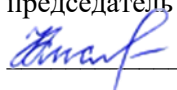


УТВЕРЖДАЮ  
 Проректор по УМР и КО,  
 председатель УМС СГСПУ  
 Н.Н. Кислова

## Объектно-ориентированное программирование

### рабочая программа дисциплины (модуля)

Закреплена за кафедрой	<b>Информатики, прикладной математики и методики их преподавания</b>	
Учебный план	ФМФИ-621ППио(4г).plx Направление подготовки 09.03.03 Прикладная информатика Направленность (профиль): «Прикладная информатика в государственном и муниципальном управлении»	
Квалификация	<b>бакалавр</b>	
Форма обучения	<b>очная</b>	
Общая трудоемкость	<b>23 ЗЕТ</b>	
Часов по учебному плану	828	Виды контроля в семестрах: экзамен 2,3,4
в том числе:		
аудиторные занятия	312	
самостоятельная работа	516	

#### Распределение часов дисциплины по семестрам

Семестр(Курс.Номер семестра на курсе)	2(1.2)		3(2.1)		4(2.2)		Итого	
	УП	РПД	УП	РПД	УП	РПД	УП	РПД
Вид занятий								
Лекции	30	30	40	40	40	40	110	110
Лабораторные занятия	58	58	72	72	72	72	202	202
В том числе инт.	18	18	22	22	22	22	62	62
Итого ауд.	88	88	112	112	112	112	312	312
Контактная работа	88	88	112	112	112	112	312	312
Сам. работа	164	164	176	176	176	176	516	516
Итого	252	252	288	288	288	288	828	828

Направление подготовки 09.03.03 Прикладная информатика, направленность (профиль): «Прикладная информатика в государственном и муниципальном управлении»

Рабочая программа дисциплины «Объектно-ориентированное программирование»

Программу составил(и):

**Пугач О.И.**

Рабочая программа дисциплины

**Объектно-ориентированное программирование**

разработана в соответствии с ФГОС ВО:

Федеральный государственный образовательный стандарт высшего образования по направлению подготовки 09.03.03 Прикладная информатика (уровень бакалавриата) (приказ Минобрнауки России от 19.09.2017 г. № 922)

составлена на основании учебного плана:

Направление подготовки 09.03.03 Прикладная информатика

Направленность (профиль): «Прикладная информатика в государственном и муниципальном управлении»

утвержденного учёным советом вуза от 31.08.2020 протокол № 1.

Рабочая программа одобрена на заседании кафедры

**Информатики, прикладной математики и методики их преподавания**

Протокол от 25.08.2020 г. № 1

Зав. кафедрой Добудько Т.В.

Начальник УОП



\_\_\_\_\_  
Н.А. Доманина

### 1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

**Цель изучения дисциплины:** изучение методов и приемов современного программирования и разработки приложений, формирование навыков разработки прототипов и законченных приложений с использованием современных языков объектно-ориентированного программирования.

**Задачи изучения дисциплины:**

- студент должен в полном объеме освоить концепции и методы промышленного программирования и особенности его реализации на различных языках (объектно-ориентированный подход, функциональный подход);
- знакомство студента с инструментальными средствами разработки ПО: системами контроля версий, отслеживания ошибок, инструментами автоматизированного тестирования;
- формирование навыков разработки несложных приложений на основе внешних баз данных.

**Область профессиональной деятельности:** (указывается в соответствии с ФГОС ВО)

06 Связь, информационные и коммуникационные технологии (в сфере проектирования, разработки, внедрения и эксплуатации информационных систем, управления их жизненным циклом);

### 2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Цикл (раздел) ОП:	Б1.В.12
<b>2.1 Требования к предварительной подготовке обучающегося:</b>	
Содержание дисциплины базируется на материале:	
Программирование	
<b>2.2 Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:</b>	
Интернет-технологии в государственном и муниципальном управлении	
Разработка приложений для мобильных устройств	
Учебная практика (технологическая (проектно-технологическая) практика)	

### 3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

#### ПК-4. Способность проводить тестирование компонентов программного обеспечения ИС

##### ПК-4.1 Знает основные методы и подходы к тестированию программ

Знает: понятия сплошного и выборочного ручного тестирования, средства автоматизированного тестирования Java (JUnit и др)

##### ПК-4.2 Умеет проводить модульное тестирование программного обеспечения ИС, интеграционное тестирование

Умеет: использовать средства автоматизированного тестирования Java (JUnit и др)

#### ПК-5. Способность осуществлять ведение базы данных и поддержку информационного обеспечения решения прикладных задач

##### ПК-5.1 Знает принципы, технологии и приемы организации баз данных, проектирования архитектуры информационных систем, нормативный и организационные аспекты управления доступа к данным

Знает: принципы, технологии и приемы организации баз данных для десктоп-приложений (СУБД MySQL), проектирования архитектуры информационных систем

##### ПК 5.2 Умеет проектировать архитектуру ИС различными инструментальными средствами

Умеет: проектировать и реализовать простую ИС средствами MySQL+Java

### 4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Интеракт.
	<b>Раздел 1. Объектно-ориентированное программирование на Java</b>			
1.1	Объекты и классы: реализация в языке Java /Лек/	2	2	2
1.2	Объекты и классы: реализация в языке Java /Лаб/	2	4	2
1.3	Объекты и классы: реализация в языке Java /Ср/	2	12	
1.4	Средства управления жизнью объекта. /Лек/	2	2	
1.5	Средства управления жизнью объекта. /Лаб/	2	4	
1.6	Средства управления жизнью объекта. /Ср/	2	12	
1.7	Работа с массивами объектов /Лек/	2	2	
1.8	Работа с массивами объектов /Лаб/	2	4	
1.9	Работа с массивами объектов /Ср/	2	12	
1.10	Наследование в Java. /Лек/	2	2	
1.11	Наследование в Java. /Лаб/	2	4	
1.12	Наследование в Java. /Ср/	2	12	
1.13	Абстрактные классы и интерфейсы /Лек/	2	2	
1.14	Абстрактные классы и интерфейсы /Лаб/	2	4	
1.15	Абстрактные классы и интерфейсы /Ср/	2	12	
1.16	Пакеты. /Лек/	2	2	

Рабочая программа дисциплины «Объектно-ориентированное программирование»

1.17	Пакеты. /Лаб/	2	4	
1.18	Пакеты. /Лаб/	2	12	
1.19	Подсистема ввода вывода java.io. /Лек/	2	2	
1.20	Подсистема ввода вывода java.io. /Лаб/	2	4	
1.21	Подсистема ввода вывода java.io. /Ср/	2	12	
1.22	Обработка исключений. /Лек/	2	2	
1.23	Обработка исключений. /Лаб/	2	4	
1.24	Обработка исключений. /Ср/	2	12	
1.25	Классы Throwable и Exception: создание и использование. /Лек/	2	2	
1.26	Классы Throwable и Exception: создание и использование. /Лаб/	2	4	
1.27	Классы Throwable и Exception: создание и использование. /Ср/	2	12	
1.28	Обобщенное программирование (Java Generics). /Лек/	2	2	
1.29	Обобщенное программирование (Java Generics). /Лаб/	2	4	
1.30	Обобщенное программирование (Java Generics). /Ср/	2	12	
1.31	Графическая подсистема Java. Классы AWT. /Лек/	2	2	
1.32	Графическая подсистема Java. Классы AWT. /Лаб/	2	4	
1.33	Графическая подсистема Java. Классы AWT. /Ср/	2	12	
1.34	Обработка событий в графической подсистеме (Listeners). /Лек/	2	2	
1.35	Обработка событий в графической подсистеме (Listeners). /Лаб/	2	4	
1.36	Обработка событий в графической подсистеме (Listeners). /Ср/	2	12	
1.37	Основные возможности графической системы Swing. /Лек/	2	8	4
1.38	Основные возможности графической системы Swing. /Лаб/	2	10	10
1.39	Основные возможности графической системы Swing. /Ср/	2	20	
	<b>Раздел 2. Программы на Java в архитектуре клиент-сервер</b>			
2.1	Многопоточное программирование на Java. /Лек/	3	4	4
2.2	Многопоточное программирование на Java. /Лаб/	3	4	4
2.3	Многопоточное программирование на Java. /Ср/	3	14	
2.4	Класс Thread и интерфейс Runnable. /Лек/	3	2	
2.5	Класс Thread и интерфейс Runnable. /Лаб/	3	6	
2.6	Класс Thread и интерфейс Runnable. /Ср/	3	14	
2.7	Особенности написания многопоточных программ. /Лек/	3	4	4
2.8	Особенности написания многопоточных программ. /Лаб/	3	4	4
2.9	Особенности написания многопоточных программ. /Ср/	3	14	
2.10	Организация пакетов стандартной библиотеки Java. /Лек/	3	2	
2.11	Организация пакетов стандартной библиотеки Java. /Лаб/	3	6	6
2.12	Организация пакетов стандартной библиотеки Java. /Ср/	3	14	
2.13	Класс String и производительность в Java приложениях. /Лек/	3	4	
2.14	Класс String и производительность в Java приложениях. /Лаб/	3	4	
2.15	Класс String и производительность в Java приложениях. /Ср/	3	14	
2.16	Перечисления Java(java.lang.Enum). /Лек/	3	2	
2.17	Перечисления Java(java.lang.Enum). /Лаб/	3	6	
2.18	Перечисления Java(java.lang.Enum). /Ср/	3	14	
2.19	Контейнеры и коллекции в Java. /Лек/	3	4	
2.20	Контейнеры и коллекции в Java. /Лаб/	3	4	
2.21	Контейнеры и коллекции в Java. /Ср/	3	14	
2.22	Итераторы /Лек/	3	2	
2.23	Итераторы /Лаб/	3	6	
2.24	Итераторы /Ср/	3	14	
2.25	Технология коллективной разработки Java приложений. /Лек/	3	4	
2.26	Технология коллективной разработки Java приложений. /Лаб/	3	4	
2.27	Технология коллективной разработки Java приложений. /Ср/	3	14	
2.28	Автоматизация сборки и размещения Java приложений /Лек/	3	2	
2.29	Автоматизация сборки и размещения Java приложений /Лаб/	3	6	
2.30	Автоматизация сборки и размещения Java приложений /Ср/	3	14	
2.31	Разработка прототипа и приложения (проект). /Лек/	3	4	
2.32	Разработка прототипа и приложения (проект). /Лаб/	3	4	
2.33	Разработка прототипа и приложения (проект). /Ср/	3	14	
2.34	Взаимодействие с базой данных /Лек/	3	2	
2.35	Взаимодействие с базой данных /Лаб/	3	6	
2.36	Взаимодействие с базой данных /Ср/	3	14	
2.37	Оптимизация приложений /Лек/	3	4	
2.38	Оптимизация приложений /Лаб/	3	12	
2.39	Оптимизация приложений /Ср/	3	8	
	<b>Раздел 3. Специальные инструменты и методы современного</b>			

Рабочая программа дисциплины «Объектно-ориентированное программирование»

	<b>программирования</b>			
3.1	Системное программирование и язык C/C++. Сравнительный анализ с Java. Вопросы эффективности и безопасности. /Лек/	4	10	2
3.2	Системное программирование и язык C/C++. Сравнительный анализ с Java. Вопросы эффективности и безопасности. /Лаб зан/	4	18	4
3.3	Системное программирование и язык C/C++. Сравнительный анализ с Java. Вопросы эффективности и безопасности. /Ср/	4	44	
3.4	Основные конструкции и типы данных в C++. Массивы. Синтаксические особенности, стандарты, совместимость. Часто используемые библиотеки и функции. /Лек/	4	10	2
3.5	Основные конструкции и типы данных в C++. Массивы. Синтаксические особенности, стандарты, совместимость. Часто используемые библиотеки и функции. /Лаб зан/	4	18	4
3.6	Основные конструкции и типы данных в C++. Массивы. Синтаксические особенности, стандарты, совместимость. Часто используемые библиотеки и функции. /Ср/	4	44	
3.7	Объектно-ориентированное программирование в C++. Коллекции. Потoki ввода-вывода. Работа с сетью. Низкоуровневое программирование /Лек/	4	10	2
3.8	Объектно-ориентированное программирование в C++. Коллекции. Потoki ввода-вывода. Работа с сетью. Низкоуровневое программирование /Лаб зан/	4	18	4
3.9	Объектно-ориентированное программирование в C++. Коллекции. Потoki ввода-вывода. Работа с сетью. Низкоуровневое программирование /Ср/	4	44	
3.10	Python как специализированный язык. Основной синтаксис. Области применения. Библиотеки для анализа данных, распознавания речи и образов. /Лек/	4	10	2
3.11	Python как специализированный язык. Основной синтаксис. Области применения. Библиотеки для анализа данных, распознавания речи и образов. /Лаб зан/	4	18	2
3.12	Python как специализированный язык. Основной синтаксис. Области применения. Библиотеки для анализа данных, распознавания речи и образов. /Ср/	4	44	

**5. Оценочные и методические материалы по дисциплине (модулю)**

**5.1. Содержание аудиторной работы по дисциплине (модулю)**

№	Тема лекции	Содержание	Литература
<b>Раздел 1 Объектно-ориентированное программирование на Java</b>			
1.1	Объекты и классы: реализация в языке Java	Объекты и классы. Абстракция данных. Сценарий построения объектно-ориентированной программы. Члены классов. Методы и поля. Специальные методы классов (конструкторы). Конструктор по умолчанию. Модификаторы уровня доступа (default, public, protected, private). Статические методы классов. Методы функционального и процедурного типа. Процедурное программирование. Объектно ориентированное программирование. Сравнение парадигм. Сигнатура метода. Перегрузка методов. Инкапсуляция. Наследование. Управление наследованием. Интерфейсы как средство реализации множественного наследования. Полиморфизм. Средства реализации полиморфизма.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.2	Средства управления жизнью объекта.	Представление объектов и классов. Структура объявления класса. Доступ к членам класса. Спецификаторы доступа для обеспечения инкапсуляции. Знакомство с final. Средства управления жизнью объекта. Конструкторы и метод finalize(). Принципы работы сборщика мусора.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.3	Работа с массивами объектов.	Работа с массивами объектов. Статические поля и методы классов. Принцип работы ClassLoader. Реализация отношений между объектами и классами. Использование и зависимость от интерфейсов. Объекты при передаче параметров и возврате из методов. Внутренние классы.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.4	Наследование в Java.	Средства объектно-ориентированного программирования Java. Наследование в Java. Производные классы. Класс Object. Метод toString(). Управление доступом в производных классах.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.5	Абстрактные классы и	Абстрактные классы и интерфейсы. Реализация интерфейсов	<a href="https://docs.oracle.com/java">https://docs.oracle.com/java</a>

Рабочая программа дисциплины «Объектно-ориентированное программирование»

	интерфейсы.	как альтернатива множественному наследованию. Информация о типе на этапе исполнения. Оператор instanceof. Приведение типов.	se/tutorial/ <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.6	Пакеты.	Пакеты. Определение пакета. Ограничение доступа. Импортирование пакетов. Разрешение конфликтов имен.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.7	Подсистема ввода вывода java.io.	Подсистема ввода вывода java.io. Общие концепции организации ввода –вывода в библиотеке Java. Проблема платформенной независимости и локализации. Основные классы потоков ввода-вывода в Java и методы работы с ними. Использование потоков ввода вывода при работе с файлами. Эффективность.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.8	Обработка исключений.	Обработка исключений. Основные принципы и типы исключительных ситуаций. Перехват исключительных ситуаций. Операторы try, throw, throws, catch, finally. Использование нескольких блоков catch и вложенный оператор try. Не перехваченные исключительные	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.9	Классы Throwable и Exception: создание и использование.	Встроенные исключительные исключения Java. Классы Throwable и Exception. Принципы создания и использования исключительных ситуаций.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.10	Обобщенное программирование (Java Generics).	Родовые компоненты и обобщенное программирование. Java Generics. Ограничения на параметры. Совместимость на уровне байт-кода.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.11	Графическая подсистема Java. Классы AWT.	Графическая подсистема Java. Классы AWT. Основы работы с окнами. Component, Container, Panel, Window, Frame, Canvas.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.12	Обработка событий в графической подсистеме (Listeners).	Доставка и обработка событий в графической подсистеме. Механизм Listeners. Создание программы с оконным интерфейсом. Рисование графических примитивов. Использование управляющих элементов, диспетчеров компоновки и меню. Связь классов AWT с оконным интерфейсом операционной системы.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.13	Основные возможности графической системы Swing.	Классы Swing. Основные принципы графической системы Swing. Платформенная независимость, понятие Look&Feel. Написание графического интерфейса с использованием Swing компонентов. Введение в технологии разработки программного обеспечения.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
<b>Раздел 2 Программы на Java в архитектуре клиент-сервер</b>			
2.1	Многопоточное программирование на Java.	Многопоточное программирование на Java. Параллелизм. Модель потока в Java. Зависимость от реализации потока в операционной системе.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.2	Класс Thread и интерфейс Runnable.	Свойства потока. Синхронизация. Передача сообщений. Класс Thread и интерфейс Runnable. Главный поток и способы создания потоков. Управление потоками и приоритеты потоков. Группы потоков.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.3	Особенности написания многопоточных программ.	Особенности написания многопоточных программ. Использование синхронизирующих блоков и мониторов объектов. Синхронизированные методы объектов. Взаимная блокировка. Использование пула потоков.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.4	Организация пакетов стандартной библиотеки Java.	Стандартная библиотека Java. Организация пакетов стандартной библиотеки Java. Пакеты java и javax.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.5	Класс String и производительность Java приложениях.	Использование и методы класса String. Класс String – краеугольный камень производительности в Java приложениях. Использование StringBuffer.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.6	Перечисления Java(java.lang.Enum).	Перечисления Java(java.lang.Enum). Особенности классов перечислений. Члены перечисления, поведение. Использование перечислений.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>

Рабочая программа дисциплины «Объектно-ориентированное программирование»

			va/javase/11/
2.7	Контейнеры и коллекции в Java.	Общие принципы организации контейнеров и коллекций в Java. Использование множеств и списков.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.8	Итераторы	Использование отображений и ассоциативных контейнеров. Итераторы и исключительные ситуации при работе с классами утилит.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.9	Технология коллективной разработки Java приложений.	Технология коллективной разработки Java приложений. Использование документирующих комментариев. Соглашения при написании кода и именовании классов и объектов.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.10	Автоматизация сборки и размещения Java приложений.	Автоматизация сборки и размещения Java приложений. Магазины.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.11	Разработка прототипа и приложения (проект).	Разработка прототипа и приложения (проект). Выбор инструментальных средств разработки. Использование git (системы версионного контроля) и баг-трекера. Документирование.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.12	Взаимодействие с базой данных.	Взаимодействие с базой данных. JSON. Транзакции. Виды запросов. Подключение базы данных в среде NetBeans. Разработка элементов информационной системы.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.13	Оптимизация приложений.	Оптимизация, общие понятия. Оптимизация по времени и по памяти. Алгоритмы и подходы.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
<b>Раздел 3. Специальные инструменты и методы современного программирования</b>			
3.1	Системное программирование и язык C/C++.	Системное программирование и язык C/C++. Сравнительный анализ с Java. Вопросы эффективности и безопасности.	Л1.1, Л1.2, Л1.3
3.2	Основные конструкции и типы данных в C++.	Основные конструкции и типы данных в C++. Массивы. Синтаксические особенности, стандарты, совместимость. Часто используемые библиотеки и функции.	Л1.1, Л1.2, Л1.3
3.3	Объектно-ориентированное программирование в C++.	Объектно-ориентированное программирование в C++. Коллекции. Потоки ввода-вывода. Работа с сетью. Низкоуровневое программирование /Лек/	Л1.1, Л1.2, Л1.3
3.4	Python как специализированный язык	Python как специализированный язык. Основной синтаксис. Области применения. Библиотеки для анализа данных, распознавания речи и образов. /Лек/	Л1.1, Л1.2, Л1.3

№	Название лабораторной	Вопросы	Литература
<b>Раздел 1. Объектно-ориентированное программирование на Java</b>			
1.1	Объекты и классы: реализация в языке Java	Объекты и классы. Абстракция данных. Сценарий построения объектно-ориентированной программы. Члены классов. Методы и поля. Специальные методы классов (конструкторы). Конструктор по умолчанию. Модификаторы уровня доступа (default, public, protected, private). Статические методы классов. Методы функционального и процедурного типа. Процедурное программирование. Объектно-ориентированное программирование. Сравнение парадигм. Сигнатура метода. Перегрузка методов. Инкапсуляция. Наследование. Управление наследованием. Интерфейсы как средство реализации множественного наследования. Полиморфизм. Средства реализации полиморфизма.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.2	Средства управления жизнью объекта.	Представление объектов и классов. Структура объявления класса. Доступ к членам класса. Спецификаторы доступа для обеспечения инкапсуляции. Знакомство с final. Средства управления жизнью объекта. Конструкторы и метод finalize(). Принципы работы сборщика мусора.	<a href="https://docs.oracle.com/javase/tutorial/">https://docs.oracle.com/javase/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.3	Работа с массивами объектов.	Работа с массивами объектов. Статические поля и методы	<a href="https://docs.oracle.com/java">https://docs.oracle.com/java</a>

Рабочая программа дисциплины «Объектно-ориентированное программирование»

		классов. Принцип работы ClassLoader. Реализация отношений между объектами и классами. Использование и зависимость от интерфейсов. Объекты при передаче параметров и возврате из методов. Внутренние классы.	se/tutorial/ <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.4	Наследование в Java.	Средства объектно-ориентированного программирования Java. Наследование в Java. Производные классы. Класс Object. Метод toString(). Управление доступом в производных классах.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.5	Абстрактные классы и интерфейсы.	Абстрактные классы и интерфейсы. Реализация интерфейсов как альтернатива множественному наследованию. Информация о типе на этапе исполнения. Оператор instanceof. Приведение типов.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.6	Пакеты.	Пакеты. Определение пакета. Ограничение доступа. Импортирование пакетов. Разрешение конфликтов имен.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.7	Подсистема ввода вывода java.io.	Подсистема ввода вывода java.io. Общие концепции организации ввода –вывода в библиотеке Java. Проблемы платформенной независимости и локализации. Основные классы потоков ввода-вывода в Java и методы работы с ними. Использование потоков ввода вывода при работе с файлами. Эффективность.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.8	Обработка исключений.	Обработка исключений. Основные принципы и типы исключительных ситуаций. Перехват исключительных ситуаций. Операторы try, throw, throws, catch, finally. Использование нескольких блоков catch и вложенный оператор try. Не перехваченные исключительные	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.9	Классы Throwable и Exception: создание и использование.	Встроенные исключительные исключения Java. Классы Throwable и Exception. Принципы создания и использования исключительных ситуаций.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.10	Обобщенное программирование (Java Generics).	Родовые компоненты и обобщенное программирование. Java Generics. Ограничения на параметры. Совместимость на уровне байт-кода.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.11	Графическая подсистема Java. Классы AWT.	Графическая подсистема Java. Классы AWT. Основы работы с окнами. Component, Container, Panel, Window, Frame, Canvas.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.12	Обработка событий в графической подсистеме (Listeners).	Доставка и обработка событий в графической подсистеме. Механизм Listeners. Создание программы с оконным интерфейсом. Рисование графических примитивов. Использование управляющих элементов, диспетчеров компоновки и меню. Связь классов AWT с оконным интерфейсом операционной системы.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
1.13	Основные возможности графической системы Swing.	Классы Swing. Основные принципы графической системы Swing. Платформенная независимость, понятие Look&Feel. Написание графического интерфейса с использованием Swing компонентов. Введение в технологии разработки программного обеспечения.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>

**Раздел 2 Программы на Java в архитектуре клиент-сервер**

2.1	Многопоточное программирование на Java.	Многопоточное программирование на Java. Параллелизм. Модель потока в Java. Зависимость от реализации потока в операционной системе.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.2	Класс Thread и интерфейс Runnable.	Свойства потока. Синхронизация. Передача сообщений. Класс Thread и интерфейс Runnable. Главный поток и способы создания потоков. Управление потоками и приоритеты потоков. Группы потоков.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.3	Особенности написания многопоточных программ.	Особенности написания многопоточных программ. Использование синхронизирующих блоков и мониторов объектов. Синхронизированные методы объектов. Взаимная блокировка. Использование пула потоков.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.4	Организация пакетов стандартной библиотеки	Стандартная библиотека Java. Организация пакетов стандартной библиотеки Java. Пакеты java и javax.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a>



Рабочая программа дисциплины «Объектно-ориентированное программирование»

	Java.		<a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.5	Класс String и производительность в Java приложениях.	Использование и методы класса String. Класс String – краеугольный камень производительности в Java приложениях. Использование StringBuffer.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.6	Перечисления Java(java.lang.Enum).	Перечисления Java(java.lang.Enum). Особенности классов перечислений. Члены перечисления, поведение. Использование перечислений.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.7	Контейнеры и коллекции в Java.	Общие принципы организации контейнеров и коллекций в Java. Использование множеств и списков.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.8	Итераторы	Использование отображений и ассоциативных контейнеров. Итераторы и исключительные ситуации при работе с классами утилит.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.9	Технология коллективной разработки Java приложений.	Технология коллективной разработки Java приложений. Использование документирующих комментариев. Соглашения при написании кода и именовании классов и объектов.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.10	Автоматизация сборки и размещения Java приложений.	Автоматизация сборки и размещения Java приложений. Магазины.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.11	Разработка прототипа приложения (проект).	Разработка прототипа и приложения (проект). Выбор инструментальных средств разработки. Использование git (системы версионного контроля) и баг-трекера. Документирование.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.12	Взаимодействие с базой данных.	Взаимодействие с базой данных. JSON. Транзакции. Виды запросов. Подключение базы данных в среде NetBeans. Разработка элементов информационной системы.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>
2.13	Оптимизация приложений.	Оптимизация, общие понятия. Оптимизация по времени и по памяти. Алгоритмы и подходы.	<a href="https://docs.oracle.com/java/se/tutorial/">https://docs.oracle.com/java/se/tutorial/</a> <a href="https://docs.oracle.com/en/java/javase/11/">https://docs.oracle.com/en/java/javase/11/</a>

**Раздел 3. Специальные инструменты и методы современного программирования**

3.1	Системное программирование и язык C/C++.	Системное программирование и язык C/C++. Сравнительный анализ с Java. Вопросы эффективности и безопасности.	Л1.1, Л1.2, Л1.3
3.2	Основные конструкции и типы данных в C++.	Основные конструкции и типы данных в C++. Массивы. Синтаксические особенности, стандарты, совместимость. Часто используемые библиотеки и функции.	Л1.1, Л1.2, Л1.3
3.3	Объектно-ориентированное программирование в C++.	Объектно-ориентированное программирование в C++. Коллекции. Потоки ввода-вывода. Работа с сетью. Низкоуровневое программирование /Лек/	Л1.1, Л1.2, Л1.3
3.4	Python как специализированный язык	Python как специализированный язык. Основной синтаксис. Области применения. Библиотеки для анализа данных, распознавания речи и образов.	Л1.1, Л1.2, Л1.3

**5.2. Содержание самостоятельной работы по дисциплине (модулю)**

**Содержание обязательной самостоятельной работы по дисциплине**

№ п/п	Разделы дисциплины	Содержание самостоятельной работы студентов	Продукты деятельности
1	Объектно-ориентированное программирование на Java	Индивидуальное домашнее задание	Отчет (работающее приложение, UML, документация)
2	Программы на Java в архитектуре клиент-сервер	Индивидуальное домашнее задание	Отчет (работающее приложение, UML, документация)
3	Специальные инструменты и методы современного программирования	Индивидуальное домашнее задание	Отчет (работающее приложение, UML, документация)

**Содержание самостоятельной работы по дисциплине на выбор студента**

№ п/п	Темы дисциплины	Содержание самостоятельной работы студентов	Продукты деятельности
1	Объектно-ориентированное программирование на Java	Решение задач повышенной сложности	Программа
2	Программы на Java в архитектуре клиент-сервер	Решение задач повышенной сложности	Программа
3	Специальные инструменты и методы современного программирования	Решение задач повышенной сложности	Программа

**5.3. Образовательные технологии**

При организации изучения дисциплины будут использованы следующие образовательные технологии: информационно-коммуникационные технологии, технология организации самостоятельной работы, технология рефлексивного обучения, технология модульного обучения, технология игрового обучения, технологии групповой дискуссии, интерактивные технологии, технология проблемного обучения, технология организации учебно-исследовательской деятельности, технология проектного обучения, технология развития критического мышления.

**5.4. Текущий контроль, промежуточный контроль и промежуточная аттестация**

Балльно-рейтинговая карта дисциплины оформлена как приложение к рабочей программе дисциплины. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине оформлен отдельным документом.

**6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ**

**6.1. Рекомендуемая литература**

**6.1.1. Основная литература**

	Авторы, составители	Заглавие, ссылка на электронную библиотечную систему	Издательство, год
Л1.1	Николаев Е.И.	Объектно-ориентированное программирование: учебное пособие <a href="http://biblioclub.ru/index.php?page=book&amp;id=458133">http://biblioclub.ru/index.php?page=book&amp;id=458133</a>	Ставрополь: СКФУ, 2015
Л1.2	Комлева Н.В.	Методы программирования: Учебно-методический комплекс <a href="http://biblioclub.ru/index.php?page=book&amp;id=90390">http://biblioclub.ru/index.php?page=book&amp;id=90390</a>	М.: Евразийский открытый институт, 2011
Л1.3	Москвитин, А.А.	Решение задач на компьютерах: учебное пособие <a href="http://biblioclub.ru/index.php?page=book&amp;id=273667">http://biblioclub.ru/index.php?page=book&amp;id=273667</a>	М.; Берлин: Директ-Медиа, 2015

**6.1.2. Дополнительная литература**

	Авторы, составители	Заглавие, ссылка на электронную библиотечную систему	Издательство, год
Л2.1	Николаев Е.И.	Объектно-ориентированное программирование: лабораторный практикум <a href="http://biblioclub.ru/index.php?page=book&amp;id=458134">http://biblioclub.ru/index.php?page=book&amp;id=458134</a> .	Ставрополь: СКФУ, 2015, Ч.1
Л2.2	Николаев Е.И.	Объектно-ориентированное программирование : лабораторный практикум : в 2 ч. / <a href="http://biblioclub.ru/index.php?page=book&amp;id=458135">http://biblioclub.ru/index.php?page=book&amp;id=458135</a>	Ставрополь: СКФУ, 2015. - Ч. 2

**6.2 Перечень программного обеспечения**

- Acrobat Reader DC
- Dr. Web Desktop Security Suite, Dr. Web Server Security Suite
- GIMP
- Microsoft Office 365 Pro Plus - subscription license (12 month) (Пакет программ Word, Excel, Access, PowerPoint, Outlook, OneNote, Publisher, Teams, OneDrive, Yammer, Stream, SharePoint Online).
- Microsoft Windows 10 Education
- XnView
- Архиватор 7-Zip
- 1С:ИТС ПРОФ ВУЗ
- Программная система для обнаружения текстовых заимствований в учебных и научных работах «Антиплагиат.ВУЗ»

**6.3 Перечень информационных справочных систем**

- Elsevier (база данных «Freedom Collection» и коллекции электронных книг «Freedom Collection eBook collection», национальная подписка на полнотекстовые ресурсы)
- SCOPUS издательства Elsevier
- SpringerNature (национальная подписка на полнотекстовые ресурсы)
- БД «Polpred.com. Обзор СМИ»
- УИС РОССИЯ
- ЭБС «E-LIBRARY.RU»

- ЭБС «ЛАНЬ»
- ЭБС «РУКОНТ» (Контекстум)
- ЭБС «Университетская библиотека онлайн»
- ЭБС «ЮРАЙТ» (Коллекция Легендарные книги)
- ЭБС «IPR BOOKS»

#### 7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1	Наименование специального помещения: учебная аудитория для проведения занятий лекционного типа, практических занятий, групповых консультаций, индивидуальных консультаций, текущего контроля, промежуточной аттестации, Учебная аудитория. Оснащенность: Меловая доска-1шт., Комплект учебной мебели
7.2	Наименование специального помещения: помещение для самостоятельной работы, Читальный. Оснащенность: ПК -4шт., Письменный стол-4 шт., Парта-2 шт.

#### 8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

Работа над теоретическим материалом происходит кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю.

Проработка рабочей программы дисциплины, уделяя особое внимание целям и задачам, структуре и содержанию дисциплины. Конспектирование источников, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы, работа с информационными источниками в разных форматах.

Также в процессе изучения дисциплины методические рекомендации могут быть изданы отдельным документом.

Балльно-рейтинговая карта дисциплины «Объектно-ориентированное программирование»

Курс 1 Семестр 2

Вид контроля		Минимальное количество баллов	Максимальное количество баллов
<b>Раздел 1. Объектно-ориентированное программирование на Java</b>			
Текущий контроль по разделу:			
1	Аудиторная работа	13	26
2	Самостоятельная работа (специальные обязательные формы)	5	10
3	Самостоятельная работа (специальные формы на выбор студента)	2	4
Контрольное мероприятие по разделу			
Промежуточный контроль		20	40
Промежуточная аттестация		36	60
Итого:		<b>56</b>	<b>100</b>

Виды контроля		Перечень или примеры заданий, критерии оценки и количество баллов	Темы для изучения и образовательные результаты
<b>Текущий контроль по разделу «Объектно-ориентированное программирование на Java»</b>			
1	Аудиторная работа	Лабораторная работа №1 «Объекты и классы» Записать объект, моделирующий свойства и методы сотрудника организации. Составить UML-диаграмму. Решение задач Критерии оценивания: • не решал задачи или решил неправильно – 0 баллов; • задачи решены с несущественными ошибками – 1 балл; • задачи решены без ошибок – 2 балла. Итого – 13x2=26 баллов	Темы: Объекты и классы: реализация в языке Java Средства управления жизнью объекта Работа с массивами объектов Наследование в Java Абстрактные классы и интерфейсы Пакеты Подсистема ввода вывода java.io Обработка исключений Классы Throwable и Exception: создание и использование Обобщенное программирование (Java Generics) Графическая подсистема Java. Классы AWT. Обработка событий в графической подсистеме (Listeners). Основные возможности графической системы Swing Образовательные результаты Знает: понятия сплошного и выборочного ручного тестирования, средства автоматизированного тестирования Java (JUnit и др) Умеет: использовать средства автоматизированного тестирования Java (JUnit и др)
2	Самостоятельная работа (обязательные формы)	Индивидуальное домашнее задание (ИДЗ) (x2) • решены все задачи ИДЗ – 3 балла; • решения задач с иллюстрациями оформлены развернуто, в соответствии с требованиями преподавателя – 1 балл; • отчет представлен преподавателю (загружен на проверку в систему управления обучением) в установленные сроки – 1 балл.	Темы: Объекты и классы: реализация в языке Java Средства управления жизнью объекта Работа с массивами объектов Наследование в Java Абстрактные классы и интерфейсы

		Итого – 5x2=10 баллов	<p>Пакеты</p> <p>Подсистема ввода вывода java.io</p> <p>Обработка исключений</p> <p>Классы Throwable и Exception: создание и использование</p> <p>Обобщенное программирование (Java Generics)</p> <p>Графическая подсистема Java. Классы AWT.</p> <p>Обработка событий в графической подсистеме (Listeners).</p> <p>Основные возможности графической системы Swing</p> <p>Образовательные результаты</p> <p>Знает: понятия сплошного и выборочного ручного тестирования, средства автоматизированного тестирования Java (JUnit и др)</p> <p>Умеет: использовать средства автоматизированного тестирования Java (JUnit и др)</p>
3	Самостоятельная работа (на выбор студента)	<p>Решение задач повышенной сложности.</p> <ul style="list-style-type: none"> <li>решены все задачи – 3 балла;</li> <li>решения задач с иллюстрациями оформлены развернуто, в соответствии с требованиями преподавателя – 1 балл.</li> </ul> <p>Итого – 4 балла</p>	<p>Темы:</p> <p>Объекты и классы: реализация в языке Java</p> <p>Средства управления жизнью объекта</p> <p>Работа с массивами объектов</p> <p>Наследование в Java</p> <p>Абстрактные классы и интерфейсы</p> <p>Пакеты</p> <p>Подсистема ввода вывода java.io</p> <p>Обработка исключений</p> <p>Классы Throwable и Exception: создание и использование</p> <p>Обобщенное программирование (Java Generics)</p> <p>Графическая подсистема Java. Классы AWT.</p> <p>Обработка событий в графической подсистеме (Listeners).</p> <p>Основные возможности графической системы Swing</p> <p>Образовательные результаты</p> <p>Знает: понятия сплошного и выборочного ручного тестирования, средства автоматизированного тестирования Java (JUnit и др)</p> <p>Умеет: использовать средства автоматизированного тестирования Java (JUnit и др)</p>
Контрольное мероприятие по разделу			
Промежуточный контроль (количество баллов)		Минимальное количество баллов – 20, максимальное – 40	
Промежуточная аттестация		Представлены в фонде оценочных средств для промежуточной аттестации по дисциплине	

Курс 2 Семестр 3

Вид контроля		Минимальное количество баллов	Максимальное количество баллов
<b>Раздел 2. Программы на Java в архитектуре клиент-сервер</b>			
Текущий контроль по разделу:			
1	Аудиторная работа	13	26
2	Самостоятельная работа (специальные обязательные формы)	5	10
3	Самостоятельная работа (специальные формы на выбор студента)	2	4
Контрольное мероприятие по разделу			
Промежуточный контроль		20	40
Промежуточная аттестация		36	60
<b>Итого:</b>		<b>56</b>	<b>100</b>

Виды контроля		Перечень или примеры заданий, критерии оценки и количество баллов	Темы для изучения и образовательные результаты
<b>Текущий контроль по разделу «Программы на Java в архитектуре клиент-сервер»</b>			
1	Аудиторная работа	<p>Практическое занятие №6 «Класс String и производительность в Java приложениях.»</p> <p>Вывести гласные слова S, встречающихся более одного раза. Решить задачу с и без использования StringBilder.</p> <p>Решение задач</p> <p>Критерии оценивания:</p> <ul style="list-style-type: none"> <li>• не решал задачи или решил неправильно – 0 баллов;</li> <li>• задачи решены с несущественными ошибками – 1 балл;</li> <li>• задачи решены без ошибок – 2 балла.</li> </ul> <p>Итого – 13x2=26 баллов</p>	<p>Темы:</p> <p>Многопоточное программирование на Java</p> <p>Класс Thread и интерфейс Runnable</p> <p>Особенности написания многопоточных программ</p> <p>Организация пакетов стандартной библиотеки Java</p> <p>Класс String и производительность в Java приложениях</p> <p>Перечисления Java(java.lang.Enum)</p> <p>Контейнеры и коллекции в Java</p> <p>Итераторы</p> <p>Технология коллективной разработки Java приложений</p> <p>Автоматизация сборки и размещения Java приложений</p> <p>Разработка прототипа и приложения (проект)</p> <p>Взаимодействие с базой данных</p> <p>Оптимизация приложений</p> <p>Образовательные результаты</p> <p>Знает: принципы, технологии и приемы организации баз данных для десктоп-приложений (СУБД MySQL), проектирования архитектуры информационных систем,</p> <p>Умеет: проектировать и реализовать простую ИС средствами MySQL+Java</p>
2	Самостоятельная работа (обязательные формы)	<p>Индивидуальное домашнее задание (ИДЗ) (x2)</p> <ul style="list-style-type: none"> <li>• решены все задачи ИДЗ – 3 балла;</li> <li>• решения задач с иллюстрациями оформлены развернуто, в соответствии с требованиями преподавателя – 1 балл;</li> <li>• отчет представлен преподавателю (загружен на проверку в систему управления обучением) в установленные сроки – 1 балл.</li> </ul>	<p>Темы:</p> <p>Многопоточное программирование на Java</p> <p>Класс Thread и интерфейс Runnable</p> <p>Особенности написания многопоточных программ</p> <p>Организация пакетов стандартной библиотеки Java</p> <p>Класс String и производительность в Java приложениях</p>

		Итого – 5x2=10 баллов	<p>Перечисления Java(java.lang.Enum)                  Контейнеры и коллекции в Java                  Итераторы                  Технология коллективной разработки Java приложений                  Автоматизация сборки и размещения Java приложений                  Разработка прототипа и приложения (проект)                  Взаимодействие с базой данных                  Оптимизация приложений                  Образовательные результаты                  Знает: принципы, технологии и приемы организации баз данных для десктоп-приложений (СУБД MySQL), проектирования архитектуры информационных систем,                  Умеет: проектировать и реализовать простую ИС средствами MySQL+Java</p>
3	Самостоятельная работа (на выбор студента)	<p>Решение задач повышенной сложности.</p> <ul style="list-style-type: none"> <li>решены все задачи – 3 балла;</li> <li>решения задач с иллюстрациями оформлены развернуто, в соответствии с требованиями преподавателя – 1 балл.</li> </ul> <p>Итого – 4 балла</p>	<p>Темы:                  Многопоточное программирование на Java                  Класс Thread и интерфейс Runnable                  Особенности написания многопоточных программ                  Организация пакетов стандартной библиотеки Java                  Класс String и производительность в Java приложениях                  Перечисления Java(java.lang.Enum)                  Контейнеры и коллекции в Java                  Итераторы                  Технология коллективной разработки Java приложений                  Автоматизация сборки и размещения Java приложений                  Разработка прототипа и приложения (проект)                  Взаимодействие с базой данных                  Оптимизация приложений                  Образовательные результаты                  Знает: принципы, технологии и приемы организации баз данных для десктоп-приложений (СУБД MySQL), проектирования архитектуры информационных систем,                  Умеет: проектировать и реализовать простую ИС средствами MySQL+Java</p>
Контрольное мероприятие по разделу			
Промежуточный контроль (количество баллов)		Минимальное количество баллов – 20, максимальное – 40	
Промежуточная аттестация		Представлены в фонде оценочных средств для промежуточной аттестации по дисциплине	

Курс 2 Семестр 4

Вид контроля		Минимальное количество баллов	Максимальное количество баллов
<b>Раздел 3. Специальные инструменты и методы современного программирования</b>			
Текущий контроль по разделу:			
1	Аудиторная работа	13	26
2	Самостоятельная работа (специальные обязательные формы)	5	10
3	Самостоятельная работа (специальные формы на выбор студента)	2	4
Контрольное мероприятие по разделу			
Промежуточный контроль		20	40
Промежуточная аттестация		36	60
Итого:		<b>56</b>	<b>100</b>

Виды контроля	Перечень или примеры заданий, критерии оценки и количество баллов	Темы для изучения и образовательные результаты
<b>Текущий контроль по разделу «Специальные инструменты и методы современного программирования»</b>		
1	<p>Аудиторная работа</p> <p>Лабораторная работа 4. Средствами языка Python обработать естественный текст сообщения, выделив в нем ключевые слова и составить запрос к базе данных услуг ЖКХ. Критерии оценивания:  <ul style="list-style-type: none"> <li>• не решал задачи или решил неправильно – 0 баллов;</li> <li>• задачи решены с несущественными ошибками – 3 балл;</li> <li>• задачи решены без ошибок – 6 балла.</li> </ul>                     Соблюдение графика сдачи заданий – 2 балла                      Итого – <math>6 \times 4 + 2 = 26</math> баллов</p>	<p>Темы:                      Системное программирование и язык C/C++. Сравнительный анализ с Java. Вопросы эффективности и безопасности                      Основные конструкции и типы данных в C++. Массивы. Синтаксические особенности, стандарты, совместимость. Часто используемые библиотеки и функции                      Объектно-ориентированное программирование в C++. Коллекции. Потоки ввода-вывода. Работа с сетью. Низкоуровневое программирование                      Python как специализированный язык. Основной синтаксис. Области применения. Библиотеки для анализа данных, распознавания речи и образов                      Образовательные результаты                      Знает: понятия сплошного и выборочного ручного тестирования, средства автоматизированного тестирования                      Умеет: использовать средства автоматизированного тестирования, проектировать и реализовать простую ИС</p>
2	<p>Самостоятельная работа (обязательные формы)</p> <p>Индивидуальное домашнее задание (ИДЗ) (x2)  <ul style="list-style-type: none"> <li>• решены все задачи ИДЗ – 3 балла;</li> <li>• решения задач с иллюстрациями оформлены развернуто, в</li> </ul></p>	<p>Темы:                      Системное программирование и язык C/C++. Сравнительный анализ с Java. Вопросы эффективности и безопасности</p>



		<p>соответствии с требованиями преподавателя – 1 балл;</p> <ul style="list-style-type: none"> <li>• отчет представлен преподавателю (загружен на проверку в систему управления обучением) в установленные сроки – 1 балл.</li> </ul> <p>Итого – 5x2=10 баллов</p>	<p>Основные конструкции и типы данных в C++. Массивы. Синтаксические особенности, стандарты, совместимость. Часто используемые библиотеки и функции Объектно-ориентированное программирование в C++. Коллекции. Потоки ввода-вывода. Работа с сетью. Низкоуровневое программирование Python как специализированный язык. Основной синтаксис. Области применения. Библиотеки для анализа данных, распознавания речи и образов Образовательные результаты Знает: понятия сплошного и выборочного ручного тестирования, средства автоматизированного тестирования Умеет: использовать средства автоматизированного тестирования, проектировать и реализовать простую ИС</p>
3	Самостоятельная работа (на выбор студента)	<p>Решение задач повышенной сложности.</p> <ul style="list-style-type: none"> <li>• решены все задачи – 3 балла;</li> <li>• решения задач с иллюстрациями оформлены развернуто, в соответствии с требованиями преподавателя – 1 балл.</li> </ul> <p>Итого – 4 балла</p>	<p>Темы: Системное программирование и язык C/C++. Сравнительный анализ с Java. Вопросы эффективности и безопасности Основные конструкции и типы данных в C++. Массивы. Синтаксические особенности, стандарты, совместимость. Часто используемые библиотеки и функции Объектно-ориентированное программирование в C++. Коллекции. Потоки ввода-вывода. Работа с сетью. Низкоуровневое программирование Python как специализированный язык. Основной синтаксис. Области применения. Библиотеки для анализа данных, распознавания речи и образов Образовательные результаты Знает: понятия сплошного и выборочного ручного тестирования, средства автоматизированного тестирования Умеет: использовать средства автоматизированного тестирования, проектировать и реализовать простую ИС</p>
Контрольное мероприятие по разделу			
Промежуточный контроль (количество баллов)		Минимальное количество баллов – 20, максимальное – 40	
Промежуточная аттестация		Представлены в фонде оценочных средств для промежуточной аттестации по дисциплине	