

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Галунин Сергей Александрович  
Должность: проректор по учебной работе  
Дата подписания: 20.03.2023 10:32:52  
Уникальный программный ключ:  
08ef34338325bdb0ac5a47baa5472ce36cc3fc3b

Приложение к ОПОП  
«Компьютерное моделирование  
и проектирование»



**СПбГЭТУ «ЛЭТИ»**  
ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ

МИНОБРНАУКИ РОССИИ

федеральное государственное автономное образовательное учреждение высшего образования  
**«Санкт-Петербургский государственный электротехнический университет  
«ЛЭТИ» им. В.И.Ульянова (Ленина)»**  
**(СПбГЭТУ «ЛЭТИ»)**

---

## **РАБОЧАЯ ПРОГРАММА**

**ДИСЦИПЛИНЫ**

**«ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ KOTLIN»**

для подготовки бакалавров

по направлению

09.03.01 «Информатика и вычислительная техника»

по профилю

**«Компьютерное моделирование и проектирование»**

Санкт-Петербург

2022

## ЛИСТ СОГЛАСОВАНИЯ

Разработчики:

ст. преподаватель, к.т.н. Кузьмин С.А.

Рабочая программа рассмотрена и одобрена на заседании кафедры САПР  
21.12.2021, протокол № 7

Рабочая программа рассмотрена и одобрена учебно-методической комиссией  
ФКТИ, 24.02.2022, протокол № 2

Согласовано в ИС ИОТ

Начальник ОМОЛА Загороднюк О.В.

## 1 СТРУКТУРА ДИСЦИПЛИНЫ

Обеспечивающий факультет	ФКТИ
Обеспечивающая кафедра	САПР
Общая трудоемкость (ЗЕТ)	4
Курс	3
Семестр	6
<b>Виды занятий</b>	
Лекции (академ. часов)	34
Практические занятия (академ. часов)	34
Иная контактная работа (академ. часов)	1
Все контактные часы (академ. часов)	69
Самостоятельная работа, включая часы на контроль (академ. часов)	75
Всего (академ. часов)	144
<b>Вид промежуточной аттестации</b>	
Дифф. зачет (курс)	3

## **2 АННОТАЦИЯ ДИСЦИПЛИНЫ**

### **«ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ KOTLIN»**

В рамках дисциплины “Программирование на языке Kotlin” рассматриваются основные возможности и концепции языка программирования Kotlin применительно к разработке приложений для JVM, изучаются возможности стандартной библиотеки. Также рассматриваются способы автоматизации сборки проектов, в которых используются сторонние библиотеки.

### **SUBJECT SUMMARY**

#### **«KOTLIN PROGRAMMING»**

The discipline “Programming language” describes the general concepts and abilities of the Kotlin programming language and its core APIs used in the development of applications for the JVM. This course also includes exploring the possibilities of using the Gradle build system to assemble Kotlin applications with external libraries.

## **3 ОБЩИЕ ПОЛОЖЕНИЯ**

### **3.1 Цели и задачи дисциплины**

1. Целью дисциплины является формирование устойчивых знаний, умений и практических навыков у студентов в области объектно-ориентированного программирования на языке Kotlin.

2. Задачей дисциплины является дать студенту теоретические и практические знания по принципам объектно-ориентированного проектирования, умения работать с программными инструментами для языка Kotlin, а также выработать у студента навыки разработки объектно-ориентированного программного обеспечения для ЭВМ.

3. Приобретение знаний:

-основных возможностей и концепций языка программирования Kotlin применительно к разработке приложений для выполнения на JVM (виртуальной машине Java);

-о принципах работы JVM и промежуточном представлении кода.

4. Приобретение умений работать в программных инструментах для разработки и отладки приложений для JVM: интегрированной среды разработки IntelliJ IDEA.

5. Формирование практических навыков:

-проектирования и разработки приложений на языке программирования Kotlin под JVM;

-использования интегрированной среды разработки для сборки Kotlin-проектов со сторонними зависимостями.

### **3.2 Место дисциплины в структуре ОПОП**

Дисциплина изучается на основе ранее освоенных дисциплин учебного плана:

1. «Программирование»
2. «Алгоритмы и структуры данных»
3. «Объектно-ориентированное программирование»

и обеспечивает изучение последующих дисциплин:

1. «Инженерный документооборот»

### 3.3 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения образовательной программы обучающийся должен достичь следующие результаты обучения по дисциплине:

<b>Код компетенции/ индикатора компетенции</b>	<b>Наименование компетенции/индикатора компетенции</b>
ПК-0	Способен разрабатывать информационные модели и применять их для решения задач профессиональной деятельности
<i>ПК-0.1</i>	<i>Знает современные виды информационных моделей, применяемых при решении задач профессиональной деятельности</i>
<i>ПК-0.2</i>	<i>Создает и модифицирует информационные модели для решения задач профессиональной деятельности</i>
<i>ПК-0.3</i>	<i>Применяет информационные модели для решения задач профессиональной деятельности</i>

## 4 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 4.1 Содержание разделов дисциплины

#### 4.1.1 Наименование тем и часы на все виды нагрузки

№ п/п	Наименование темы дисциплины	Лек, ач	Пр, ач	ИКР, ач	СР, ач
1	Введение	1	0		2
2	Тема 1. Kotlin и Java Virtual Machine	2	2		4
3	Тема 2. Типы данных в языке программирования Kotlin	2	2		4
4	Тема 3. Управляющие конструкции в языке Kotlin	2	2		4
5	Тема 4. Функции в Kotlin	2	2		4
6	Тема 5. Функции высокого порядка	2	0		4
7	Тема 6. Основы объектно-ориентированного программирования в Kotlin	2	2		4
8	Тема 7. Объекты и классы в Kotlin	4	4		8
9	Тема 8. Расширенные возможности ООП в Kotlin	4	2		6
10	Тема 9. Обобщенное программирование в Kotlin	2	2		4
11	Тема 10. Коллекции в Kotlin	2	2		4
12	Тема 11. Разработка предметно-ориентированных языков на Kotlin	2	0		4
13	Тема 12. Введение в многопоточное программирование на Kotlin	2	0		4
14	Тема 13. Обзор возможностей использования языка Kotlin в разработке различных типов систем	4	12	1	16
15	Тема 14. Сборка проектов с зависимостями на Kotlin	1	2		3
	Итого, ач	34	34	1	75
	Из них ач на контроль	0	0	0	0
	Общая трудоемкость освоения, ач/зе	144/4			

#### 4.1.2 Содержание

№ п/п	Наименование темы дисциплины	Содержание
1	Введение	Язык программирования Kotlin: история появления и развития языка. Актуальные версии языка. Классификация языков программирования. Kotlin в классификации языков программирования. Области применения языка Kotlin.



№ п/п	Наименование темы дисциплины	Содержание
2	Тема 1. Kotlin и Java Virtual Machine	<p>Виртуальные машины для исполнения промежуточного кода.</p> <p>JVM: Основные понятия, особенности исполнения инструкций, байт-код</p> <p>Связь между языком Kotlin и JVM: понятие интероперабельности, интероперабельность Java и Kotlin.</p>
3	Тема 2. Типы данных в языке программирования Kotlin	<p>Типы данных. Встроенные и составные типы данных</p> <p>Численные типы данных и операции над ними. Побитовые операции. Приведение типов.</p> <p>Логический тип данных (boolean).</p> <p>Символьный тип данных.</p> <p>Строковый тип данных.</p> <p>Понятие объекта.</p> <p>Массивы и диапазоны.</p> <p>Перечисления (enum).</p> <p>Переменные и объявления. Ключевые слова var и val.</p> <p>Понятие nullability и null-safety в языке Kotlin. “Ошибка в миллиард долларов” и ее решение в Kotlin.</p>
4	Тема 3. Управляющие конструкции в языке Kotlin	<p>Ветвление алгоритма выполнения: конструкции if, if-else.</p> <p>Понятия statement и expression в языках программирования.</p> <p>Конструкция when.</p> <p>Циклы в Kotlin: конструкции for, while, do-while. Итерация по коллекциям.</p> <p>Прерывание циклов: инструкции break и continue.</p>
5	Тема 4. Функции в Kotlin	<p>Объявление и вызов функций. Ключевые слова fun и return.</p> <p>Аргументы и возвращение значений из функций. Тип Unit.</p> <p>Значение аргументы по умолчанию. Именованные аргументы.</p> <p>Single-expression функции.</p> <p>Нефиксированное число аргументов. Ключевое слово vararg.</p> <p>Понятие функций расширения.</p> <p>Лямбда функции.</p>
6	Тема 5. Функции высокого порядка	<p>Функция как параметр другой функции. Возвращения функции из функции.</p>
7	Тема 6. Основы объектно-ориентированного программирования в Kotlin	<p>Основные парадигмы ООП: инкапсуляция, наследование, полиморфизм</p> <p>Наследование классов. Интерфейсы. Модификаторы доступа. Пакеты. Структурирование кода в проекте.</p>
8	Тема 7. Объекты и классы в Kotlin	<p>Понятие объекта и класса. Синтаксис описания классов. Создание объектов, ссылка на объект. Сравнение объектов по ссылке и по значению. Контракт equals-hashCode. Поля, свойства и методы классов. Константы. Поздняя инициация свойств. Делегаты.</p>

№ п/п	Наименование темы дисциплины	Содержание
9	Тема 8. Расширенные возможности ООП в Kotlin	Функции расширения, свойства расширения и ООП. Классы данных (data class) Изолированные (запечатанные) классы (sealed class) Вложенные классы. Паттерны проектирования.
10	Тема 9. Обобщенное программирование в Kotlin	Обобщенные типы в Kotlin. Вариативность. Проекция типов. Обобщенные функции. Обобщенные ограничения.
11	Тема 10. Коллекции в Kotlin	Типы коллекций. Списки. Множества. Ассоциативные массивы (map) Основные интерфейсы и реализация. Иерархия коллекций. Изменяемые и неизменяемые коллекции. Операции на коллекциями. Фильтрация коллекций. Фильтрация с предикатом. Фильтрация с разделением по группам. Преобразование коллекций: операции map, zip, associateWith, flatten Группировка элементов коллекции: операции groupBy, groupingBy, fold, reduce.
12	Тема 11. Разработка предметно-ориентированных языков на Kotlin	Понятие предметно-ориентированных языков (DSL). Использование Kotlin для разработки DSL. Разбор примеров DSL на Kotlin.
13	Тема 12. Введение в многопоточное программирование на Kotlin	Многопоточное программирование: основные понятие и проблемы. Асинхронное программирование в Kotlin. Корутины в Kotlin. Ключевое слово suspend. Выполнение корутин. Контекст выполнения. Примеры использования корутин с java-библиотекой Swing.
14	Тема 13. Обзор возможностей использования языка Kotlin в разработке различных типов систем	Kotlin и JavaScript. Мультиплатформенное программирование (Kotlin MPP). Kotlin в разработке Android приложений.
15	Тема 14. Сборка проектов с зависимостями на Kotlin	Понятие системы сборки проекта. Популярные систем сборки. Система сборки Gradle. Основные понятия: артефакты, GAV система координат для идентификации зависимостей.

## 4.2 Перечень лабораторных работ

Лабораторные работы не предусмотрены.

## 4.3 Перечень практических занятий

Наименование практических занятий	Количество ауд. часов
1. Установка и настройка интегрированной среды разработки (IDE). Создание проекта приложения в IDE.	2
2. Выбор и использование паттерна проектирования при разработке архитектуры объектно-ориентированного приложения.	2

<b>Наименование практических занятий</b>	<b>Количество ауд. часов</b>
3. Разработка классов для представления модели данных в памяти приложения.	8
4. Разработка интерфейса для взаимодействия приложения с пользователем.	6
5. Чтение и обработка данных с внешних источников (файлов, баз данных, внешних хранилищ и т.п.).	8
6. Вывод результатов программы (в том числе на сторонних сервисах).	6
7. Сборка и публикация проекта.	2
Итого	34

#### **4.4 Курсовое проектирование**

Курсовая работа (проект) не предусмотрены.

#### **4.5 Реферат**

Реферат не предусмотрен.

#### **4.6 Индивидуальное домашнее задание**

Индивидуальное домашнее задание (ИДЗ) представляет собой комплексный программный проект, выполняемый на основе пройденного студентами цикла практических занятий.

Цель ИДЗ - продемонстрировать знания, навыки и умения в объектно-ориентированном программировании, приобретенные за время прохождения курса.

Для выполнения проекта в рамках ИДЗ, студенты разбиваются на бригады по 1-2 человека. Каждая бригада перед выполнением ИДЗ согласовывает с преподавателем тему проекта (или выбирает его из предложенных им, или же предлагает сама) и конкретное техническое задание на разработку приложения.

Примеры тем ИДЗ: "Разработка Telegram-бота с расписанием занятий", "Разработка интернет-магазина".

Вне зависимости от выбранной темы проекта, в рамках ИДЗ выполнение работ

состоит из следующих этапов:

1. Установка и настройка интегрированной среды разработки (IDE). Создание проекта приложения в IDE.
2. Выбор и использование паттерна проектирования при разработке архитектуры объектно-ориентированного приложения.
3. Разработка классов для представления модели данных в памяти приложения.
4. Разработка интерфейса для взаимодействия приложения с пользователем.
5. Чтение и обработка данных с внешних источников
6. Вывод результатов программы
7. Сборка и публикация проекта.

Все этапы ИДЗ соответствуют циклу практических занятий. Потому, получая от преподавателя свой вариант задания, студенты в рамках ИДЗ, фактически, выполняют практическую работу согласно своему варианту.

Для контроля выполнения студентами работы в рамках ИДЗ, предполагается проведение в семестре следующих контрольных точек:

1. Формирование бригады. Согласование темы и задания на ИДЗ.
2. Разработка архитектуры приложения.
3. Разработка программной модели данных приложения.
4. Разработка интерфейса приложения.
5. Разработка программного модуля для чтения и обработки данных.
6. Разработка программного модуля для вывода результатов программы.

По окончании разработки, студенты готовят отчет по ИДЗ.

Требования к отчету по ИДЗ:

Оформление отчетных материалов должно происходить согласно шаблонам оформления студенческих работ, размещенным на официальном сайте университета.

Количество литературных источников - от 3 до 10 шт.

Объем отчета - от 10 до 50 стр.

Электронный документ должен быть в формате .doc, .docx или .pdf, шрифт Times new Roman 12пт. с одинарным межстрочным интервалом.

Рисунки нумеруются арабскими цифрами сквозной нумерацией и располагаются по центру страницы, подпись под рисунком. Таблица предваряется заголовком, включающим слово «Таблица» (с указанием номера, выравнивается по правому краю) и наименование таблицы.

Работа содержит следующие разделы (некоторые из них могут отсутствовать в соответствии с особенностями формулировки и выполнения ИДЗ):

1. Исходная формулировка задания
2. Спецификация программы
3. Описание интерфейса пользователя программы
4. Текст программы
5. Выводы

Отчет по ИДЗ сдается преподавателю в печатном виде, а также отправляется на электронную почту.

#### **4.7 Доклад**

Доклад не предусмотрен.

#### **4.8 Кейс**

Кейс не предусмотрен.

#### **4.9 Организация и учебно-методическое обеспечение самостоятельной работы**

Изучение дисциплины сопровождается самостоятельной работой студентов с рекомендованными преподавателем литературными источниками и информационными ресурсами сети Интернет.

Планирование времени для изучения дисциплины осуществляется на весь

период обучения, предусматривая, при этом, регулярное повторение пройденного материала.

Обучающимся, в рамках внеаудиторной самостоятельной работы, необходимо регулярно дополнять сведениями из литературных источников материал, законспектированный на лекциях. При этом, на основе изучения рекомендованной литературы целесообразно составить конспект основных положений, терминов и определений, необходимых для освоения разделов учебной дисциплины.

Особое место уделяется консультированию, как одной из форм обучения и контроля самостоятельной работы.. Консультирование предполагает особым образом организованное взаимодействие между преподавателем и студентами, при этом предполагается, что консультант либо знает готовое решение, которое он может предписать консультируемому, либо он владеет способами деятельности, которые указывают путь решения проблемы.

<b>Текущая СРС</b>	<b>Примерная трудоемкость, ач</b>
Работа с лекционным материалом, с учебной литературой	26
Опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	0
Самостоятельное изучение разделов дисциплины	12
Выполнение домашних заданий, домашних контрольных работ	27
Подготовка к лабораторным работам, к практическим и семинарским занятиям	7
Подготовка к контрольным работам, коллоквиумам	0
Выполнение расчетно-графических работ	0
Выполнение курсового проекта или курсовой работы	0
Поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	0
Работа над междисциплинарным проектом	0
Анализ данных по заданной теме, выполнение расчетов, составление схем и моделей, на основе собранных данных	0
Подготовка к зачету, дифференцированному зачету, экзамену	3
<b>ИТОГО СРС</b>	<b>75</b>

## 5 Учебно-методическое обеспечение дисциплины

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

№ п/п	Название, библиографическое описание	К-во экз. в библ.
Основная литература		
1	Жемеров Д. Kotlin в действии [Электронный ресурс], 2018. -402 с.	неогр
2	Анисимов, Владимир Иванович. Объектно-ориентированное программирование интерфейсных средств на Java 2 [Электронный ресурс] : учеб. пособие / В.И. Анисимов, 2008. -1 эл. опт. диск (CD-ROM)	неогр.
3	Скин Джош Kotlin. Программирование для профессионалов [Электронный ресурс] / Джош Скин, Дэвид Гринхол, 2019. -464 с.	неогр.
Дополнительная литература		
1	Сомон. П. И. Волшебство Kotlin [Электронный ресурс] : руководство / П. И. Сомон., 2020. -536 с.	неогр
2	Дон Гриффитс Head First. Kotlin [Электронный ресурс] / Гриффитс Дон, Гриффитс Дэвид, 2021. -464 с.	неогр.

### 5.2 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», используемых при освоении дисциплины

№ п/п	Электронный адрес
1	Online Kotlin Playground <a href="https://play.kotlinlang.org/">https://play.kotlinlang.org/</a>
2	Kotlin Language Docs <a href="https://kotlinlang.org/docs/home.html">https://kotlinlang.org/docs/home.html</a>
3	Руководство по языку Kotlin на русском языке. <a href="https://kotlinlang.ru/">https://kotlinlang.ru/</a>
4	Gradle Build Tool Docs <a href="https://docs.gradle.org/current/userguide/userguide.html">https://docs.gradle.org/current/userguide/userguide.html</a>

### 5.3 Адрес сайта курса

Адрес сайта курса: <https://vec.etu.ru/moodle/course/view.php?id=11334>

## 6 Критерии оценивания и оценочные материалы

### 6.1 Критерии оценивания

Для дисциплины «Программирование на языке Kotlin» предусмотрены следующие формы промежуточной аттестации: зачет с оценкой.

#### Зачет с оценкой

<b>Оценка</b>	<b>Описание</b>
Неудовлетворительно	Курс не освоен. Студент испытывает серьезные трудности при ответе на ключевые вопросы дисциплины
Удовлетворительно	Студент в целом овладел курсом, но некоторые разделы освоены на уровне определений и формулировок
Хорошо	Студент овладел курсом, но в отдельных вопросах испытывает затруднения. Умеет решать задачи
Отлично	Студент демонстрирует полное овладение курсом, способен применять полученные знания при решении конкретных задач.



## Особенности допуска

Доступ студентов к промежуточной аттестации по дисциплине (дифференцированному зачёту) осуществляется только после выполнении ими всех этапов индивидуального домашнего задания (ИДЗ), что фиксируется в результатах прохождения ими контрольных точек по нему.

Перед проведением дифференцированного отчёта, студенты должны предоставить преподавателю:

- файлы проекта (в архиве, или же опубликованные на стороннем сервисе);
- отчёт по ИДЗ.

В случае, если до проведения дифференцированного зачёта, студенты успели сдать преподавателю все разработанные ими программные модули приложения -то на самом дифф. зачёте происходит проверка оформленных ими отчётов по нему. Проверка отчёта осуществляется в присутствии всех студентов бригады. В процессе проверки отчёта, преподаватель вправе задавать студентам бригады уточняющие вопросы по использованным ими технологиям, алгоритмам и программным конструкциям. Убедившись в самостоятельности выполнения всех этапов ИДЗ и получив ответы на интересующие вопросы, преподаватель принимает решение об окончательной оценке по дисциплине для студента (критерии оценивания см. выше).

## 6.2 Оценочные материалы для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине

### Вопросы к дифф.зачету

№ п/п	Описание
1	Какой функционал выполняет программа?
2	Какие паттерны проектирования Вы использовали в программе?
3	Какие классы были разработаны для объектно-ориентированной модели данных приложения?
4	Как реализован интерфейс взаимодействия с пользователем?
5	Как реализовано чтение данных в программе?

6	Как реализована обработка входных данных в программе?
7	Как реализован вывод данных в программе?
8	Какие сторонние библиотеки были использованы в программе?
9	Как реализовано хранение данных в программе?
10	Работа с какой СУБД была реализована в программе?
11	Какие методы для работы с БД были реализованы?
12	Покажите модель данных для работы с БД
13	Какие алгоритмы были использованы в программе?
14	Как обрабатываются некорректные входные данные?
15	Был ли использован многопоточный режим обработки данных?
16	Покажите созданную диаграмму классов
17	Какая сложность разработанных алгоритмов?
18	Можно ли оптимизировать существующий код?
19	В какой среде Вы осуществляли сборку проекта?
20	С помощью каких сервисов код был опубликован в общий доступ?

Весь комплект контрольно-измерительных материалов для проверки сформированности компетенции (индикатора компетенции) размещен в закрытой части по адресу, указанному в п. 5.3

### 6.3 График текущего контроля успеваемости

Неделя	Темы занятий	Вид контроля
1	Введение	
2		Практическая работа
3	Тема 8. Расширенные возможности ООП в Kotlin	
4		Практическая работа
5	Тема 2. Типы данных в языке программирования Kotlin Тема 6. Основы объектно-ориентированного программирования в Kotlin Тема 7. Объекты и классы в Kotlin	
6		
7		
8		Практическая работа
9	Тема 13. Обзор возможностей использования языка Kotlin в разработке различных типов систем	
10		Практическая работа
11	Тема 3. Управляющие конструкции в языке Kotlin Тема 4. Функции в Kotlin	
12		
13		
14		Практическая работа
15	Тема 13. Обзор возможностей использования языка Kotlin в разработке различных типов систем Тема 14. Сборка проектов с зависимостями на Kotlin	
16		Практическая работа
17	Тема 13. Обзор возможностей использования языка Kotlin в разработке различных типов систем	
18		ИДЗ / ИДРГЗ / ИДРЗ

### 6.4 Методика текущего контроля

#### на лекционных занятиях

Текущий контроль включает в себя контроль посещаемости (не менее **80** % занятий), по результатам которого студент получает допуск к дифф. зачету.

#### на практических (семинарских) занятиях

Текущий контроль включает в себя контроль посещаемости (не менее **80** % занятий), по результатам которого студент получает допуск к дифф. зачету.

Активность студентов на практических занятиях также может учитываться преподавателем, как один из способов текущего контроля на практических занятиях.

#### самостоятельной работы студентов

Контроль самостоятельной работы студентов осуществляется на лекци-

онных и практических занятиях студентов по методикам, описанным выше.

### **При выполнении индивидуального домашнего задания**

Текущий контроль при выполнении индивидуального домашнего задания (ИДЗ) осуществляется в соответствии с пунктом 4.6 настоящей рабочей программы.

ИДЗ оценивается следующим образом:

- неудовлетворительно - грубые ошибки в проектных решениях, программная реализация не работает;
- удовлетворительно - ошибки в проектных решениях, программная реализация работает частично;
- хорошо - ошибки в проектных решениях, программная реализация работает полностью;
- отлично - в проектных решениях ошибки отсутствуют, программная реализация работает полностью.

Оформление отчета по индивидуальному домашнему заданию выполняется в соответствии с требованиями к студенческим работам принятым в СПбГ-ЭТУ "ЛЭТИ".

Защита индивидуального домашнего задания осуществляется в соответствии с пунктом 6.1 настоящей рабочей программы.

## 7 Описание информационных технологий и материально-технической базы

Тип занятий	Тип помещения	Требования к помещению	Требования к программному обеспечению
Лекция	Лекционная аудитория	Количество посадочных мест – в соответствии с контингентом, рабочее место преподавателя, ПК или ноутбук, проектор, меловая или маркерная доска	1) Windows 7 и выше; 2) Microsoft Office 2007 и выше
Практические занятия	Аудитория	Количество посадочных мест – в соответствии с контингентом, рабочее место преподавателя, проектор, экран, ПК или ноутбук, меловая или маркерная доска	1) Windows 7 и выше; 2) Microsoft Office 2007 и выше 3) IntelliJ IDEA версии 2021.2 или старше
Самостоятельная работа	Помещение для самостоятельной работы	Оснащено компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.	1) Windows XP и выше; 2) Microsoft Office 2007 и выше

## **8 Адаптация рабочей программы для лиц с ОВЗ**

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медико-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.

## ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

<b>№ п/п</b>	<b>Дата</b>	<b>Изменение</b>	<b>Дата и номер протокола заседания УМК</b>	<b>Автор</b>	<b>Начальник ОМОЛА</b>