

ФГБОУ ВО «БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Утверждено:  
на заседании кафедры  
протокол № 7 от « 26 » января 20 21 г.  
Зав. кафедрой \_\_\_\_\_ /Мустафина С.А.

Согласовано:  
Председатель УМК факультета  
\_\_\_\_\_ / Ефимов А.М.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

дисциплина Язык программирования Java и Java-технологии

Факультатив

**программа бакалавриата**

Направление подготовки (специальность)  
01.03.02 «Прикладная математика и информатика»

Направленность (профиль) подготовки  
Математическое моделирование, численные методы и комплексы программ  
Мобильные, облачные и интеллектуальные технологии

Квалификация  
бакалавр

Разработчик (составитель) Доцент, к.ф.-м.н.	_____ /Ткачев В.И.
Ассистент	_____ /Ахметьянова А.И.

Для приема: 2020

Уфа 2021 г.

Составитель / составители: Ткачев В.И., Ахметьянова А.И..

Рабочая программа дисциплины *утверждена* на заседании кафедры протокол № 7 от «26» января 2021 г. № 7

## **Список документов и материалов**

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с установленными в образовательной программе индикаторами достижения компетенций
2. Цель и место дисциплины в структуре образовательной программы
3. Содержание рабочей программы (объем дисциплины, типы и виды учебных занятий, учебно-методическое обеспечение самостоятельной работы обучающихся)
4. Фонд оценочных средств по дисциплине
  - 4.1. Перечень компетенций и индикаторов достижения компетенций с указанием соотнесенных с ними запланированных результатов обучения по дисциплине. Описание критериев и шкал оценивания результатов обучения по дисциплине.
  - 4.2. Типовые контрольные задания или иные материалы, необходимые для оценивания результатов обучения по дисциплине, соотнесенных с установленными в образовательной программе индикаторами достижения компетенций. Методические материалы, определяющие процедуры оценивания результатов обучения по дисциплине.
5. Учебно-методическое и информационное обеспечение дисциплины
  - 5.1. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины
  - 5.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» и программного обеспечения, необходимых для освоения дисциплины, включая профессиональные базы данных и информационные справочные системы
6. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

# 1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с установленными в образовательной программе индикаторами достижения компетенций

По итогам освоения дисциплины обучающийся должен достичь следующих результатов обучения:

Категория (группа) компетенций	Формируемая компетенция (с указанием кода)	Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине
Теоретические и практические основы профессиональной деятельности	ОПК-2 Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач	ОПК-2.1. Обладает фундаментальными знаниями по существующим математическим методам и системам программирования для разработки и реализации алгоритмов решения прикладных задач	Знать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач
		ОПК-2.2. Умеет использовать аппарат существующих математических методов и систем программирования для разработки и реализации алгоритмов решения прикладных задач в профессиональной деятельности.	Уметь использовать аппарат существующих математических методов и систем программирования для разработки и реализации алгоритмов решения прикладных задач в профессиональной деятельности.
		ОПК-2.3. Имеет навыки применения аппарата существующих математических методов и систем программирования для разработки и реализации алгоритмов при решении конкретных задач.	Владеть способностью применения аппарата существующих математических методов и систем программирования для разработки и реализации алгоритмов при решении конкретных задач
	ПК-3 Способен использовать основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.	ПК-3.1. Знает основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.	Знает основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.
		ПК-3.2. Умеет программировать в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования.	Умеет программировать в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования.
		ПК-3.3. Имеет практический опыт разработки программ в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования.	Имеет практический опыт разработки программ в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования.

Категория (группа) компетенций	Формируемая компетенция (с указанием кода)	Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине
	ПК-4 Способен использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования	ПК-4.1. Знает современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования.	Знает современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования.
		ПК-4.2. Умеет разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования.	Умеет разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования.
		ПК-4.3. Имеет практический опыт разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования.	Имеет практический опыт разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования.

## 2. Цель и место дисциплины в структуре образовательной программы

Дисциплина «Язык программирования Java и Java-технологии» является факультативной дисциплиной. Дисциплина изучается на 2 курсе в 1-ом и 2-ом семестрах.

Цели изучения дисциплины:

- освоение теоретического и практического материала на занятиях;
- самостоятельная практическая работа: выполнение домашних заданий, мини проектов (небольшие приложения, которые реализуются учениками преимущественно на занятиях совместно с учителем с небольшими самостоятельными доработками в качестве домашнего задания);
- промежуточная помодульная аттестация в форме электронного тестирования;
- разработка и защита индивидуального проекта;

Для освоения дисциплины необходимы компетенции, сформированные в рамках изучения следующих дисциплин: «Практикум на ЭВМ», «Языки и методы программирования», устойчивые навыки программирования на любом алгоритмическом языке

## 3. Содержание рабочей программы (объем дисциплины, типы и виды учебных занятий, учебно-методическое обеспечение самостоятельной работы обучающихся)

Содержание рабочей программы представлено в Приложении № 1.

#### 4. Фонд оценочных средств по дисциплине

##### 4.1. Перечень компетенций и индикаторов достижения компетенций с указанием соотнесенных с ними запланированных результатов обучения по дисциплине. Описание критериев и шкал оценивания результатов обучения по дисциплине.

Код и формулировка компетенции: ОПК-2. Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач.

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Критерии оценивания результатов обучения	
		(«Не зачтено»)	(«Зачтено»)
ОПК-2.1. Обладает фундаментальными знаниями по существующим математическим методам и системам программирования для разработки и реализации алгоритмов решения прикладных задач	Знать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач	Фрагментарные представления о существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач.	Сформированные, но содержащие возможно отдельные пробелы представления о существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач.
ОПК-2.2. Умеет использовать аппарат существующих математических методов и систем программирования для разработки и реализации алгоритмов решения прикладных задач в профессиональной деятельности.	Уметь использовать аппарат существующих математических методов и систем программирования для разработки и реализации алгоритмов решения прикладных задач в профессиональной деятельности.	Фрагментарные умения использовать аппарат существующих математических методов и систем программирования для разработки и реализации алгоритмов решения прикладных задач в профессиональной деятельности.	В целом успешное, систематическое, но возможно содержащее отдельные пробелы умение использовать аппарат существующих математических методов и систем программирования для разработки и реализации алгоритмов решения прикладных задач в профессиональной деятельности.
ОПК-2.3. Имеет навыки применения аппарата существующих математических методов и систем программирования для разработки и реализации алгоритмов при решении конкретных задач.	Владеть способностью применения аппарата существующих математических методов и систем программирования для разработки и реализации алгоритмов при решении конкретных задач	Фрагментарное владение способностью применения аппарата существующих математических методов и систем программирования для разработки и реализации алгоритмов при решении конкретных задач	В целом успешное, систематическое, но возможно содержащее отдельные пробелы применение способности применения аппарата существующих математических методов и систем программирования для разработки и реализации алгоритмов при решении конкретных задач.

Код и формулировка компетенции: ПК-3. Способен использовать основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Критерии оценивания результатов обучения	
		(«Не зачтено»)	(«Зачтено»)
ПК-3.1. Знает основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.	Знает основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.	Фрагментарные представления об основных концептуальных положениях функционального, логического, объектно-ориентированного и визуального направлений программирования, методах, способах и средствах разработки программ в рамках этих направлений	Сформированные, но содержащие возможно отдельные пробелы представления об основных концептуальных положениях функционального, логического, объектно-ориентированного и визуального направлений программирования, методах, способах и средствах разработки программ в рамках этих направлений.
ПК-3.2. Умеет программировать в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования	Умеет программировать в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования.	Фрагментарные умения программировать в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования.	В целом успешное, систематическое, но возможно содержащее отдельные пробелы умение программировать в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования
ПК-3.2. Умеет программировать в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования	Имеет практический опыт разработки программ в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования	Фрагментарное владение способностью разработки программ в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования.	В целом успешное, систематическое, но возможно содержащее отдельные пробелы применение способности разработки программ в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования

Код и формулировка компетенции: ПК-4. Способен использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Критерии оценивания результатов обучения	
		(«Не зачтено»)	(«Зачтено»)
ПК-4.1. Знает современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования	Знает современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования.	Фрагментарные представления о современных методах разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования.	Сформированные, но содержащие возможно отдельные пробелы представления о современных методах разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования.
ПК-4.2. Умеет разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	Умеет разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	Фрагментарные умения разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования.	В целом успешное, систематическое, но возможно содержащее отдельные пробелы умение разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования.
ПК-4.3. Имеет практический опыт разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования.	Имеет практический опыт разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования.	Фрагментарное владение способностью разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования.	В целом успешное, систематическое, но возможно содержащее отдельные пробелы применение способности разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования.



**4.2. Типовые контрольные задания или иные материалы, необходимые для оценивания результатов обучения по дисциплине, соотнесенных с установленными в образовательной программе индикаторами достижения компетенций. Методические материалы, определяющие процедуры оценивания результатов обучения по дисциплине.**

<b>Код и наименование индикатора достижения компетенции</b>	<b>Результаты обучения по дисциплине</b>	<b>Оценочные средства</b>
ОПК-2.1. Обладает фундаментальными знаниями по существующим математическим методам и системам программирования для разработки и реализации алгоритмов решения прикладных задач	Знать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач.	Индивидуальный, групповой опрос; контрольная работа, собеседование
ПК-3.1. Знает основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.	Знает основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.	
ПК-4.1. Знает современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования.	Знает современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования	
ОПК-2.2. Умеет использовать аппарат существующих математических методов и систем программирования для разработки и реализации алгоритмов решения прикладных задач в профессиональной деятельности	Уметь использовать аппарат существующих математических методов и систем программирования для разработки и реализации алгоритмов решения прикладных задач в профессиональной деятельности	Индивидуальный, групповой опрос; лабораторные работы; собеседование
ПК-3.2. Умеет программировать в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования.	Умеет программировать в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования.	
ПК-4.2. Умеет разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	Умеет разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	
ОПК-2.3. Имеет навыки применения аппарата существующих математических методов и систем программирования для разработки и реализации алгоритмов при решении конкретных задач.	Владеть способностью применения аппарата существующих математических методов и систем программирования для разработки и реализации алгоритмов при решении конкретных задач	Практическое задание, РГР; экзамен

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные средства
ПК-3.3. Имеет практический опыт разработки программ в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования.	Имеет практический опыт разработки программ в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования.	
ПК-4.3. Имеет практический опыт разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования.	Имеет практический опыт разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования.	

Критериями оценивания при *модульно-рейтинговой системе* являются баллы, которые выставляются преподавателем за виды деятельности (оценочные средства) по итогам изучения модулей (разделов дисциплины), перечисленных в рейтинг-плане дисциплины (*для зачета*: текущий контроль – максимум 50 баллов; рубежный контроль – максимум 50 баллов, поощрительные баллы – максимум 10).

Шкалы оценивания:

(*для зачета*):

зачтено – от 60 до 110 рейтинговых баллов (включая 10 поощрительных баллов),  
не зачтено – от 0 до 59 рейтинговых баллов).

## Рейтинг – план дисциплины

### Язык программирования Java и Java-технологии

(название дисциплины согласно рабочему учебному плану)

направление/специальность 01.03.02 Прикладная математика и информатика

курс 2, семестр 2 (3)

Виды учебной деятельности студентов	Балл за конкретное задание	Число заданий за семестр	Баллы	
			Минимальный	Максимальный
<b>Модуль 1 Основы программирования на языке Java</b>				
<b>Текущий контроль</b>				
1. Аудиторная работа	10	2	0	20
2. Тестовый контроль	10	1	0	10
<b>Модуль 2 Введение в объектно-ориентированное программирование</b>				
<b>Текущий контроль</b>				
1. Аудиторная работа	10	2	0	20
2. Тестовый контроль	10	1	0	10
<b>Модуль 3 Основы программирования Android приложений</b>				
<b>Текущий контроль</b>				
1. Аудиторная работа	10	2	0	20
2. Тестовый контроль	10	2	0	20
<b>Поощрительные баллы</b>				
1. Студенческая олимпиада	10	1	0	<b>10</b>
2. Публикация статей				
3. Творческие задания (выступление, презентация)				
<b>Посещаемость (баллы вычитаются из общей суммы набранных баллов)</b>				
1. Посещение лекционных занятий			<b>0</b>	<b>-6</b>
2. Посещение практических (семинарских, лабораторных занятий)			<b>0</b>	<b>-10</b>
<b>Итоговый контроль</b>				
1. Зачет (дифференцированный зачет)				

курс 2, семестр 2 (4)

Виды учебной деятельности студентов	Балл за конкретное задание	Число заданий за семестр	Баллы	
			Минимальный	Максимальный
<b>Модуль 4 Алгоритмы и структуры данных на языке Java</b>				
<b>Текущий контроль</b>				
1. Аудиторная работа	10	2	0	20
2. Тестовый контроль	10	2	0	20
<b>Рубежный контроль</b>				
1. Письменная контрольная работа	10	1	0	10
<b>Модуль 5 Основы разработки серверной части мобильных приложений</b>				
<b>Текущий контроль</b>				
1. Аудиторная работа	10	2	0	20
2. Тестовый контроль	10	2	0	20
<b>Рубежный контроль</b>				
1. Письменная контрольная работа	10	1	0	10
<b>Поощрительные баллы</b>				
4. Студенческая олимпиада	10	1	0	<b>10</b>
5. Публикация статей				
6. Творческие задания (выступление, презентация)				
<b>Посещаемость (баллы вычитаются из общей суммы набранных баллов)</b>				
3. Посещение лекционных занятий			<b>0</b>	<b>-6</b>
4. Посещение практических (семинарских, лабораторных занятий)			<b>0</b>	<b>-10</b>
<b>Итоговый контроль</b>				
1. Зачет (дифференцированный зачет)				

## Примерный перечень вопросов для индивидуального, группового опроса, собеседования

1. Мобильное программирование, платформы для разработки. Виды мобильных приложений и их архитектура.
2. Области применения языка Java. Объявление класса. Стандартная библиотека классов. Классы-оболочки, автоупаковка и автораспаковка.
3. Основные понятия ООП. Наследование.
4. Основные понятия ООП. Полиморфизм.
5. Основные понятия ООП. Инкапсуляция.
6. Модификаторы доступа в Java.
7. Интерфейсы. Абстрактные классы.
8. Структура Android проекта.
9. Структура Android проекта. Android Manifest.
10. Структура Android проекта. Ресурсы Android приложения.
11. Разработка UI Android приложения.
12. Меню Android приложения: параметров, контекстного меню, всплывающего меню
13. Жизненный цикл Activity.
14. Элементы экрана и их свойства.
15. XML-разметка для UI Android приложения.
16. Обработчики событий. OnClickListener, TextWatcher
17. Адаптеры в Android. Base Adapter.
18. Диалоговые окна. Класс Dialog.
19. Намерения (Intent). Объект Intent. Явные и неявные намерения.
20. Намерения (Intent). IntentFilter.
21. Получение результата операции. Метод startActivityForResult.
22. Сохранение данных Activity при повороте экрана. Метод onSaveInstanceState.
23. Хранение данных. Preferences.
24. Всплывающие сообщения. Toasts, snackBar.
25. Широковещательные сообщения. Класс Broadcast Receiver.
26. Уведомления (Notifications) в Android. Работа с Notifications.
27. Службы (Services) в Android.
28. Работа с файлами. Сохранение файлов. Сохранение файла во внутренней памяти. Сохранение файла во внешнем хранилище.
29. Работа с файлами. Выбор внутреннего или внешнего хранилища. Запрос доступного пространства. Удаление файла.
30. Рисование в Android. Доступ к Canvas. Canvas-преобразования. Методы save и restore. Метод onDraw().
31. Работа с анимацией.
32. Работа с медиафайлами. MediaPlayer – аудио/видео плеер, основные возможности.
33. Отладка Android приложений.
34. Обработка исключений (Exceptions) в Android.
35. Сигнализация. Отложенная сигнализация.
36. Картографические сервисы. Фоновые службы и процессы.
37. Язык AIDL . Служба компоновки и создания фрагментов.
38. Управление сенсорами. Управление сетевыми соединениями.
39. Получение информации об устройстве. Служба отправки и получения СМС.
40. Поддержка протоколов Bluetooth /Wi -Fi . Установка шлюза через Wi -Fi Direct
41. Управление анимацией. Использование NFC. Служба push - нотификаций.
42. Служба уведомлений и доставки. Управление потоками и асинхронными задачами.
43. Мобильное программирование, платформы для разработки. Виды мобильных приложений и их архитектура.

44. Области применения языка Java. Объявление класса. Стандартная библиотека классов. Классы-оболочки, автоупаковка и автораспаковка.
45. Процессы и потоки в Android. AsyncTask
46. Очередь сообщений в Android. Класс Handler.
47. Отличия декларативного программирования от императивного. Проникновение декларативного программирования в современные языки программирования.
48. Рекурсия. Области применения. Хвостовая Рекурсия. Оптимизация хвостовой рекурсии.
49. Байт-код виртуальных машин. На примере Python и Java.
50. Оптимизации, применяемые к функциям без побочных эффектов.
51. Лицензии для ПО. Open Source. Свободный (libre) софт. Использование чужого кода в своих разработках.
52. Системы контроля версий. GIT. Основные возможности. Работа с клиентом командной строки.
53. Библиотеки для мат. вычислений в Python.
54. Декоратор как шаблон проектирования. Декораторы функций и классов в Python.
55. Событийное программирование. Особенности.
56. Асинхронные приложения. Способы поддержки асинхронности.
57. Базовые императивные конструкции. Циклы. Условия. Последовательное выполнение. Блоки. Функции.
58. Основные встроенные типы данных в современных языках программирования.
59. ООП. Особенности применения. Недостатки. Сравнение с функциональным программированием.
60. Технологии быстрой разработки ПО.
61. Математические основы программирования. Вычислимость. Машина Тьюринга.
62. Методы отладки приложений, в том числе удаленных.
63. Разработка мобильных приложений. Особенности работы с iOS и Android.
64. Тестирование приложений. Дос-тесты, Unit-тесты. Тестирование черного и белого ящика.
65. Создание собственных модулей. Выкладка их в общий репозиторий на PyPi. Создание инсталляционных пакетов.

#### 1) Задача A + B

Вы должны написать полностью программу на Java: необходимые директивы import, один public класс в пакете по умолчанию (в коде отсутствует подстрока package) с функцией main(), которая решает задачу.

Ввод осуществляется с консоли, вывод - на консоль.

Программа должна работать **точно по протоколу**, который показан в примерах. Например, нельзя выводить ничего лишнего, скажем, фразу "Введите N".

#### 2) Последняя цифра

Дано целое положительное число NN. Выведите его последнюю цифру.

Входные данные

Единственная строка содержит число NN ( $1 \leq N \leq 109$ ) ( $1 \leq N \leq 109$ ).

Выходные данные

Выведите ответ на задачу.

#### Пример:

Входные данные	Выходные данные

Входные данные	Выходные данные
179	9
40	0

### 3) Сумма цифр

Дано трехзначное число NN. Найдите сумму его цифр.

Входные данные

Вводится число NN ( $100 \leq N \leq 999$ ).

Выходные данные

Выведите сумму цифр числа NN.

**Пример:**

Входные данные	Выходные данные
179	17
829	19

### 4) Следующее четное

Дано целое положительное число NN. Выведите следующее за ним **четное** число. При решении этой задачи нельзя использовать условную инструкцию if и циклы.

Входные данные

Вводится число N ( $1 \leq N \leq 109$ ).

Выходные данные

Выведите наименьшее четное число большее, чем NN.

**Пример:**

Входные данные	Выходные данные
7	8
8	10

### 5) Стоимость покупки

Пирожок в столовой стоит aa рублей и bb копеек. Определите, сколько рублей и копеек нужно заплатить за nn пирожков.

Входные данные

Программа получает на вход три числа: a ( $0 \leq a \leq 3000$ ), b ( $0 \leq b \leq 99$ ), n ( $0 \leq n \leq 3000$ ).

Выходные данные

Программа должна вывести в единственной строке два целых числа через пробел - стоимость покупки в рублях и копейках.

**Пример:**

Входные данные	Выходные данные
10	20 30

Входные данные	Выходные данные
15 2	
2 50 4	10 0

б) Электронные часы - 2

Электронные часы показывают время в формате **h:mm:ss**, то есть сначала записывается количество часов, потом обязательно **двузначное** количество минут, затем обязательно двузначное количество секунд. Количество минут и секунд при необходимости дополняются до двузначного числа нулями.

С начала суток прошло  $n$  секунд. Выведите, что покажут часы.

Входные данные

Вводится целое число  $n$ .

Выходные данные

Выведите ответ на задачу, соблюдая требуемый формат.

**Пример:**

Входные данные	Выходные данные
3602	1:00:02
129700	12:01:40

7) Придумайте иерархию 3-4 объектов из реальной жизни и опишите инкапсулированные свойства и методы их родительского класса, свойства и методы наследуемых от них классов (отличающиеся от родителя) и полиморфные методы, применимые к ним.

Например, животные: кенгуру, слон, карась. Способ передвижения, способ получения и выращивания потомства.

В ответе на задание нужно прикрепить текстовый файл со словесным описанием объектов (.doc, .txt, .rtf). Максимальный размер файла 500 Кб.

8) Модифицировать приложение из Задания 2.4.3 так, чтобы оно решало квадратное уравнение  $ax^2+bx+c=0$ , а так же корректно обрабатывало ситуацию нулевых коэффициентов.

9) Создайте активность со следующими элементами:

- полями ввода логина и пароля: они должны задаваться в коде в виде констант и содержать подсказку (hint);
- кнопкой «Вход»;
- текстовым полем, отображающим верно ли введен пароль: если верно, то вывести зеленым цветом «Верно», если не верно - красным «Вы ошиблись в логине или пароле», при этом поля ввода очищаются.

Проект Android Studio заархивируйте, и архив приложите к заданию.

10) Придумайте тему своего минипроекта - текстового квеста. Например это могут быть квест по мотивам популярной игры или симулятор выживания или квест из какой либо области знаний, например информатики. По приведенному в главе образцу для своего квеста проведите проектирование. То есть:

- проведите анализ задания,
- синтез моделей,
- постройте диаграмму классов UML,
- создайте все нужные классы,
- продемонстрируйте работу вашего квеста.

В дальнейшем можно развивать этот минипроект, и возможно удастся создать полноценную игру-квест для Android.

11) Реализация посимвольного сравнения двух файлов или страниц в интернете. Выводить требуется все отличающиеся символы в произвольном формате. Если символов очень много нужно вывести только часть и количество различий.

12) Разработать пример программы, совершающей загрузку картинки из интернета и устанавливающей ее на экран.

Картинку из интернета можно загрузить по ссылке. Для работы с ссылками в Android есть класс **URL**. Один из конструкторов класса **URL** позволяет передать в него текстовую ссылку:

```
URL url = new URL("http://...");
```

Используя метод **getContent()** для объекта **URL** можно получить ресурс который располагается по ссылке **URL**. Метод возвращает поток данных **InputStream**, или **null** в случае неудачи. Графический ресурс **Bitmap** можно получить из потока **InputStream** с помощью метода **BitmapFactory.decodeStream(InputStream)**

13) Напишите метод, который сохранял бы уже вычисленные значения Фибоначчи (например, в **ArrayList<Long>**) и использовал их для дальнейших вычислений.

14) Реализуйте класс, который, используя **TreeSet** находит список учеников, сдавших экзамен на оценку выше заданного числа. Для решения задачи нужно реализовать собственный класс **Student**, реализующий интерфейс **Comparable**.

15) Узнать собственный внешний IP-адрес с помощью сервиса: <http://2ip.ru/>. Убедиться, что внешний IP-адрес у всех в классе одинаковый.

В сервисе <http://www.nic.ru/whois/> изучить информацию о собственном внешнем IP-адресе. Можно также посмотреть информацию об IP-адресах: 8.8.8.8, 87.240.131.99, 211.45.27.198 или 173.194.71.113.

16) Опробовать все изученные команды. В процессе выполнения записать IP-адреса популярных сервисов и время ответа на команду **ping**.

17) Произведите при помощи утилиты **telnet** обмен по протоколу **HTTP** веб сервером, например для URL <http://www.samsung.com/ru/> . Скриншоты обмена приложите к ответу.

18) 1. Создайте свой аккаунт на <https://www.heroku.com/>. В правом верхнем углу нажмите на кнопку **Sign up** для регистрации:

Log in or

Sign up



Заполните открывшуюся форму. В поле “Pick your primary development language” укажите язык Java.

2. Изучите описание платформы. Какие решения она поддерживает?

19) Напишите свое приложение для веб сервера. Приложение должно работать следующим образом:

- клиентское приложений передает Фамилию Имя и Класс ученика,
- после нажатия кнопки отправить сервер показывает список учеников.

В качестве клиента используйте HTML форму. К ответу приложите архив с серверным приложением, файл с HTML формой, скриншот работающего приложения.

### **Критерии оценки (в баллах):**

- **25-30 баллов** выставляется студенту, если студент дал полные, развернутые ответы на все теоретические вопросы билета, продемонстрировал знание функциональных возможностей, терминологии, основных элементов, умение применять теоретические знания при выполнении практических заданий. Студент без затруднений ответил на все дополнительные вопросы. Практическая часть работы выполнена полностью без неточностей и ошибок;

- **17-24 баллов** выставляется студенту, если студент раскрыл в основном теоретические вопросы, однако допущены неточности в определении основных понятий. При ответе на дополнительные вопросы допущены небольшие неточности. При выполнении практической части работы допущены несущественные ошибки;

- **10-16 баллов** выставляется студенту, если при ответе на теоретические вопросы студентом допущено несколько существенных ошибок в толковании основных понятий. Логика и полнота ответа страдают заметными изъянами. Заметны пробелы в знании основных методов. Теоретические вопросы в целом изложены достаточно, но с пропусками материала. Имеются принципиальные ошибки в логике построения ответа на вопрос. Студент не решил задачу или при решении допущены грубые ошибки;

- **0-10 баллов** выставляется студенту, если он отказался от ответа или не смог ответить на вопросы билета, ответ на теоретические вопросы свидетельствует о непонимании и крайне неполном знании основных понятий и методов. Обнаруживается отсутствие навыков применения теоретических знаний при выполнении практических заданий. Студент не смог ответить ни на один дополнительный вопрос.

## **5. Учебно-методическое и информационное обеспечение дисциплины**

### **5.1. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины**

#### **Основная литература:**

1. Хабитуев, Б. В. Программирование на языке Java: практикум : учебное пособие / Б. В. Хабитуев. — Улан-Удэ : БГУ, 2020. — 94 с. — ISBN 978-5-9793-1548-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/171791>
2. Строганкова, Н. В. Шаблоны программных платформ языка Java : учебное пособие / Н. В. Строганкова, К. В. Касьяненко, А. В. Хозяинов ; составители практикума входят: использование. — Москва : РТУ МИРЭА, 2021. — 83 с. — Текст : электронный // Лань :

электронно-библиотечная система. — URL: <https://e.lanbook.com/book/182466>

3. Вязовик, Н. А. Программирование на Java : учебное пособие / Н. А. Вязовик. — 2-е изд. — Москва : ИНТУИТ, 2016. — 603 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/100405>

#### **Дополнительная литература:**

4. Наир, В. Предметно-ориентированное проектирование в Enterprise Java : руководство / В. Наир ; перевод с английского А. В. Снастина. — Москва : ДМК Пресс, 2020. — 306 с. — ISBN 978-5-97060-872-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/179503>

5. Мархакшинов, А. Л. Практикум по программированию на языке Java : учебное пособие / А. Л. Мархакшинов. — Улан-Удэ : БГУ, 2017. — 70 с. — ISBN 978-5-9793-0016-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/154292>

### **5.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» и программного обеспечения, необходимых для освоения дисциплины, включая профессиональные базы данных и информационные справочные системы**

1. <http://www.socionet.ru/> – Онлайн-овая научная инфраструктура, научно образовательная социальная сеть;

2. <http://www.msu.ru/> – МГУ им. М.В. Ломоносова;

3. <http://window.edu.ru/> – Наиболее обширная электронная база учебников и методических материалов на сайте информационной системы «Единое окно доступа к образовательным ресурсам»;

4. <http://www.rusneb.ru/> – Национальная электронная библиотека (НЭБ);

5. <http://enip.ras.ru/> – Единое научное информационное пространство РАН;

6. <http://univertv.ru/video/matematika/> – Открытый образовательный видеопортал Uni-verTV.ru. Образовательные фильмы на различные темы. Лекции в ведущих российских и зарубежных вузах. Научная конференция или научно-популярная лекция по интересующему вопросу);

7. <http://eqworld.ipmnet.ru/ru/library.htm> – Учебно-образовательная физико-математическая библиотека;

8. [www.lib.mexmat.ru/books/41](http://www.lib.mexmat.ru/books/41) – Электронная библиотека МГУ;

9. [www.newlibrary.ru](http://www.newlibrary.ru) – Новая электронная библиотека;

10. [www.edu.ru](http://www.edu.ru) – Федеральный портал российского образования;

11. [www.mathnet.ru](http://www.mathnet.ru) – Общероссийский математический портал;

12. [www.elibrary.ru](http://www.elibrary.ru) – Научная электронная библиотека;
13. [www.mathburo.ru](http://www.mathburo.ru) – Матбюро: решения задач по высшей математике;
14. [www.nehudlit.ru](http://www.nehudlit.ru) – Электронная библиотека учебных материалов;
15. [www.wolframcenter.ru/instructors](http://www.wolframcenter.ru/instructors) – Wolfram Mathematica, Русскоязычная поддержка,

**6. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине**

<i>Наименование специализированных аудиторий, кабинетов, лабораторий</i>	<i>Вид занятий</i>	<i>Наименование оборудования, программного обеспечения</i>
<i>1</i>	<i>2</i>	<i>3</i>
Аудитория № 528 (физико-математический корпус - учебное)	Лекции	Компьютер, мультимедийный проектор, экран, доска, учебная мебель, доска настенная меловая
Лаборатория аудитория № 520а (физико-математический корпус - учебное), № 521 (физико-математический корпус - учебное), аудитория № 522 (физико-математический корпус - учебное), аудитория № 524 (физико-математический корпус - учебное), аудитория № 525 (физико-математический корпус - учебное)	Лабораторные работы	<p><b>Аудитория № 520а</b> Учебная мебель, доска, монитор LG 19 L1942S SF 1280 x 1024,5ms,8000:1,black (3,4 кг,VGA,19"(48,3см)5мс, мониторы LG 19" L1942SBF 1280x1024,5ms,8000:1,black 10 шт., системный блок HPPavilionSlimlineS3500FAMD Athlon64 X2 5400+/2.8GHz,4Gb,500Gb 12шт.,доска аудитор. ДА36.</p> <p><b>Аудитория № 521</b> Учебная мебель, доска, коммутатор HP V1905-24 Switch 24*10/100+2*10/100/1000, персональные компьютеры в комплекте DEPO Neos 460MDi5 2300/4GDDR1333/T500G/DVD W – 12 шт., проектор Optoma EX542i.DLP3D.XGA(1024*768).2700 ANSI Lm.3000 1.Lamp5000+/-40 ver, шкаф TLKTWP-065442-G-GY, экран на штативе DraperDiplomat (1:1) 84/84* 213*213 MW, доска аудитор. ДА36.</p> <p><b>Аудитория № 522</b> Учебная мебель, доска, персональный компьютер LenovoThinkCentre A70z IntelPentium E 5800, 320 Gb, 19" – 13 шт., кондиционер LessarLS/LUH24KB2.</p> <p><b>Аудитория № 524</b> Учебная мебель, доска настенная меловая, коммутатор HP V1905-24 Switch 24*10/100+2*10/100/1000, персональный компьютер в комплекте HP AiO 20"CQ 100 eu – 27 шт., экран ScreeMediaGolgview 274*206 NW 4:3, универсальное потолочное крепление ScreeMedia для проектора, регулировка высоты , шкаф TLKTWP-065442-G-GY, патч-корд (1296), доска аудитор. ДА32.</p> <p><b>Аудитория № 525</b> Учебная мебель, доска, персональные компьютеры в комплекте DEPO Neos 460MDi5 2300/4GDDR1333/T500G/DVDW/ - 13 шт., доска аудитор. ДА32</p> <p>1. Windows 8 Russian. Windows Professional 8 Russian Upgrade. Договор № 104 от 17.06.2013 г. Лицензии бессрочные. 2. Microsoft Office Standard 2013 Russian. Договор № 114 от 12.11.2014 г. Лицензии бессрочные. 3. Среда разработки Microsoft Visual Studio Community 2017 (Условия лицензии на программное обеспечение Microsoft Visual Studio Community 2017, свободное программное обеспечение). 4. AcademicEdition Networked Volume Licenses RAD Studio XE3 Professional Concurrent AppWaveEnglish; договор №263 от 07.12.2012 г.</p>

ФГБОУ ВО «БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

**СОДЕРЖАНИЕ РАБОЧЕЙ ПРОГРАММЫ**

дисциплины *Язык программирования Java и Java-технологии* на 3 и 4 семестр

очная форма обучения

Вид работы	Объем дисциплины
Общая трудоемкость дисциплины (з.е. / часов)	1/36
Учебных часов на контактную работу с преподавателем:	
лекций	
практических/ семинарских	
лабораторных	32
других (групповая, индивидуальная консультация и иные виды учебной деятельности, предусматривающие работу обучающихся с преподавателем) (ФКР)	0,2
Учебных часов на самостоятельную работу обучающихся (СР)	3.8
Учебных часов на подготовку к экзамену/зачету/дифференцированному зачету (Контроль)	

Форма(ы) контроля:

Зачет 3 семестр

Вид работы	Объем дисциплины
Общая трудоемкость дисциплины (з.е. / часов)	1/36
Учебных часов на контактную работу с преподавателем:	
лекций	
практических/ семинарских	
лабораторных	32
других (групповая, индивидуальная консультация и иные виды учебной деятельности, предусматривающие работу обучающихся с преподавателем) (ФКР)	0,2
Учебных часов на самостоятельную работу обучающихся (СР)	3.8
Учебных часов на подготовку к экзамену/зачету/дифференцированному зачету (Контроль)	

Форма(ы) контроля:

Зачет 4 семестр

№ п/п	Тема и содержание	Форма изучения материалов: лекции, практические занятия, семинарские занятия, лабораторные работы, самостоятельная работа и трудоемкость (в часах)				Задания по самостоятельной работе студентов	Форма текущего контроля успеваемости (коллоквиумы, контрольные работы, компьютерные тесты и т.п.)
		ЛК	ПР/ СЕМ	ЛР	СР		
1	2	3	4	5	6	8	9
1.	Типы данных и операции: переменные, примитивные типы данных, арифметические операторы, выражения и присваивания			2		Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
2.	Представление данных в памяти: представление числовых (в прямом коде) и символьных данных. Представление отрицательных и вещественных чисел в памяти. Представление чисел в обратном и дополнительном коде. Поразрядные операции. Логические выражения. Тип Boolean. Операции отношения, логические и тернарные операции			2		Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
3.	Условные конструкции. Понятие блока, область действия блоков. Условные конструкции: if-else, switch. Итеративные конструкции while, do-while. Безусловные операторы перехода break. Итеративные конструкции for. Массивы. Вложенные циклы. Безусловные операторы перехода break с меткой, continue. Одномерные массивы, цикл for each в Java. Разбор примеров нахождения максимума и минимума, поиска на числовых массивах			2		Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
4.	Методы. Изучение понятия функций на примере методов Java. Передача параметров, возвращение результата. Видимость переменных. Указатели на примере языка Си: понятие, использование в качестве аргументов функций. Методы передачи аргументов в функцию. Динамическое выделение памяти. Многомерные массивы. Неровные массивы			2		Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
5.	Понятие класса и объекта. Цели и задачи ОО-подхода к			2		Проработка лекционного	Групповой и индивидуальный

	проектированию и разработке ПО. Объект, сообщение, класс, экземпляр объекта, метод. Общее понятие о парадигмах ООП: абстракция, инкапсуляция, наследование, полиморфизм. Описание протокола класса. Обзор классов, соответствующих примитивным типам				материала, литературных источников.	опрос, тесты, выполнение заданий.
6.	Работа с экземплярами класса Конструкторы и деструкторы. Статические методы. Открытые и закрытые поля. Доступ к полям объекта. Работа с экземплярами класса Инициализация переменных, массивов, полей классов в конструкторе. Конструктор по умолчанию. Перегрузка методов на примере конструкторов.			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
7.	Строки. Основы тестирования и отладки Начальные приемы тестирования и отладки, сценарии тестирования на примерах со строками			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
8.	Знакомство с Android разработкой ОС Android. Среда разработки. Принципиальная архитектура Android-приложения			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
9.	Интерфейс Android приложения Построение простейшего интерфейса пользователя. Язык разметки XML. Описание ресурсов Android с помощью XML. Разметки (Layouts) и их применение. Представления (Views)			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
10.	Наследование, инкапсуляция и полиморфизм Производные классы и наследование. Защищенные части классов и правила доступа для классов и объектов в Java. Сравнение иерархии классов и контейнеризации классов. Полиморфные методы и позднее связывание в Java. Абстрактные методы и классы, интерфейсы.			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
11.	Context и Intent Понятия контекста (Context) и намерения (Intent) в Android. Параметризованные типы (Generic) как пример статического			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.

	полиморфизма в Java.						
12.	Практикум ООП проектирования. Разбор кейсов проектирования архитектуры классов приложения. Диаграммы UML			2		Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
13.	Ввод -вывод в Java. Исключения Библиотечные классы ввода - вывода. Обработка исключений и классы исключений. Стандартные исключения Java. Работа с файлами в Android			2	1	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
14.	Внутренние и анонимные классы на примерах обработчиков событий пользовательского интерфейса			2	1	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
15.	Параллелизм и синхронизация. Процессы и потоки в Android. Классы AsyncTask и Thread. Реализация логики потоков. Синхронизация потоков			2	1	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
16.	Сервисы в Android. Жизненный цикл сервисов и управление им. Класс IntentService			2	0,8	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
	<b>Всего часов за 3 семестр:</b>			<b>32</b>	<b>3,8</b>		
1.	Фрагменты (Fragments). Сенсоры Создание и управление фрагментами. Класс Fragment и его методы. Взаимодействие фрагментов и активностей. Типы сенсоров и обработка событий			2		Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
2.	Двумерная графика в Android приложениях. Класс Canvas			2		Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
3.	Разработка игровых приложений. Этапы проектирования и реализации. Профессии в мире индустрии игр. Понятие игрового движка. Реализация графики на основе SurfaceView			2		Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
4.	Разработка 3D игр с использованием фреймворка libGDX <a href="https://habrahabr.ru/post/276139/">https://habrahabr.ru/post/276139/</a>			2		Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
5.	Массив, как базовая структура данных. Алгоритм двоичного			2		Проработка лекционного	Групповой и индивидуальный

	поиска: идея, применения, реализация. Классы Arrays (массивы), ArrayList (неограниченный массив) Список, как базовая структура данных: стеки, очереди, односвязные и двусвязные списки				материала, литературных источников.	опрос, тесты, выполнение заданий.
6.	Адаптеры в Андроид Назначение и применение. Стандартные адаптеры			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
7.	Реляционная модель данных и реляционные схемы. Представление данных в виде таблиц. Типы связей: один к одному, один ко многим, многие ко-многим. Проектирование простейшей БД. Необходимость возникновения и история развития СУБД. Обзор и классификация современных СУБД.			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
8.	Локальная СУБД на примере SQLite. Введение в SQL. Создание и наполнение таблиц: команды CREATE, INSERT. Команда SELECT для выборки данных, UPDATE для изменения, DELETE для удаления записей из таблицы. Дополнения к запросам SELECT. Ключевые слова ORDER BY, DISTINCT. Агрегация в SELECT в запросах: COUNT, SUM, AVG, MAX, MIN			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
9.	Рекурсия Линейная и ветвящаяся рекурсия. Стек вызовов Дерево, как базовая структура данных. Сбалансированные деревья. Двоичные деревья			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
10.	Обзор алгоритмов сортировок: пузырьковая, вставкой и быстрая в сравнении трудоемкости. Компараторы Хэш -таблица и функция хэширования Поддержка хэширования в Java, метод hashCode. Семейства контейнеров Collections и Map			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.
11.	Ассоциативные массивы. Класс Map, контейнеры HashMap, TreeMap. Хранение данных в Android Preferences Контент - провайдеры в Android. Назначение и применение. Стандартные контент -			2	Проработка лекционного материала, литературных источников.	Групповой и индивидуальный опрос, тесты, выполнение заданий.



	провайдеры ContactsContract, MediaStore					
12.	Введение в криптографию и криптоанализ. Простейшие алгоритмы подстановочного шифрования Введение в шифрование Понятие ключа шифрования. Симметричные и асимметричные ключи. Алгоритм DES. Алгоритм RSA. Примеры использования.			2		Проработка лекционного материала, литературных источников. Групповой и индивидуальный опрос, тесты, выполнение заданий.
13.	IP -сети Адресация в IP -сетях. IPv4. Автоматизация назначения IP -адресов (DHCP). Доменные имена (DNS), URL-ссылки. Несколько IP адресов для одного сайта. Популярные сетевые команды ping, tracer, ipconfig. Сервисы работы с IP -адресами. Web сервер. HTTP запросы и ответы Протокол HTTP. Понятие web - сервера. Запросы клиента POST и GET, коды ответов сервера, заголовки запросов и ответов. Разбор запросов и ответов сервера.			2	1	Проработка лекционного материала, литературных источников. Групповой и индивидуальный опрос, тесты, выполнение заданий.
14.	Клиент -серверная архитектура мобильных приложений: структура, схема взаимодействия сервера и клиента мобильного приложения. Формат JSON и XML. Сериализация. Библиотека Retrofit. Облачные платформы. REST взаимодействие Облачные сервисы для хостинга серверной части приложений. Реализация серверной части средствами Java, *PHP. Стиль взаимодействия REST. OAuth - авторизация.			2	1	Проработка лекционного материала, литературных источников. Групповой и индивидуальный опрос, тесты, выполнение заданий.
15.	Серверные СУБД Хранение данных на сервере в виде БД. Индексы. Знакомство с серверной СУБД на примере PostgreSQL/MySQL. Реализация back -end части web сервера средствами Java, PHP.			2	1	Проработка лекционного материала, литературных источников. Групповой и индивидуальный опрос, тесты, выполнение заданий.
16.	Дизайн программного обеспечения и приложений Material Design			2	0,8	Проработка лекционного материала, литературных источников. Групповой и индивидуальный опрос, тесты, выполнение заданий.
	<b>Всего часов за 4 семестр:</b>			<b>32</b>	<b>3,8</b>	