

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Галунин Сергей Александрович
Должность: проректор по учебной работе
Дата подписания: 28.06.2023 14:55:53
Уникальный программный ключ:
08ef34338325bdb0ac5a47baa5472ce36cc3fc3b

Приложение к ОПОП
«Математические методы в ин-
формационных технологиях»



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

«ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ»

для подготовки бакалавров

по направлению

01.03.02 «Прикладная математика и информатика»

по профилю

«Математические методы в информационных технологиях»

Санкт-Петербург

2023

ЛИСТ СОГЛАСОВАНИЯ

Разработчики:

доцент, к.т.н. Заславский М.М.

Рабочая программа рассмотрена и одобрена на заседании кафедры МОЭВМ
14.02.2023, протокол № 2

Рабочая программа рассмотрена и одобрена учебно-методической комиссией
ФКТИ, 16.02.2023, протокол № 2

Согласовано в ИС ИОТ

Начальник ОМОЛА Загороднюк О.В.

1 СТРУКТУРА ДИСЦИПЛИНЫ

Обеспечивающий факультет	ФКТИ
Обеспечивающая кафедра	МОЭВМ
Общая трудоемкость (ЗЕТ)	3
Курс	1
Семестр	2
Виды занятий	
Лекции (академ. часов)	34
Лабораторные занятия (академ. часов)	34
Иная контактная работа (академ. часов)	1
Все контактные часы (академ. часов)	69
Самостоятельная работа, включая часы на контроль (академ. часов)	39
Всего (академ. часов)	108
Вид промежуточной аттестации	
Дифф. зачет (курс)	1

2 АННОТАЦИЯ ДИСЦИПЛИНЫ

«ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ»

Данная дисциплина включает рассмотрение разделов по темам: "Парадигмы программирования", "Введение в алгоритмы и структуры" и "Введение в анализ данных". При рассмотрении парадигм программирования студенты знакомятся с основными определениями и техниками программирования, в частности изучают объектно-ориентированное и функциональное программирование с практическими примерами на Python. В разделе "Введение в алгоритмы и структуры" студенты знакомятся с основными структурами данных, алгоритмов поиска, сортировок и асимптотической оценки сложности; практикуются в реализации связанных списков и работе с ними. В разделе "Введение в анализ данных" студенты знакомятся с основными понятиями и определениями, рассматривается введение в классические задачи машинного обучения, изучаются основы предварительной обработки данных с помощью модуля pandas, выполняется обучение простых моделей с помощью библиотеки scikit-learn.

SUBJECT SUMMARY

«INFORMATION TECHNOLOGY»

This discipline includes consideration of sections on the topics: "Programming Paradigms", "Introduction to Algorithms and Structures" and "Introduction to Data Analysis". When considering programming paradigms, students are introduced to the basic definitions and techniques of programming, in particular, they study object-oriented and functional programming with practical examples in Python. In the "Introduction to Algorithms and Structures" section, students are introduced to basic data structures, search algorithms, sortings, and asymptotic complexity estimation; Practice implementing linked lists and working with them. In the Introduction to Data Analysis section, students are introduced to the basic concepts and definitions, an introduction to classical machine learning problems is considered, the basics of data

preprocessing are studied using the pandas module, and simple models are trained using the scikit-learn library.

3 ОБЩИЕ ПОЛОЖЕНИЯ

3.1 Цели и задачи дисциплины

1. Цели изучения дисциплины: при освоении дисциплины обучающиеся должны получить теоретические знания о парадигмах программирования, базовых алгоритмах и структурах данных, алгоритмах обработки табличных данных, задачах кластеризации и классификации, а также практические навыки применения их в профессиональной деятельности.

2. Задачи дисциплины включают:

-овладение знаниями о стандартной терминологии, определениях и обозначениях;

-выработку навыков выбора и использования подходящей парадигмы программирования для решения практических задач;

-приобретение умения проводить оценку асимптотической сложности базовых алгоритмов и сложности работы с базовыми структурами данных;

-работать с табличными данными при решении задач кластеризации и классификации.

3. В результате изучения дисциплины должны быть получены знания:

-основных определений парадигм программирования; основных принципов объектно-ориентированного программирования;

-основных принципов функционального программирования;

-базовых алгоритмов и структур данных и основных принципов работы с ними;

-основных принципов работы с табличными данными при решении задач задач кластеризации и классификации.

4. В процессе изучения дисциплины формируются умения:

-выбирать и использовать подходящую парадигму программирования для решения практических задач;

-оценить асимптотическую сложность базового алгоритма алгоритма; оценить сложность работы с базовой структурой данных;
-работать с табличными данными при решении задач кластеризации и классификации.

5. В процессе изучения дисциплины обучающиеся приобретают навыки:

-работы с разными парадигмами программирования на языке Python для решения практических задач;
-анализа наилучшего, наихудшего и среднего случая работы алгоритма;
-обработки табличных данных при решении задач кластеризации и классификации.

3.2 Место дисциплины в структуре ОПОП

Дисциплина изучается на основе ранее освоенных дисциплин учебного плана:

1. «Информатика»

и обеспечивает изучение последующих дисциплин:

1. «Алгоритмы и структуры данных»

2. «Дискретная математика и теоретическая информатика»

3. «Объектно-ориентированное программирование»

4. «Организация ЭВМ и систем»

5. «Операционные системы»

6. «Построение и анализ алгоритмов»

7. «Базы данных»

8. «Основы промышленной разработки программного обеспечения»

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:

Код компетенции/ индикатора компетенции	Наименование компетенции/индикатора компетенции
ОПК-2	Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач
<i>ОПК-2.1</i>	<i>Знает существующие математические методы и современные системы программирования, в том числе отечественного производства</i>
ОПК-4	Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности
<i>ОПК-4.3</i>	<i>Имеет навыки подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций, и библиографии по научно исследовательской работе с учетом требований информационной безопасности</i>
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>

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	Лаб, ач	ИКР, ач	СР, ач
1	Введение	1			
2	Тема 1.1 Парадигмы программирования. Объектно ориентированное программирование	4	4		2
3	Тема 1.2 Функциональное программирование. Исключения	4	4		2
4	Тема 2.1 Введение в алгоритмы (поиск, сортировка). Оценка сложности	4	4		3
5	Тема 2.2 Введение в структуры данных. Линейный список	6	6		3
6	Тема 3.1 Введение в анализ данных. Основные определения	6	6		4
7	Тема 3.2 Предварительная обработка данных. Визуализация данных. Модули pandas и matplotlib	4	4		5
8	Тема 3.3 Элементы машинного обучения. Модуль scikit-learn	4	6	0	5
9	Заключение	1		1	15
	Итого, ач	34	34	1	39
	Из них ач на контроль	0	0	0	0
	Общая трудоемкость освоения, ач/зе	108/3			

4.1.2 Содержание

№ п/п	Наименование темы дисциплины	Содержание
1	Введение	Обзор курса. Основные цели и задачи. Предмет и задачи дисциплины. Содержание курса.
2	Тема 1.1 Парадигмы программирования. Объектно ориентированное программирование	Итеративное и декларативное программирование. Функциональное программирование. Реализация лямбда функций в языке Python. Введение в объектно-ориентированное программирование на языке Python.
3	Тема 1.2 Функциональное программирование. Исключения	Чистые функции. Функции высшего порядка. Итератор. Итерируемый объект. Функциональное программирование на Python: функции map, filter, zip и др. lambda-функции. Применение функционального программирования в практических задачах.

№ п/п	Наименование темы дисциплины	Содержание
4	Тема 2.1 Введение в алгоритмы (поиск, сортировка). Оценка сложности	Анализ сложности наилучшего наихудшего и среднего случая. Асимптотические обозначения. Скорость роста функции. Отношения доминирования. Сложение и умножение функций. Сортировки: алгоритм, сложность и реализация. Алгоритмы на строках.
5	Тема 2.2 Введение в структуры данных. Линейный список	Список и массивы: отличия. Линейный список. Очередь. Стек. Дерево двоичного поиска. Основные методы таких структур данных.
6	Тема 3.1 Введение в анализ данных. Основные определения	Основные определения: классические задачи, машинное обучение, регрессия, кластеризация, классификация. Разные виды данных для машинного обучения: текстовые, графические и табличные данные.
7	Тема 3.2 Предварительная обработка данных. Визуализация данных. Модули pandas и matplotlib	Основные задачи предварительной обработки данных. Основы работы с методом pandas. Визуализация данных: основные виды графиков. Использование модуля matplotlib при решении практических задач.
8	Тема 3.3 Элементы машинного обучения. Модуль scikit-learn	Методы сокращения размерности пространства признаков: применение метода главных компонент. Основные задачи машинного обучения на примере модуля scikit learn. Пример решения задачи регрессии, кластеризации и классификации.
9	Заключение	Основные выводы по курсу

4.2 Перечень лабораторных работ

Наименование лабораторной работы	Количество ауд. часов
1. Парадигмы программирования	8
2. Введение в алгоритмы и структуры данных	10
3. Введение в анализ данных	16
Итого	34

4.3 Перечень практических занятий

Практические занятия не предусмотрены.

4.4 Курсовое проектирование

Курсовая работа (проект) не предусмотрены.

4.5 Реферат

Реферат не предусмотрен.

4.6 Индивидуальное домашнее задание

Индивидуальное домашнее задание не предусмотрено.

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.

Текущая СРС	Примерная трудоемкость, ач
Работа с лекционным материалом, с учебной литературой	10
Опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	0
Самостоятельное изучение разделов дисциплины	0
Выполнение домашних заданий, домашних контрольных работ	0
Подготовка к лабораторным работам, к практическим и семинарским занятиям	7
Подготовка к контрольным работам, коллоквиумам	7
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	15
ИТОГО СРС	39

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	Информатика. Введение в Python [Электронный ресурс] : электрон. учеб. пособие / К. В. Кринкин [и др.], 2020. -1 эл. опт. диск (CD-ROM)	неогр.
Дополнительная литература		
1	Кнут Д.Э. Искусство программирования [Текст] : в 3 т. : [учеб. пособие] : пер. с англ. -(Классический труд). Т. 1 : Основные алгоритмы : учебное пособие, 2000. -712 с.	14
2	Кнут Д. Э. Искусство программирования: В 3 т.: Пер. с англ [Текст] : Учеб. пособие. Т. 2 : Получисленные алгоритмы : учебное пособие, 2000. -828 с.	14
3	Кнут Д.Э. Искусство программирования: В 3 т.: Пер с англ [Текст] : Учеб. пособие. Т. 3 : Сортировка и поиск : учебное пособие, 2000. -822 с.	14

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

№ п/п	Электронный адрес
1	Введение в теоретическую информатику https://stepik.org/course/104/syllabus
2	Теоретическая информатика: вычислимость https://stepik.org/course/1611/syllabus
3	Алгоритмы: теория и практика https://stepik.org/course/217/syllabus
4	Теоретическая информатика: сложность вычислений https://stepik.org/course/1613/syllabus
5	Алгоритмы: теория и практика. Структуры данных https://stepik.org/course/1547/syllabus
6	Информатика. Практические задания. Второй семестр http://e.moevm.info/course/view.php?id=69

5.3 Адрес сайта курса

Адрес сайта курса: <https://vec.etu.ru/moodle/course/view.php?id=10607>

6 Критерии оценивания и оценочные материалы

6.1 Критерии оценивания

Для дисциплины «Информационные технологии» формой промежуточной аттестации является дифф. зачет. Оценивание качества освоения дисциплины производится с использованием рейтинговой системы.

Дифференцированный зачет

Оценка	Количество баллов	Описание
Неудовлетворительно	0 – 6	теоретическое содержание курса не освоено, необходимые практически навыки и умения не сформированы, выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над курсом не приведет к существенному повышению качества выполнения учебных заданий
Удовлетворительно	7 – 15	теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки и умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки
Хорошо	16 – 21	теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки и умения сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками
Отлично	22 – 25	теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки и умения сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено количеством баллов, близким к максимальному

Особенности допуска

До дифференцированного зачета допускается студент набравший по рейтингу не менее 7 баллов в совокупности по всем видам текущего контроля (см. п.6.4). Оценка за промежуточную аттестацию по выбору студента может быть выставлена на основании рейтинга, либо студент отвечает на вопросы билета. Студенты, не набравшие 7 баллов, проходят промежуточную аттестацию в форме устных ответов на билет.

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

Вопросы к дифф.зачету

№ п/п	Описание
1	Объектно-ориентированное программирование: объект, класс, атрибуты
2	Наследование, полиморфизм, инкапсуляция. Исключения
3	Функциональное программирование: чистые функции, функции высшего порядка, отношение детерминирования функций
4	Итератор, итерируемый объект
5	Функции map, filter, zip, lambda
6	Базовые алгоритмы: сортировка пузырьком
7	Сортировка слиянием
8	Сортировка вставками, TimSort
9	Алгоритм бинарного поиска
10	Асимптотическая оценка сложности алгоритмов.
11	Нотация O-большое
12	Скорость роста функций
13	Базовые структуры данных. Массивы, линейные списки, хеш-таблицы
14	Оценка сложности вставки, удаление и поиска в базовых структурах данных
15	Введение в анализ данных. Основные определения
16	Классические задачи машинного обучения
17	Виды машинного обучения
18	Основные задачи предварительной обработки данных
19	Визуализация данных
20	Работа с модулями pandas и matplotlib
21	Определение задач регрессии
22	Определение задач кластеризации и классификации
23	Основы работы с модулем scikit-learn

Форма билета

Министерство науки и высшего образования Российской Федерации
ФГАОУ ВО «Санкт-Петербургский государственный электротехнический
университет «ЛЭТИ» имени В.И. Ульянова (Ленина)»

Дифференцированный зачет. БИЛЕТ № 1

Дисциплина **Информационные технологии ФКТИ**

1. Итератор, итерируемый объект. Функции `map`, `filter`, `zip`, `lambda`
2. Определение задач регрессии, кластеризации и классификации
3. Задача. Напишите класс `MyStr`, который наследуется от встроенного класса `str` и обладает следующими свойствами: символы строки выводятся на экран через символ дефис `"-"`, а длина строки определяется количеством заглавных символов в строке.

УТВЕРЖДАЮ

Заведующий кафедрой

К.В. Кринкин

Образцы задач (заданий) для контрольных (проверочных) работ

Вариант № 1

1. Реализуйте класс `MyStr`, который наследуется от встроенного класса `str` и обладает следующими свойствами: символы строки выводятся на экран через символ дефис `"-"`, а длина строки определяется количеством заглавных символов в строке.
2. Реализуйте класс `MyList`, который наследуется от встроенного класса `list` и обладает следующими свойствами: длина списка определяется количеством строк четной длины, а метод `join` работает как для элементов-строк, так и для элементов-чисел (целых, с плавающей точкой, комплексных).

3. Приведите значения оценок сложности вставки и удаление элементов в массиве и линейный односвязный списке.

4. Реализуйте алгоритм бинарного поиска на числах с плавающей точкой. Точность для чисел считать равной: $\epsilon=0.0005$.

5. Решите задачи кластеризации и классификации на наборе данных, предоставленном преподавателем.

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	Тема 1.1 Парадигмы программирования. Объектно ориентированное программирование Тема 1.2 Функциональное программирование. Исключения	
2		
3		
4		
5		
6		
7		
8		Отчет по лаб. работе
9	Тема 2.1 Введение в алгоритмы (поиск, сортировка). Оценка сложности Тема 2.2 Введение в структуры данных. Линейный список	
10		
11		
12		Отчет по лаб. работе
13	Тема 3.1 Введение в анализ данных. Основные определения Тема 3.2 Предварительная обработка данных. Визуализация данных. Модули pandas и matplotlib Тема 3.3 Элементы машинного обучения. Модуль scikit-learn	
14		
15		
16		Отчет по лаб. работе
17		Контрольная работа

6.4 Методика текущего контроля

На лабораторных занятиях.

Порядок выполнения лабораторных работ, подготовки отчетов и их защиты.

В процессе обучения студент может выполнить 3 лабораторные работы. Под выполнением лабораторных работ подразумевается подготовка к работе, проведение экспериментальных исследований (решение задачи лабораторной работы), подготовка отчета и его защита на занятии. Студент имеет право защищать лабораторную работу на любом занятии по расписанию до установленного рейтинговой системой срока и на одном занятии по расписанию, следующем на неделе после установленного срока, если до этого не было попыток защиты.

Выполнение лабораторных работ студентами осуществляется индивидуально. Оформление отчета студентами осуществляется индивидуально в соот-

ветствии с принятыми в СПбГЭТУ правилами оформления студенческих работ. Отчет оформляется после выполнения экспериментальных исследований (после решения задачи), и представляется преподавателю на проверку вместе с исходным кодом соответствующей лабораторной работы через репозиторий. После проверки преподавателем отчет либо возвращается (при наличии замечаний) на доработку, либо принимается.

Лабораторные работы защищаются студентами индивидуально. Каждый студент в дистанционном курсе для защиты лабораторных работ получает вариант на защиту, который включает три задания: тест по теоретической части и задачи на написание программного кода по пройденным темам (аналоги задач дистанционного курса или подзадач лабораторной работы), после чего студенту предоставляется время для подготовки ответа. В случае если студент правильно решил все три задачи и продемонстрировал достаточное знание темы при ответе на уточняющие вопросы преподавателя, работа считается защищенной на 3 балла. Если студент решил 1 или 2 задачи, то работа считается защищенной и студент получает 1 или 2 балла соответственно. Если студент не решил ни одной задачи, работа не считается защищенной и студент получает 0 баллов. Студент, получивший 0 баллов, может повторно защитить работу на следующих занятиях по расписанию до установленного срока.

На защите лабораторной работы студент должен показать: понимание методики исследования и знание особенностей её применения, понимание и умение объяснять особенности применяемых методов, возможные области их применения и т.д., умение давать качественную и количественную оценку полученных экспериментальных результатов и прогнозировать реакции исследуемого объекта на различные воздействия, навыки и умения, приобретенные при выполнении лабораторной работы.

Работа студента в семестре включает следующие активности:

- 1) выполнение и сдачу отчетов и исходного кода по соответствующим ла-

бораторным работам в соответствии с установленными сроками (по 2 балла за каждую лабораторную работу, максимально студент может получить 6 баллов за 3 лабораторные работы);

2) защиту лабораторных работ в соответствии с установленными сроками (0, 1, 2, 3) балла за защиту каждой лабораторной работы, максимально студент может получить 9 баллов: за защиту на 3 балла каждой из 3-х лабораторных работ; защита лабораторных работ возможна только при сдаче исходного кода и отчета в п. 1);

3) прохождение указанных дистанционных курсов в соответствии с установленными сроками (0-5) баллов за основной курс (курс №6 в пункте 5.2) по предмету, максимальное количество баллов – 5 баллов);

4) решение задач контрольной работы в конце семестра (0-5) баллов за решение 5 задач контрольной работы, если задача полностью решена – 1 балл, в противном случае – 0 баллов; максимальное количество баллов за контрольную работу – 5 баллов).

Итого, максимальное количество баллов, которое может получить студент в течение семестра составляет 25 баллов.

Каждая из активностей ограничена по времени. Своевременное выполнение активностей приносит максимальное количество баллов. Несоблюдение сроков выполнения активностей приводит к невозможности получения максимального количества баллов. Студент вправе сам выбирать активности под определенное количество баллов для получения желаемой оценки.

Самостоятельной работы студентов.

Контроль самостоятельной работы студентов осуществляется на лабораторных занятиях и на контрольной работе в конце семестра.

Формирования итоговой оценки по дисциплине

Совокупность баллов, полученных студентом в результате выполнения

активностей текущего контроля в течение семестра, учитывается преподавателем при проведении промежуточной аттестации в форме дифференцированного зачета. Градация баллов представлена в п. 6.1.

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, IBM-совместимый компьютер Pentium или выше, проектор, экран, меловая или маркерная доска	1) Linux Альт Образование 10 и выше; 2) P7-Офис 7 и выше либо LibreOffice 7 и выше
Лабораторные работы	Лаборатория	Количество посадочных мест – в соответствии с контингентом, рабочее место преподавателя. Оснащено компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.	1) Linux Альт Образование 10 и выше; 2) P7-Офис 7 и выше либо LibreOffice 7 и выше
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) Linux Альт Образование 10 и выше; 2) P7-Офис 7 и выше либо LibreOffice 7 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

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