. классов
---------------------	---	-----------------

	договор, сроки использования).	
Программное обеспечение компьютеров: Операционные системы: Windows 7; Антивирусное ПО: Eset Endpoint Security 5.0.		
Лицензионное ПО		
Офисное ПО		
Microsoft Office, Visio	Лицензия DreamSpark Premium Electronic Software Delivery (3 years) Renewal по сублицензионному договору №Tr000083174 от 12.04.2016г.	501, 502, 508, 509, 602
Интерегрированные программы системы		
Microsoft Visual Studio 2010	Лицензия DreamSpark Premium Electronic Software Delivery (3 years) Renewal по сублицензионному договору №Tr000083174 от 12.04.2016г.	501, 502, 508, 509, 602
Свободнораспространяемое ПО по лицензиям GNU GPL, MIT, BSD License, Mozilla Public License		
Системное ПО		
7-zip	Бесплатно	501, 502, 508, 509, 602
Эмуляторы		
Oracle VM Virtual Box	Бесплатно	501, 502, 508, 509, 602
Инструментальное ПО		
Bloodshed Dev- C ++ 4.9.9.2	Бесплатно	501, 502, 508, 509, 602
Java	Бесплатно	501, 502, 508, 509, 602
Mpich 2	Бесплатно	501, 502, 508, 509, 602
Qt	Бесплатно	508
Foxit Reader	Бесплатно	501, 502, 508, 509, 602
Интегрированная среда разработки		
Eclipse	Бесплатно	501, 502, 508, 509, 602
Браузеры и дополнения		
IE 8	Бесплатно	501, 502, 508, 509, 602
Firefox 14	Бесплатно	501, 502, 508, 509, 602
Opera 12	Бесплатно	501, 502, 508, 509, 602
Netbeans IDE 7.0.1 для Firefox	Бесплатно	501, 502, 508, 509, 602

5.2.2 Современные профессиональные базы данных и информационные справочные системы.

1. Информационная система «Единое окно доступа к образовательным ресурсам. Раздел. Информатика и информационные технологии» - http://window.edu.ru/catalog/?p_rubr=2.2.75.6

2. Крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки- <https://github.com/>

3. База книг и публикаций Электронной библиотеки "Наука и Техника" - <http://www.n-t.ru>

6 Иные сведения и (или) материалы.

6.1.Примерные темы и варианты письменных учебных работ

Примерный перечень тем рефератов:

1. Сравнительный обзор мобильных операционных систем iOS, Windows mobile, Android.

2. Мобильное программирование, платформы для разработки.
3. Краткая история ОС Android.
4. Intel для Android: партнерство и инструментарий разработчика.
5. Архитектура приложений для Android.
6. Инструментарий разработки приложений для Android.
7. Обзор шагов разработки типичного приложения под Android.
8. Особенности разработки с использованием эмулятора.
9. Тестирование приложения с помощью Dalvik Debug Monitor Server.
10. Юзабилити пользовательского интерфейса.
11. Развертывание мобильного приложения в маркете.
12. Акселерометр и служба определения местоположения.
13. Приложения на Silverlight.

6.2. Примерные вопросы и задания / задачи для промежуточной аттестации

Таблица 6 - Примерные теоретические вопросы к экзамену

Разделы и темы	Примерные теоретические вопросы	Примерные практические задания / задачи
Классификация мобильных устройств. Архитектура мобильных устройств и их компонентов.	<ol style="list-style-type: none"> 1. История появления мобильных устройств и их архитектура. 2. Операционные системы для мобильных устройств. 3. Возможности современных ОС для мобильных устройств. 4. Мобильные устройства на примере устройств для ОС iOS, особенности. 5. Мобильные устройства на примере устройств для ОС Android, особенности. 6. Мобильные устройства на примере устройств для ОС WindowsMobile, особенности. 	Типовое практическое задание
Java для мобильных устройств.	<ol style="list-style-type: none"> 7. Java для мобильных устройств, архитектура и возможности. 8. Недостатки и преимущества Java при программировании для мобильных устройств. 9. Мидлеты. Определение и особенности. 10. Жизненный цикл мидлета. Загрузка и выполнение. 11. События Java, обработка событий. 12. Компоненты пользовательского интерфейса Java. 13. Взаимодействие с аппаратной средой из Java, работа с сетью. 14. Клиент-серверное взаимодействие мобильных приложений. 15. Виртуальная машина Java в Android, особенности. 	Типовое практическое задание
Программирование для ОС Android.	<ol style="list-style-type: none"> 16. Создание приложений под ОС Android: способы разработки приложений. 17. Android SDK и Android NDK. Назначение и особенности. 18. Принципы работы с ОС Android: Activity и Intents. Определения, пример. 19. Принципы работы с ОС Android: Views, Services. Назначение, пример. 20. Принципы работы с ОС Android: ContentProvider, BroadcastReceiver. Назначение. 21. Инструментарий элементов управления Android. 22. Модель обработки событий ОС Android. Пример 	Типовое практическое задание

	<p>обработчиков событий.</p> <p>23. Модель документ/представление в мобильном программировании.</p> <p>24. Доступ к оборудованию в ОС Android (общие принципы).</p> <p>25. Доступ к оборудованию в ОС Android: получение снимка видеокамерой.</p> <p>26. Доступ к оборудованию в ОС Android: получение координат GPS.</p> <p>27. Доступ к оборудованию в ОС Android: акселерометры и гироскопы.</p> <p>28. Анимация и жесты в ОС Android.</p> <p>29. C++ программы для ОС Android. Преимущества и недостатки.</p> <p>30. C++ программы для ОС Android. Задачи, для которых целесообразно применять C++.</p> <p>31. Работа с Android NDK.</p>	
<p>Особенности экосистемы iOS.</p> <p>Особенности разработки приложений под WindowsPhone.</p>	<p>32. Концепция закрытой экосистемы Apple.</p> <p>33. Требования Apple к программам для iOS. Статус AppleDeveloper.</p> <p>34. Одобрение приложений для iOS. Способы распространения приложений iOS.</p> <p>35. Особенности разных версий iOS. Концепции пользовательского интерфейса iOS.</p> <p>36. Программирование на Objective-C: особенности, инструментарий разработки.</p> <p>37. Программирование на Objective-C: классы, методы и обработка событий.</p> <p>38. Сторонний инструментарий для разработки под iOS.</p>	<p>Типовое практическое задание</p>

Типовые практические задания:

Пример 1

Имеется следующий код:

```
public class Overload {
    public void method(Object o) {
        System.out.println("Object");
    }
    public void method(java.io.FileNotFoundException f) {
        System.out.println("FileNotFoundException");
    }
    public void method(java.io.IOException i) {
        System.out.println("IOException");
    }
    public static void main(String args[]) {
        Overload test = new Overload();
        test.method(null);
    }
}
```

Результатом его компиляции и выполнения будет:

1. Ошибка компиляции
2. Ошибка времени выполнения
3. «Object»
4. «FileNotFoundException»
5. «IOException»

Пример 2

```
Float f1 = new Float(Float.NaN);
Float f2 = new Float(Float.NaN);
System.out.println( "" + (f1 == f2) + " " + f1.equals(f2) + " " + (Float.NaN == Float.NaN) );
```

Что будет выведено в результате выполнения данного куска кода:

1. falsefalsefalse
2. falsetruefalse
3. truefalsefalse
4. falsetruefalse
5. truefalsefalse

Пример 3

```
class Mountain {
    static String name = "Himalaya";
    static Mountain getMountain() {
        System.out.println("Getting Name ");
        return null;
    }
    public static void main(String[] args) {
        System.out.println( getMountain().name );
    }
}
```

Что произойдет при попытке выполнения данного кода:

1. Будет выведено «Himalaya» но НЕ будет выведено «Getting Name»
2. Будет выведено «Getting Name» и «Himalaya»
3. Ничего не будет выведено
4. Будет выброшен NullPointerException
5. Будет выведено «Getting Name», а потом выброшено NullPointerException

Пример 4

```
Integer a = 120;
Integer b = 120;
Integer c = 130;
Integer d = 130;
System.out.println(a == b);
System.out.println(c == d);
```

В результате выполнения данного кода будет выведено:

1. true true
2. false false
3. false true
4. true false
5. произойдет ошибка времени выполнения

Пример 5

```
//In File Other.java
package other;
public class Other { public static String hello = "Hello"; }
//In File Test.java
package testPackage;
import other.*;
```

```

class Test{
    public static void main(String[] args) {
        String hello = "Hello", lo = "lo";
        System.out.print((testPackage.Other.hello == hello) + " ");
        System.out.print((other.Other.hello == hello) + " ");
        System.out.print((hello == ("Hel"+"lo")) + " ");
        System.out.print((hello == ("Hel"+lo)) + " ");
        System.out.println(hello == ("Hel"+lo).intern());
    }
}
class Other { static String hello = "Hello"; }

```

В результате мы получим:

1. false true true false true
2. false false true false true
3. true true true true true
4. true true true false true
5. Все ответы неверны

Пример 6

Дана сигнатура метода:

```
public static <E extends CharSequence> List<? super E> doIt(List<E> nums)
```

Который вызывается так:

```
result = doIt(in);
```

Какого типа должны быть result и in?

1. ArrayList<String> in; List<CharSequence> result;
2. List<String> in; List<Object> result;
3. ArrayList<String> in; List result;
4. List<CharSequence> in; List<CharSequence> result;
5. ArrayList<Object> in; List<CharSequence> result;

Пример 7

```

public static void doIt(String String) { //1
    inti = 10;
    i : for (int k = 0 ; k< 10; k++) { //2
        System.out.println( String + i); //3
        if( k*k > 10) continue i; //4
    }
}

```

Данный код:

1. Не скомпилируется из-за строки 1
2. Не скомпилируется из-за строки 2
3. Не скомпилируется из-за строки 3
4. Не скомпилируется из-за строки 4
5. Скомпилируется и запустится без проблем

Пример 8

```

public class Main {
    static void method(int... a) {
        System.out.println("inside int...");
    }
}

```

```

    }
    static void method(long a, long b) {
        System.out.println("inside long");
    }
    static void method(Integer a, Integer b) {
        System.out.println("inside INTEGER");
    }
    public static void main(String[] args) {
        int a = 2;
        int b = 3;
        method(a,b);
    }
}

```

В результате мы получим:

1. Ошибку компиляции
2. Ошибку времени выполнения
3. «insideint...»
4. «insidelong»
5. «inside INTEGER»

Пример 9

```

classSuper{ staticString ID = "QBANK"; }
classSubextendsSuper{
    static { System.out.print("InSub"); }
}
classTest{
    publicstaticvoidmain(String[] args) {
        System.out.println(Sub.ID);
    }
}

```

В результате выполнения данного кода:

1. Он даже не скомпилируется
2. Результат зависит от реализации JVM
3. Будет выведено «QBANK»
4. Будет выведено «InSub» и «QBANK»
5. Все ответы неверны

Пример 10

Имеется два класса:

//in file A.java

```

package p1;
public class A{
    protected inti = 10;
    public intgetI() { return i; }
}

```

//in file B.java

```

package p2;
import p1.*;
public class B extends A{
    public void process(A a) {

```



```

        a.i = a.i*2;
    }
    public static void main(String[] args) {
        A a = new B();
        B b = new B();
        b.process(a);
        System.out.println( a.getI() );
    }
}

```

В результате выполнения класса В мы получим:

1. Будет выведено «20»
2. Будет выведено «10»
3. Код не скомпилируется
4. Возникнет ошибка времени выполнения
5. Все ответы неверны

Сведения о разработке и утверждении рабочей программы дисциплины

Рабочая программа дисциплины Б1.В.ДВ.02.02 «Проектирование и разработка мобильных приложений» составлена в соответствии с ФГОС ВО и утверждена в комплекте с ООП направления **09.03.03 Прикладная информатика.**

Составитель: Маркидонов А.В., д.ф.-м.н., доцент, заведующий кафедрой информатики и вычислительной техники им. В.К. Буторина