Подписано электронной подписью:
Вержицкий Данил Григорьевич
Должность: Директор КГПИ КемГУ
Дата и время: 2023-04-25 00:00:00
Уникальный программный ключ: 6b5279da4e034bff23857cf7a7035bf6fb03a5b6fdf6436

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования
«КЕМЕРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Кузбасский гуманитарно-педагогический институт

Факультет информатики, математики и экономики

УТВЕРЖДАЮ

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

Рабочая программа дисциплины

К.М.04.06 Проектирование и разработка мобильных приложений
Код, название дисциплины

Направление подготовки
09.03.01 Информатика и вычислительная техника
Код, название направления

Направленность (профиль) подготовки
Автоматизированные системы обработки информации и управления

Программа бакалавриата

Квалификация выпускника
бакалавр

Форма обучения
Заочная

Год набора 2023

Новокузнецк 2023

Содержание

1 Цель дисциплины. 3
1.1 Формируемые компетенции, индикаторы достижения компетенций, знания, умения, навыки 3
1.2 Место дисциплины 4
2 Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации. 4
3. Учебно-тематический план и содержание дисциплины. 5
4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации. 5
5 Материально-техническое, программное и учебно-методическое обеспечение дисциплины. 7
5.1 Учебная литература 7
5.2 Материально-техническое и программное обеспечение дисциплины. 7
5.3 Современные профессиональные базы данных и информационные справочные системы. 8
6 Иные сведения и (или) материалы. 8
6.1.Примерные темы письменных учебных работ 8
6.2. Примерные вопросы и задания / задачи для промежуточной аттестации 8

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.

1.1 Формируемые компетенции, индикаторы достижения компетенций, знания, умения, навыки

Таблица 1 – Индикаторы достижения компетенций, 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 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.	**Знать:** - особенности архитектуры и аппаратной среды мобильных устройств; - способы установки мобильных приложений в разных ОС; - особенности архитектуры мобильных устройств с точки зрения программирования; - основные приемы разработки программ для мобильных устройств; - возможности инструментария Java по разработке мобильных приложений; - особенности реализации пользовательского интерфейса в мобильных устройствах; - устройство и архитектуру ОС Android; - основные компоненты архитектуры мобильных платформ; - жизненный цикл мобильных приложений и их структуру; - основные элементы пользовательского интерфейса мобильных приложений; работу с файлами, базами данных, пользовательскими настройками в мобильных устройствах; - инструменты для программирования и основ проектирования мобильных приложений; - возможности программных интерфейсов, обеспечивающих функции телефонии, отправки/получения SMS; возможности взаимодействия с геолокационными, картографическими сервисами; **Уметь:** - инсталлировать программное обеспечение для мобильных устройств; - использовать и применять на практике полученные знания для проектирования и создания мобильных приложений на современном уровне; - программировать и проводить эффективное тестирование программ и приложений для мо-

Код и название компетенции	Индикаторы достижения компетенции, закрепленные за дисциплиной	Знания, умения, навыки (ЗУВ), формируемые дисциплиной
		бильных устройств; - создавать приложения для мобильных устройств. **Владеть:** - навыками написания приложений для мобильных устройств; - практическими навыками работы в специализированной инструментальной среде
ОПК-9 Способен осваивать методики использования программных средств для решения практических задач.	ОПК-9.4. Самостоятельно осваивает и применяет типовые и специализированные программные	**Знать:** - возможности инструментария для разработки приложений для ОС Android. **Уметь:** - самостоятельно разрабатывать приложения и программы для различных платформ и устройств под управлением операционных систем Android; **Владеть:** - современными методами и инструментальными средствами разработки и проектирования программного обеспечения для мобильных устройств.

1.2 Место дисциплины

Дисциплина включена в модуль «Программирование» ОПОП ВО, обязательная часть. Дисциплина осваивается на 2 курс, в летнюю сессию.

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

Таблица 2 – Объем и трудоемкость дисциплины по видам учебных занятий

Общая трудоемкость и виды учебной работы по дисциплине, проводимые в разных формах	Объём часов по формам обучения		
	ОФО	ОЗФО	ЗФО
1 Общая трудоемкость дисциплины			**180**
2 Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)			
Аудиторная работа (всего):			10
в том числе:			
лекции			2
практические занятия, семинары			
практикумы			
лабораторные работы			8
в интерактивной форме			
в электронной форме			
Внеаудиторная работа (всего):			
в том числе, индивидуальная работа обучающихся с преподавателем			

Общая трудоемкость и виды учебной работы по дисциплине, проводимые в разных формах	Объём часов по формам обучения		
	ОФО	ОЗФО	ЗФО
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)			166
4 Промежуточная аттестация обучающегося – **зачет с оценкой**			4

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.)*	Трудоемкость занятий (час.) ЗФО				Формы текущего контроля и промежуточной аттестации успеваемости	
			Аудиторн. занятия			СРС		
			лекц.	практ.	лаб.			
1	Классификация мобильных устройств. Архитектура мобильных устройств и их компонентов.	**36**	2		2	32	Устный опрос, решение учебных задач	Реферат
2	Java для мобильных устройств.	**36**			2	24	Устный опрос, решение учебных задач	
3	Программирование для ОС Android.	**36**			2	20	Устный опрос, решение учебных задач	
4	Особенности экосистемы iOS. Особенности разработки приложений под WindowsPhone.	**36**			2	32	Устный опрос, решение учебных задач	
	Промежуточная аттестация-зачет с оценкой	**4**					4	
	Всего:	**180**	**2**		**8**	**166**		

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	**60** (100% /баллов	Лекционные занятия (9 занятий)	посещение 1 лекционного занятия конспект	7 - 9

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

(посещение занятий по расписанию и выполнение заданий)	приведенной шкалы)	Лабораторные занятия	посещение 1 занятия и выполнение задания на 51-85% 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	реферат написан на «удовлетворительно» реферат написан на «хорошо» 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 (зачет с оценкой)				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 теоретическими основами дисциплины и научной терминологией, грамотно излагает материал, способен иллюстрировать ответ	продвинутый	отлично	86-100

примерами, фактами, данными научных исследований, применять теоретические знания для решения практических профессиональных задач. Правильно интерпретирует полученные результаты и делает обоснованные выводы.			

5 Материально-техническое, программное и 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 Материально-техническое и программное обеспечение дисциплины.

Учебные занятия по дисциплине проводятся в учебных аудиториях КГПИ КемГУ.

Таблица 5

Программа / система	Сведения о праве использования (лицензия, договор, сроки использования).	№ комп. классов
Программное обеспечение компьютеров: Операционные системы: Windows 7; Антивирусное ПО: Eset Endpoint Security 5.0.		
Лицензионное ПО		
Офисное ПО		
Microsoft Office, Visio	Лиценизия DreamSpark Premium Electronic Software Delivery (3 years) Renewal по сублицензионному договору	**501, 502, 508, 509, 602**
Интерегрированные программые системы		
Microsoft Visual Studio 2010	Лиценизия DreamSpark Premium Electronic Software Delivery (3 years) Renewal по сублицензионному договору	**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**

5.3 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 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 зачету с оценкой

Разделы и темы	Примерные теоретические вопросы	Примерные практические задания / задачи
Классификация мобильных устройств. Архитектура мобильных устройств и их компонентов.	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.	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.	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 обработчиков событий. 23. Vодель документ/представление в мобильном программировании. 24. Доступ к оборудованию в ОС Android (общие принципы). 25. Доступ к оборудованию в ОС Android: получение снимка видеокамерой. 26. Доступ к оборудованию в ОС Android: получение координат GPS. 27. Доступ к оборудованию в ОС Android: акселерометры и гироскопы. 28. Анимация и жесты в ОС Android. 29. C++ программы для ОС Android. Преимущества и недостатки. 30. C++ программы для ОС Android. Задачи, для которых целесообразно применять C++. 31. Работа с Android NDK.	Типовое практическое задание
Особенности экосистемы iOS. Особенности разработки приложений под WindowsPhone.	32. Концепция закрытой экосистемы Apple. 33. Требования Apple к программам для iOS. Статус AppleDeveloper. 34. Одобрение приложений для iOS. Способы распространения приложений iOS.	Типовое практическое задание

	35. Особенности разных версий iOS. Концепции пользовательского интерфейса iOS. 36. Программирование на Objective-C: особенности, инструментарий разработки. 37. Программирование на Objective-C: классы, методы и обработка событий. 38. Сторонний инструментарий для разработки под iOS.	

Типовые практические задания:

Пример 1

Имеется следующий код:

```
publicclassOverload{
    publicvoidmethod(Object o) {
        System.out.println("Object");
    }
    publicvoidmethod(java.io.FileNotFoundException f) {
        System.out.println("FileNotFoundException");
    }
    publicvoidmethod(java.io.IOException i) {
        System.out.println("IOException");
    }
    publicstaticvoidmain(Stringargs[]) {
        Overloadtest = newOverload();
        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 = newFloat(Float.NaN);
Float f2 = newFloat(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. truetruefalse
4. falsetruetrue
5. truetruetrue

Пример 3

```
classMountain {
    staticStringname = "Himalaya";
    staticMountaingetMountain() {
        System.out.println("GettingName ");
        returnnull;
```

```
    }
    publicstaticvoidmain(String[ ] args) {
        System.out.println( getMountain().name );
    }
}
```

Что произойдет при попытке выполнения данного кода:
1. Будет выведено «Himalaya» но НЕ будет выведено «GettingName»
2. Будет выведено «Getting Name» и «Himalaya»
3. Ничего не будет выведено
4. Будет выброшен NullPointerException
5. Будет выведено «GettingName», 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. truetrue
2. falsefalse
3. falsetrue
4. truefalse
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. falsetruetruefalsetrue
2. falsefalsetruefalsetrue
3. truetruetruetruetrue
4. truetruetruefalsetrue

5. Все ответы неверны

Пример 6
Дана сигнатура метода:
publicstatic<E extendsCharSequence>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 B мы получим:
1. Будет выведено «20»
2. Будет выведено «10»
3. Код не скомпилирутся
4. Возникнет ошибка времени выполнения
5. Все ответы неверны

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