

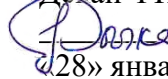
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«КАМЧАТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КамчатГТУ»)

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ЭКОНОМИКИ И УПРАВЛЕНИЯ

Кафедра «Системы управления»

УТВЕРЖДАЮ

Декан ФИТЭУ

 И. А. Рычка/
«28» января 2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Программирование на Python»

направление подготовки (специальность)

09.03.03 Прикладная информатика

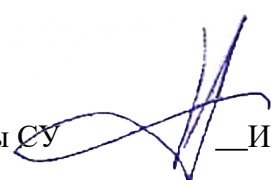
(уровень подготовки – бакалавриат)

направленность (профиль):

«Прикладная информатика в цифровой экономике»

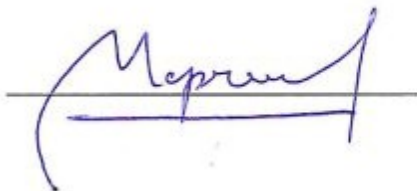
Рабочая программа дисциплины составлена на основании ФГОС ВО направления подготовки 09.03.03 «Прикладная информатика».

Составитель рабочей программы

Старший преподаватель, д.т.н., профессор кафедры СУ  — И.Г. Проценко

Рабочая программа рассмотрена на заседании кафедры «Системы управления»
«20» декабря 2025 г., протокол №5

«20» декабря 2025 г., протокол



Заведующий кафедрой
«Системы управления»
А.А. Марченко

1. ЦЕЛИ И ЗАДАЧИ ИЗУЧЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины «Программирование на Python» является углубленное закрепление студентами теоретических основ и практики применения языка программирования Python при создании веб-приложений и закрепление навыков создания программных средств.

Задача изучения дисциплины «Программирование на Python» состоит в овладении студентами основами теоретических и практических знаний языка программирования Python при создании веб-приложений с алгоритмами нейронных сетей.

Задачами изучения дисциплины «Программирование на Python» являются:

В результате изучения дисциплины студент должен:

- сформировать системное базовое представление, первичные знания о языке программирования Python;

- дать представление о роли и месте программ на языке Python в процессе проектирования информационной системы;

- сформировать умения и навыки работы в среде программирования на языке Python.

Знать: основные понятия и термины, связанные с программированием на языке Python; синтаксис языка Python; основы разработки приложений с применением модулей Python с алгоритмами нейронных сетей.

Уметь: настраивать среду разработки программ и программировать на языке Python, реализовывать программные средства взаимодействия клиентских и серверных частей веб-приложений; создавать программные прототипы решения прикладных задач с использованием языка Python.

Владеть: технологиями создания проектных решений с применением языка программирования Python.

Требования к результатам освоения основных образовательных программ подготовки специалиста

В результате изучения дисциплины у студента должны быть сформированы следующие компетенции:

* способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью (ОПК-4);

* способен разрабатывать алгоритмы и программы, пригодные для практического применения (ОПК-7).

Таблица 1 - Планируемые результаты обучения при изучении дисциплины, соотнесенные с планируемыми результатами освоения образовательной программы.

Код компетенции	Наименование компетенции	Код и наименование индикатора достижения	Планируемый результат обучения по дисциплине	Код показателя освоения
(ОПК-4)	способен участвовать в разработке стандартов, норм и правил, а так же технической документации, связанной с профессиональной деятельностью	ИД _{ОПК-4.1} . Знает основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы.	Знать: – основные понятия информатики и информации, информационных систем, кодирования информации; представление данных и основы алгоритмизации, принципы документирования процессов разработки ИС, критерии качества программ.	З(ОПК-4)1
			Уметь: – составлять алгоритмы решений для прикладных задач, проектировать дружественный интерфейс приложений.	У(ОПК-4)1
			Владеть: – навыками оформления технической документации на различных стадиях жизненного цикла информационной системы.	В(ОПК-4)1
(ОПК-7)	способен разрабатывать алгоритмы и программы, пригодные для практического применения	ИД _{ОПК-7.1} . Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий	Знать: – методы и технологии программирования на языке Python.	З(ОПК-7)1
			Уметь: – создавать программные прототипы решения прикладных задач с использованием языка Python.	У(ОПК-7)1
			Владеть: – технологиями создания проектных решений с применением языка программирования Python.	В(ОПК-7)1

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Курс «Программирование на Python» ориентирован на подготовку бакалавров по направлению 09.03.03 «Прикладная информатика». Дисциплина «Программирование на Python» является обязательной дисциплиной в структуре образовательной программы. Курс позволяет дать будущим бакалаврам теоретические знания и сформировать у них практические навыки применения языка программирования на Python.

2.1. Связь с предшествующими и дисциплинами

Курс дисциплины «Программирование на Python» продолжает цикл дисциплин разработки программных средств, опирается на знания в области информационных технологий и программирования, полученные на дисциплинах «Информатика и программирова-

ние», «Объектно-ориентированное программирование», «Базы данных», «Программная инженерия».

2.2. Связь с последующими дисциплинами

Знания, полученные обучающимися в процессе изучения дисциплины «Программирование на Python», позволят выполнять на высоком уровне лабораторные и практические работы, курсовое и дипломное проектирование. Дисциплина «Программирование на Python» является базовой дисциплиной для технологической (проектно-технологической) практики и производственной (преддипломной) практики и выполнения дипломного проектирования при подготовке итоговой квалификационной работы.

3. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

3.1. Тематический план дисциплины

Наименование разделов и тем	Всего часов	Аудиторные занятия	Контактная работа по видам учебных занятий			Самостоятельная работа	Формы текущего контроля ¹	Итоговый контроль знаний по дисциплине
			Лекции	практические занятия	Лабораторные работы			
1	2	3	4	5	6	7	8	9

Заочная форма обучения								
Тема 1. Базовые конструкции Python	14	2	2		-	12	Опрос	
Тема 2. Коллекции и работа с памятью	18	6	-		6	12	Опрос	
Тема 3. Функции в Python	14	2	-		2	12	Опрос, РЗ	
Тема 4. Объектно-ориентированное программирование	12	-	-		-	12	Опрос, РЗ	
Тема 5. Библиотеки для получения и обработки данных	12	-	-		-	12	Опрос, РЗ	
Тема 6. Работа с базами данных	12	-	-		-	12	Опрос, РЗ	
Тема 7. Использование Python	17	2	-		2	15		
Экзамен	9							9
Всего	108	12	2		10	87		9

¹РЗ – решение задач

3.2. Описание содержания дисциплины

Тема 1. Лекция №1. Базовые конструкции Python

Рассматриваемые вопросы:

Установка Python; синтаксис; программный код; режим интерпретатора; проблема кодировок; вывод ошибок Python.

СРС по теме 1. Подготовка к лекциям; изучение дополнительного теоретического материала; подготовка и прохождение тестирования.

Тема 2. Лекция №2. Коллекции и работа с памятью

Рассматриваемые вопросы:

Основные типы данных; арифметические операции; операции сравнения; логические операции; побитовые операции; операции с текстом; присваивание значений; ввод и вывод данных; операции с числами, строками; форматирование; операторы цикла `while`, `for`; операторы выбора `switch`, `match`; логические операции; операции с текстом; понятие массива; цикл по массиву; ассоциативные массивы; операции с массивами; функции для работы с массивами.

Лабораторная работа №2.1. Установка Python.

Задание: Для выполнения Python и отладки Python-программ необходим набор программных средств. Наиболее простое решение - воспользоваться официальным сайтом. Выбрать в меню Download и скачать последнюю версию Python. Для написания и редактирования Python-программ идеальным решением является IDLE, с помощью которого можно выполнять обычные для интегрированной среды задачи: просматривать, редактировать, запускать, отлаживать программы на Python. Редактор кода использует подсветку синтаксиса. IDLE предлагает дополнительные возможности для опытных пользователей, например, средство просмотра объектов.

Лабораторная работа №2.2. Элементы языка.

Задание: Написать программу, которая содержит не менее 5-и переменных. Присвоить им значения разных типов и вывести тип каждой переменной на экран. Присвоить значения двум переменным. С помощью операторов сравнения проверить, удовлетворяет ли первая переменная следующим условиям:

- равна второй;
- меньше второй;
- меньше или равна второй;
- больше второй.

Лабораторная работа №2.3. Условия, управление потоком

Задание: С помощью инструкции `if` вывести на экран строку "Группа", если переменная `$grup` находится в диапазоне от 1 до 99. Если значение переменной вне этого диапазона, то вывести строку "Кафедра". Сделать то же самое, но если переменная `grup` находится в диапазоне от 21 до 29, то вывести строку "Кафедра ИС". Написать цикл `while`, который будет выводить на экран все нечетные числа от 0 до 50. Сделать то же самое с помощью цикла `for`.

Лабораторная работа №2.4. Массивы.

Задание: Создать массив, содержащий названия кафедр, организованных по факультетам. Пусть это будет ассоциативный массив, в котором имена полей будут кафедрами ("ИС", "СУ", "ВБ" и пр.), а элементами – названия факультетов. Просмотреть созданный массив в цикле, выводя название факультета и связанные с ним кафедры на экран.

СРС по теме 2. Подготовка к лекциям; изучение дополнительного теоретического материала; подготовка теоретического материала и данных для выполнения лабораторных работ; подготовка и прохождение тестирования.

Тема 3. Лекция №3. Функции в Python

Рассматриваемые вопросы:

Создание функции; результат функции; тип аргументов и результата; механизм передачи аргументов; значения аргументов по умолчанию; рекурсия; регулярные выражения; функция eval().

Лабораторная работа №3.1. Функции.

Задание: Создать функцию, которая выводит текст с разным размером шрифта. Функция имеет два аргумента: текстовая строка и размер шрифта. Создать функцию, которая принимает четыре текстовые строки, а выводит html-код таблицы, где каждая из исходных строк расположена в отдельной ячейке.

Лабораторная работа №3.2. Строки

Задание: Проверить, содержит ли адрес электронной почты символ @, и выведите предупреждающее сообщение, если такого символа нет. Создать массив, состоящий из целочисленных и вещественных значений. Считать этот массив в цикле, преобразовывая все элементы в вещественные значения с точностью в два знака после запятой. Вывести значения нового массива в столбец с выравниванием по правому краю.

СРС по теме 3. Подготовка к лекциям; изучение дополнительного теоретического материала; подготовка теоретического материала и данных для выполнения лабораторных работ; подготовка и прохождение тестирования.

Тема 4. Лекция №4. Объектно-ориентированное программирование

Рассматриваемые вопросы:

Создание классов и объектов; наследование; множественное наследование; перегрузка операторов; абстрактные методы; ограничение доступа к атрибутам класса; полиморфизм; динамическая типизация; композиция; статические методы;

Лабораторная работа №4.1. Примеры объектно-ориентированных программ на Python

Задание: необходимо разработать виртуальную модель процесса обучения; в программе должны быть объекты-студенты, преподаватель, кладезь знаний; потребуется три класса – "преподаватель", "студент", "данные"; преподаватель и студент во многом похожи, оба – пользователи информационной системы; значит, их классы могут принадлежать одному надклассу "пользователь".

Однако в контексте данной задачи у преподавателя и студента вряд ли найдутся общие атрибуты. Определим, что должны уметь объекты для решения задачи «увеличить знания»:

- студент должен уметь брать информацию и превращать ее в свои знания;
- преподаватель должен уметь учить группу студентов;
- данные могут представлять собой список знаний.

Элементы будут извлекаться по индексу.

СРС по теме 4. Подготовка к лекциям; изучение дополнительного теоретического материала; подготовка теоретического материала и данных для выполнения лабораторных работ; подготовка и прохождение тестирования.

Тема 5. Лекция №5. Библиотеки для получения и обработки данных

Рассматриваемые вопросы:

Библиотека для работы с табличными данными **Pandas**, которая предоставляет высокоуровневые структуры данных и множество функций для их обработки. Основные структуры данных в *Pandas* - *DataFrame* и *Series*. Библиотека для работы с многомерными массивами и матрицами **NumPy**. Она предоставляет множество математических функций для выполнения высокоуровневых вычислений с массивами данных. **NumPy** используется в связке с *Pandas* и другими библиотеками для анализа данных. Библиотека для создания статических, анимированных и интерактивных визуализаций **Matplotlib**. Она предоставляет широкий спектр инструментов для создания графиков и диаграмм, что делает её незаменимой для визуализации результатов анализа данных. Другие библиотеки (**Seaborn**),

которые помогают выполнять широкий спектр задач, начиная от обработки данных и заканчивая их визуализацией.

Лабораторная работа №5.1. Установка PyCharm к Python

Задание: установить PyCharm, создать новый проект, новый документ в проекте и установить внешние библиотеки в данный проект в PyCharm.

СРС по теме 5. Подготовка к лекциям; изучение дополнительного теоретического материала; подготовка теоретического материала и данных для выполнения лабораторных работ; подготовка и прохождение тестирования.

Тема 6. Лекция №6. Работа с базами данных

Рассматриваемые вопросы:

Введение в базы данных, SQL в Python; настройки для работы с SQL в Python; подключение к базе данных в Python; отправка запросов к базе данных; выборка записей, поля выборки, сортировка; логические операции; вставка, корректировка, удаление записей; формирование и обработка элементов массива.

Лабораторная работа №6.1. Приложение с использованием Python.

Задание: разработать приложения баз данных с использованием языка Python; использовать библиотеки Python для работы с СУБД; выполнять простейшие запросы к БД; изучить принципы взаимодействия языков программирования и СУБД.

СРС по теме 7. Подготовка к лекциям; изучение дополнительного теоретического материала; подготовка теоретического материала и данных для выполнения лабораторных работ; подготовка и прохождение тестирования.

Тема 7. Лекция №7. Использование Python

Рассматриваемые вопросы:

Сценарий в HTML-документе; обработка параметров запроса; использование кнопки; использование нескольких кнопок; использование списков и опций; слайдер и переключатели.

Лабораторная работа №7.1. Математические функции

Задание: создать массив, наполнить его случайными значениями, найти максимальное и минимальное значение и поменять их местами; файл содержит набор целых чисел, отделённых друг от друга произвольным количеством пробелов и переводов строк; для каждого числа, начиная с последнего и заканчивая первым, в отдельной строке вывести его квадратный корень не менее чем с четырьмя знаками после десятичной точки.

Лабораторная работа №7.2. Регулярные выражения.

Задание: составить регулярное выражение для валидации даты в формате YYYY.mm.dd; с помощью регулярного выражения выделить из файла адреса электронной почты; сохранить эти адреса в массиве и вывести на экран.

Лабораторная работа №7.3. Работа с почтой.

Задание: с помощью регулярного выражения выделить из файла адреса электронной почты; сохранить эти адреса в массиве и вывести на экран; прочитать по протоколу IMAP последние 10 заголовков писем и вывести Subject на экран.

Лабораторная работа №7.4. Cookie

Задание: создать форму, с помощью которой пользователь может задать свой ник и выбрать цвет фона страниц сайта; использовать cookie для того, чтобы приветствовать пользователя по имени на следующих страницах с заданным фоном.

Лабораторная работа №7.5. Сессии

Задание: написать программу сохранения персональных настроек пользователя (ник и фон страниц) с использованием функций управления сессией; написать программу, которая применяет функции управления сессией для запоминания того, какие страницы уже посещались пользователем; вывести список ссылок на все посещенные страницы.

Лабораторная работа №7.6. Дата, время

Задание: написать программу отсчета дней до дня рождения, которая должна содержать форму ввода дня, месяца и года и выводить сообщение о том, сколько дней, часов,

минут и секунд осталось до дня рождения; написать программу-календарь для любого месяца между 2020 и 2030 годом; пользователь выбирает месяц и день из выпадающих списков, а даты для выбранного месяца выводятся в виде таблицы-календаря.

Лабораторная работа №7.7. ZIP-Архивы

Задание: Создать архив, добавить файл в архив с диска, создать папку, добавить файл в папку из строки, добавить комментарий к архиву; открыть архив; прочитать содержимое, распаковать все файлы с заданным расширением во временную папку.

СРС по теме 7. Подготовка к лекциям; изучение дополнительного теоретического материала; подготовка теоретического материала и данных для выполнения лабораторных работ; подготовка и прохождение тестирования.

4. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

В целом внеаудиторная самостоятельная работа обучающегося при изучении курса включает в себя следующие виды работ:

- проработка (изучение) материалов лекций;
- чтение и проработка рекомендованной основной и дополнительной литературы;
- подготовка к лабораторным работам;
- поиск и проработка материалов из Интернет-ресурсов, периодической печати;
- выполнение домашних заданий в форме творческих (проблемно-поисковых, групповых) заданий, кейс-стади, докладов;
- подготовка презентаций для иллюстрации докладов;
- выполнение тестовых заданий;
- подготовка к тестированию;
- подготовка к текущему и итоговому (промежуточная аттестация) контролю знаний по дисциплине.

Основная доля самостоятельной работы обучающихся приходится на подготовку к лабораторным работам и тестированию, тематика которых полностью охватывает содержание курса. Самостоятельная работа по подготовке к тестированию и лабораторным работам предполагает умение работать с первичной информацией.

Для проведения практических занятий, для самостоятельной работы используется учебно-методические пособия.

5. РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

5.1. Основная литература

1. Васильев А.Н. Программирование на Python в примерах и задачах. / Алексей Васильев — Издательство Москва — 2021г., УДК 004.438 ББК 32.973.26-018.1 В19 В19 421 стр.
2. Прохоренок Н.А. Python 3 и PyQt 6. Разработка приложений. / Прохоренок Николай Анатольевич — Издательство БХВ — 2023г., 300 стр.
3. Бэрри П. Изучаем программирование на Python. / Бэрри Пол — Издательство Эксмо — 2022г., 224 стр.

5.2. Дополнительная литература

1. Бейдер Д. Чистый Python. Тонкости программирования для профи. / Бейдер Дэн — Издательство Питер — 2024г., 321 стр.
2. Майер К. Однострочники Python: лаконичный и содержательный код. / Майер Кристиан — Издательство Питер — 2022г., 120 стр.

3. Кольцов Д. М. Справочник PYTHON. Кратко, быстро, под рукой. / Дубовик Е. В. — Издательство Наука и техника ISBN 978-5-94387-717-9 — 2023г., 201 стр.

4. Горелик М. Высокопроизводительные Python-приложения. Практическое руководство по эффективному программированию., / Йен Освальд — Издательство ЭКСМО — 2020г., 411 стр.

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине «Программирование на Python» представлен в приложении к рабочей программе дисциплины и включает в себя:

- перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы;
- описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;
- типовые контрольные задания или материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций;
- методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Методика преподавания данной дисциплины предполагает чтение лекций, проведение лабораторных работ, групповых и индивидуальных консультаций по отдельным (наиболее сложным) специфическим проблемам дисциплины. Предусмотрена самостоятельная работа студентов, а также прохождение аттестационных испытаний промежуточной аттестации (экзамен).

Лекции посвящаются рассмотрению наиболее важных концептуальных вопросов: основным понятиям; теоретическим основам. В ходе лекций обучающимся следует готовить конспекты лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины; проверять термины, понятия 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.

На лекциях преподаватель знакомит слушателей с основными понятиями и положениями по текущей теме. На лекциях слушатель получает только основной объём информации по теме. Только посещение лекций является недостаточным для подготовки к лабораторным занятиям и зачету. Требуется также самостоятельная работа по изучению основной и дополнительной литературы и закрепление полученных на лабораторных занятиях навыков.

При изучении дисциплины используются интерактивные методы обучения:

- проблемная лекция, предполагающая изложение материала через неоднозначность трактовки материалов к вопросам, задачам или ситуациям. При этом процесс познания происходит в научном поиске, диалоге и сотрудничестве с преподавателем в процессе анализа и сравнения точек зрения;
- лекция-визуализация - подача материала осуществляется средствами технических средств обучения с кратким комментированием демонстрируемых визуальных материалов (презентаций).

Конкретные методики, модели, методы и инструменты рассматриваются преимущественно при подготовке и выполнении лабораторных работ.

Целью выполнения *лабораторных работ* является закрепление знаний обучающихся, полученных ими в ходе изучения дисциплины на лекциях и самостоятельно. Практические задания по темам выполняются на лабораторных занятиях в компьютерном классе. Если лабораторные занятия пропущены (по уважительной или неуважительной причине), то соответствующие задания необходимо выполнить самостоятельно и представить результаты преподавателю на очередном занятии. Самостоятельная работа студентов – способ активного, целенаправленного приобретения студентом новых для него знаний, умений и навыков без непосредственного участия в этом процесса преподавателя. Качество получаемых студентом знаний напрямую зависит от качества и количества необходимого доступного материала, а также от желания (мотивации) студента их получить. При обучении осуществляется целенаправленный процесс взаимодействия студента и преподавателя для формирования знаний, умений и навыков.

8. КУРСОВОЙ ПРОЕКТ (РАБОТА)

В соответствии с учебным планом курсовое проектирование по дисциплине не предусмотрено.

9. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

9.1. Перечень информационных технологий, используемых при осуществлении образовательного процесса

При освоении дисциплины используются следующие информационные технологии:

- использование слайд-презентаций;
- интерактивное общение с обучающимися и консультирование посредством электронной почты.

9.2. Перечень программного обеспечения, используемого при осуществлении образовательного процесса

При освоении дисциплины используется программное обеспечение:

- пакет Microsoft Office;
- пакет OpenServer;
- текстовые редакторы (notepad++);
- веб-браузеры (Google chrome for Windows).

9.3. Перечень ресурсов сети «Интернет»

При освоении дисциплины используются следующие информационно-справочные системы:

- Электронная библиотека диссертаций РГБ [Электронный ресурс]. - Режим доступа: <http://www.diss.rsl.ru>
- справочно-правовая система Консультант-плюс [Электронный ресурс]. – Режим доступа: <http://www.consultant.ru/online>
- справочно-правовая система Гарант [Электронный ресурс]. – Режим доступа: <http://www.garant.ru/online>

РЕЙТИНГ-ЛИСТ

по дисциплине «Программирование на Python», семестр 4, группа 25ПИБ
преподаватель _____
