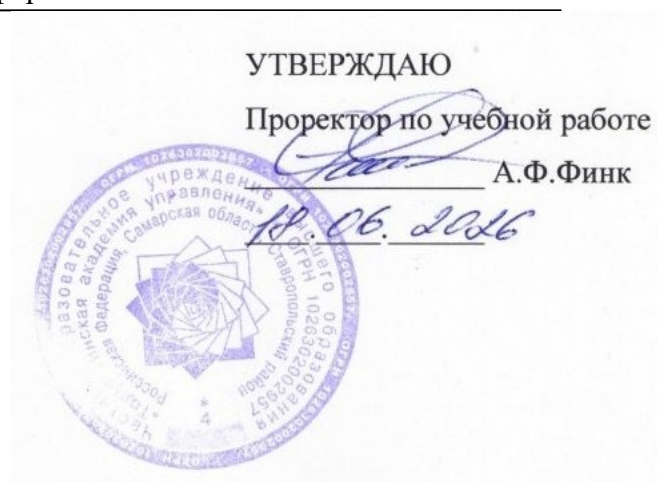


Документ подписан простой электронной подписью
 Информация о владельце:
 ФИО: Финк Анастасия Федоровна
 Должность: Проректор по учебной работе
 Дата подписания: 30.06.2026 13:39:42
 Уникальный программный ключ:
 2431bd5130e74d20a9fc74baab365dd497e3afa3

ЧОУ ВО «Тольяттинская академия управления»

Кафедра

прикладной информатики и высшей математики



Б1.В.ДВ.03.01

РАБОЧАЯ ПРОГРАММА

Учебная дисциплина	<i>Технология разработки объектно-ориентированных приложений на Python</i>
По направлению подготовки	<i>09.03.03 «Прикладная информатика»</i>
Профиль (программа бакалавриата)	<i>«Прикладная информатика в цифровой экономике»</i>
Форма обучения	<i>Очно-заочная</i>

Программа дисциплины рассмотрена (актуализирована) и утверждена на заседании кафедры прикладной информатики и высшей математики

Протокол заседания № 10 от «25» мая 2026 г.

Заведующий кафедрой Стрекалова Наталья Борисовна

1. ОБЩАЯ ИНФОРМАЦИЯ

Рабочая программа дисциплины «Технология разработки объектно-ориентированных приложений на Python» составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования – бакалавриат по направлению подготовки 09.03.03 «Прикладная информатика», утвержденного приказом Министерства образования и науки Российской Федерации 19.09.2017 №922 (с изменениями и дополнениями от 26.11.2020, 08.02.2021, 19.07.2022, 27.02.2023), и учебного плана направления подготовки 09.03.03 «Прикладная информатика», профиль «Прикладная информатика в цифровой экономике» (программа бакалавриата).

Трудоемкость дисциплины: 3 ЗЕТ / 108 академических часа, в том числе 40 часа контактной работы и 68 часов самостоятельной работы обучающихся.

Распределение часов дисциплины по семестрам и видам занятий (по учебному плану)

Вид учебной работы		Количество часов									
		Всего по учебному плану	Семестры								
			1	2	3	4	5	6	7	8	9
Контактная работа (всего):		40				40					
в том числе:											
Лекции		8				8					
Практические занятия		24				24					
Контроль самостоятельной работы (КСР)		8				8					
Самостоятельная работа (всего):		68				68					
Виды промежуточной аттестации (зачет с оценкой)						Зачет с оценкой					
ОБЩАЯ трудоемкость дисциплины:	Часы:	108				108					
	Зач. ед.:	3				3					

2. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Цель дисциплины – усвоение и закрепление основных приемов, методов и принципов разработки программных систем с помощью объектно-ориентированного языка Python и его стандартных модулей, формирование способностей применения высокопроизводительных вычислительных возможностей языка для решения профессиональных задач, в том числе аналитического характера, в различных прикладных областях.

Задачи дисциплины:

- развитие навыков объектно-ориентированного программирования;
- освоение подходов к созданию консольных и визуальных кроссплатформенных программ;

- знакомство с понятиями и языком предметной области, в том числе международной англоязычной терминологией;
- подготовка к сертификации Python-программиста.

3. МЕСТО ДИСЦИПЛИНЫ В УЧЕБНОМ ПЛАНЕ

Учебная дисциплина «Технология разработки объектно-ориентированных приложений на Python» относится к элективным дисциплинам (по выбору) части Блока 1. Дисциплины (модули), формируемой участниками образовательных отношений. Изучение данной дисциплины базируется на материале, изученном в дисциплинах «Алгоритмизация и программирование», «Объектно-ориентированное программирование». Знания, умения и навыки, приобретенные в результате изучения дисциплины, могут быть востребованы при прохождении дисциплины «Проектирование информационных систем», «Цифровой анализ данных», «Основы нейронных сетей», «Экспертные системы», «Системы принятия решений», «Технологии виртуальной и дополненной реальности» и при написании выпускной квалификационной работы.

4. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ ДЛЯ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ ОБУЧАЮЩИХСЯ

Результаты освоения образовательной программы (компетенции обучающихся) устанавливаются в соответствии с федеральным государственным образовательным стандартом по направлению подготовки и профессиональными стандартами, соответствующими профессиональной деятельности выпускников, а также на основе анализа требований работодателей, предъявляемых к выпускникам. Планируемые результаты обучения по дисциплине (знания, умения, навыки) обеспечивают достижение результатов освоения образовательной программы.

Шифр и название компетенции	Индикаторы компетенции	Планируемые результаты обучения по дисциплине
-----------------------------	------------------------	---

ПК-2 Способен проектировать информационные системы	ПК-2.1. Моделирует прикладные процессы предметной области с учетом внедрения сквозных цифровых технологий и передовых ИТ-решений прикладной сферы	знать: - особенности разработки объектно-ориентированных программ и приемы объектно-ориентированного решения задач на языке Python; уметь: - проектировать семейства классов в заданной прикладной области и применять их для решения профессиональных задач на языке Python; владеть: - навыками моделирования и описания моделей семейств классов;
	ПК-2.2. Разрабатывает модель информационной системы, в том числе с опорой на современные облачные решения	Знать: - возможности и преимущества языка Python для проектирования информационных систем разного назначения; - основные приемы создания компонентов информационных систем для обеспечения и анализа прикладных процессов Уметь: - проектировать программное обеспечение разного назначения (вычислительные системы, программы визуализации данных и т.п.) с применением библиотек языка Python (NumPy, Tkinter);. Владеть: - навыками анализа и выбора эффективных способов и средств разработки программных модулей информационных систем на языке Python для решения аналитических задач.
ПК-3 Способен разрабатывать информационные системы	ПК-3.1. Осуществляет кодирование на современных языках программирования, в том числе используя возможности специализированных цифровых платформ для индивидуальной и совместной разработки информационных систем	знать: - синтаксис, типы данных и операторы языка Python, парадигму программ и подпрограмм; уметь: - разрабатывать объектно-ориентированные программы, используя классы и модули из библиотек языка Python для решения аналитических задач. владеть: - навыками применения объектно-ориентированной технологии и стандартных библиотек языка Python для решения задач вычислительного и аналитического характера.

	ПК-3.2. Проводит тестирование компонентов информационных систем, в том числе используя возможности специализированных цифровых платформ	Знать: - основные приемы тестирования компонентов информационных систем на языке Python; Уметь: - составлять блоки тестов компонентов информационных систем на языке Python Владеть: - навыками применения тестов для информационных систем на языке Python
--	--	---

5. СТРУКТУРА И ОБЪЕМ ДИСЦИПЛИНЫ

Семестр изучения: 4

Раздел, модуль	Подраздел, тема	Виды учебной работы					Промежуточная аттестация в часах	Форма текущего контроля	Формируемые компетенции
		Контактная работа (в часах)			Самостоятельная работа				
		Лекции	Практические занятия	КСР (пропорционально темам)	в часах	формы организации самостоятельной работы			
	Тема 1. Возможности языка Python. Первая программа. Знакомство со средой разработки IDLE.	2	2		10	Повторение лекционного материала, подготовка к практическим занятиям		Устный опрос	ПК-3.1, ПК-3.2
	Тема 2. Основы Python	2	2		10	Повторение лекционного материала, подготовка к практическим занятиям		Устный опрос	ПК-3.1, ПК-3.2
	Тема 3. Разработка классов в языке Python.	1	4		12	Выполнение практического задания		Проверка выполненных работ	ПК-2.1, ПК-2.2
	Тема 4. Наследование и интерфейсы в языке Python. Документирование кода. Создание и подключение модулей	1	4		12	Выполнение практического задания		Проверка выполненных работ	ПК-2.1, ПК-2.2, ПК-3.1, ПК-3.2
	Тема 5. Обобщённые типы и коллекции значений в языке Python.	1	6		12	Выполнение практического задания		Проверка выполненных работ	ПК-2.1, ПК-2.2, ПК-3.1, ПК-3.2
	Тема 6. Работа со строками в языке Python.	1	6		12	Выполнение практического задания		Проверка выполненных работ	ПК-3.1, ПК-3.2

Форма промежуточной аттестации (зачет с оценкой)	-	-	-	-	Подготовка к промежуточной аттестации		-	-
Всего	8	24	8	68	-			
	108							

6. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Тема 1. Возможности языка Python. Первая программа. Знакомство со средой разработки IDLE

Структура языка Python. Установка и настройка IDLE Python. Синтаксис языка Python для основных алгоритмических конструкций, литералов, выражений. Описание встроенных типов данных, а также некоторые вопросы общепринятого в Python стиля программирования.

Тема 2 Основы Python

Условный оператор if. Циклы. Ключевые слова, встроенные функции. Числа. Строки. Списки (массивы). Создание списков, массивов, кортежей, словарей. Понятие генератора списков. Индексы и срезы. Методы с переменным числом аргументов. Множества. Функции.

Тема 3 Разработка классов в языке Python.

Инкапсуляция при разработке классов Python. Моделирование задачи с использованием классов Python. неизменяемые классы. Подклассы: создание и использование. Перегрузка методов класса. Использование абстрактных классов. Вложенные классы.

Тема 4 Наследование и интерфейсы в языке Python. Документирование кода. Создание и подключение модулей

Интерфейсы в Python, определение интерфейсов. Особенности использования интерфейсов и классов в программах. Расширение интерфейсов. Рефакторинг кода.

Тема 5 Обобщённые типы и коллекции значений в языке Python.

Обобщённые типы как способ создания классов в Python. Создание объектов в рамках обобщённого типа. Создание коллекций без использования обобщённых типов и с их использованием. Работа со структурами данных. Реализация стека и очереди. Перечисляемые типы.

Тема 6 Работа со строками в языке Python.

Чтение данных из командной строки. Поиск строк. Парсинг строк. Создание строк с использованием класса StringBuilder. Поиск в строке, парсинг строки и удаление строк с использованием регулярных выражений.

7. САМОСТОЯТЕЛЬНАЯ РАБОТА ОБУЧАЮЩИХСЯ

В рамках данной учебной дисциплины обучающиеся выполняют самостоятельную работу в виде самостоятельного изучения тем дисциплины, повторения лекционного материала и подготовки к практическим занятиям по темам 1-2, выполнение практического задания «Построение движения робота по заданной траектории» по темам 3-6.

8. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При изучении данной дисциплины используются следующие образовательные технологии:

Тема занятия	Вид образовательной технологии	Форма проведения занятия
Тема 1. Возможности языка Python. Первая программа. Знакомство со средой разработки IDLE.	Традиционная технология	Лекция
		Практическое занятие
Тема 2. Основы Python	Традиционная технология	Лекция
		Практическое занятие
Тема 3. Разработка классов в языке Python.	Традиционная технология	Лекция
		Практическое занятие
Тема 4. Наследование и интерфейсы в языке Python. Документирование кода. Создание и подключение модулей	Традиционная технология	Лекция
		Практическое занятие
Тема 5. Обобщённые типы и коллекции значений в языке Python.	Традиционная технология	Лекция
		Практическое занятие
Тема 6. Работа со строками в языке Python.	Традиционная технология	Лекция
		Практическое занятие

9. ТЕКУЩИЙ КОНТРОЛЬ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО ДИСЦИПЛИНЕ

9.1. Формы контроля по дисциплине

Текущий контроль. В процессе изучения учебной дисциплины обучающиеся участвуют в устных теоретических опросах, выполняют практические задания. Результаты выполнения практических заданий являются основанием для выставления оценок текущего контроля по данной учебной дисциплине. Выполнение всех практических заданий является обязательным для всех обучающихся. Обучающиеся, не выполнившие в полном объеме все задания, не допускаются к сдаче зачета с оценкой по данной учебной дисциплине.

Промежуточная аттестация. Для контроля усвоения обучающимися данной дисциплины учебным планом предусмотрен зачет с оценкой, который проводится в форме устного ответа на вопрос и выполнения практического задания.

9.2. Оценочные материалы (оценочные средства) для текущего контроля успеваемости и промежуточной аттестации по дисциплине

Текущий контроль.

Список примерных вопросов к устному опросу.

1. Каковы менее известные, но полезные функции языка программирования Python?
2. Встроенный тип str. Методы объекта str. Print() и форматирование вывода.
3. Работа с файлами. Методы open(), close(), read(), write().
4. Встроенные типы последовательностей list, tuple, range и их методы.
5. Встроенный объект dict и его методы.
6. Встроенные типы чисел — int, float, complex. Машинное представление чисел с плавающей точкой и целых. Преобразование типов при сравнении чисел.
7. Рациональные числа. Модуль fractions.
8. Двоичное представление чисел. Неассоциативность операций в арифметике с плавающей запятой. Целые числа с произвольной точностью.
9. Множества. Встроенные типы set и frozenset.
10. Инструкции и синтаксис. Составные конструкции и обработка исключений
11. Инструкции if/else/elif, логические операторы и выражения сравнения
12. Циклы while и for в Python
13. Функции в Python. Основные понятия
14. Области видимости и пространство имен в Python.
15. Передача аргументов в функцию. Специальные режимы сопоставления аргументов.
16. Парадигма объектно-ориентированного программирования. Поддержка в Python функционального программирования.
17. Объекты. Динамическая типизация. Инкапсуляция.
18. Генерация объекта class. Новое пространство имен. Объект экземпляр класса.
19. Атрибуты класса. Атрибуты данных. Атрибуты-методы. Параметр self. Добавление атрибутов к классу во время исполнения программы.
20. Специальные методы и атрибуты классов. Методы init () и del () в Python. Декораторы функций и декораторы классов. Инструменты интроспекции в Python. Метаклассы.
21. Абстрактные методы в Python. Классические классы и классы нового стиля.
22. Наследование. Базовый и производный класс. Построение производного класса.
23. Порождающие функции (функции-фабрики). Множественное наследование. Примеси (Mix-in)
24. Агрегация. Контейнеры. Иерархия наследования.
25. Полиморфизм. Подмена методов в производном классе. Доступ к методам базового класса.
26. Обработка исключений. Инструкция try... except... else... finally. Объект Менеджер контекста и конструкция with...as. Классы встроенных исключений.

27. Генераторы Python. Специальный тип функций – generator function. Объект-итератор. Итерация и потоки данных. (Data Flow Programming). Объект-функтор.
28. Модули и пакеты. Библиотеки сторонних разработчиков.
29. Модули и пакеты. Графический интерфейс.
30. Взаимодействие Python с Интернетом. Структура и функционирование сети Интернет. Архитектура клиент-сервер. Пакетная передача данных
31. Взаимодействие Python с Интернетом. Unicode и строки байтов. Разбор URL-адреса. Кодирование и декодирование строки запроса. Преобразование относительного URL-адреса в абсолютный
32. Сжатие и распаковка данных. Работа с датой и временем.

Промежуточная аттестация.

Список вопросов для подготовки к зачету с оценкой

1. Особенности языка и платформы Python.
2. Классификация программ по типу исполнения (компилируемые, интерпретируемые, исполняемые на виртуальных машинах). Виртуальная машина Python. JIT-компиляция.
3. Создание простейшей программы на Python, её компиляция в байт-код и запуск.
4. Средства разработки Python-приложений. Интегрированные среды разработки.
5. Встроенные типы данных. Способы задания литералов различных типов.
6. Хранение данных в памяти ЭВМ.
7. Приведение типов (явное и автоматическое). Константы и переменные.
8. Типы Python-приложений, их особенности.
9. Типы данных в языке Python: простые и ссылочные типы.
10. Массивы в Python: массивы простых типов и массивы объектов.
11. Оператор присваивания. Порядок действий (приоритет операторов).
12. Арифметические операторы. Операторы инкремента и декремента.
13. Встроенный класс Math. Псевдослучайные числа.
14. Операторы сравнения и логические операторы.
15. Операторы ветвления. Условный оператор. Минимизация количества проверок.
16. Операторы ветвления. Оператор множественного выбора. Его сравнение с условным оператором.
17. Встроенный класс String. Строковые операции.
18. Стандартные потоки ввода-вывода. Организация ввода и вывода данных. Класс Scanner.
19. Классы в языке Python: особенности реализации, определение класса.

20. Классы в языке Python: управление доступом к элементам класса; понятие пакета.
21. Классы в языке Python: поля класса.
22. Классы в языке Python: методы, конструкторы.
23. Интерфейсы в языке Python: определение интерфейса, реализация интерфейса.
24. Понятие Python API. Основные пакеты Python API.
25. События в Python: понятие события; типы событий; иерархия классов событий.
26. События в Python: модель делегирования событий.
27. События в Python: интерфейсы блоков прослушивания событий; способы реализации блока прослушивания.
28. События в Python: использование внутренних классов, классов-адаптеров для упрощения обработки событий.
29. Исключения: понятие исключения; классы исключений; необходимость обработки исключений.
30. Исключения: операторы языка Python, используемые для обработки исключений.
31. Исключения: организация обработки исключений; определение собственных исключений.
32. Потоки вычислений: понятия процесса, потока.
33. Ввод/вывод в Python: основные понятия.
34. Ввод/вывод в Python: основные группы классов и интерфейсов пакета Python Tkinter.

Порядок проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине для инвалидов и лиц с ОВЗ предусмотрен Положением о текущем контроле успеваемости и промежуточной аттестации обучающихся.

10. РЕСУРСНАЯ СОСТАВЛЯЮЩАЯ

Для проведения занятий лекционного типа по данной дисциплине используются аудитории с медиа-оборудованием (проектор, экран, ноутбук) и учебной мебелью.

Для проведения занятий семинарского типа (практических занятий) по данной дисциплине используются компьютерные классы, оснащенные компьютерами с доступом в Интернет и необходимым программным обеспечением.

Для самостоятельной работы обучающихся используется помещение для самостоятельной работы обучающихся, оснащенное компьютерами с необходимым программным обеспечением и доступом в Интернет и электронную информационно-образовательную среду вуза.

Для проведения контроля самостоятельной работы по данной дисциплине используются компьютерные классы, оснащенные компьютерами с доступом в Интернет и необходимым программным обеспечением.

Для проведения текущего контроля успеваемости и промежуточной аттестации по данной дисциплине используются компьютерные классы, оснащенные компьютерами с доступом в Интернет и необходимым программным обеспечением.

Перечень лицензионного программного обеспечения

№ п/п	Наименование	Тип ресурса
1	Microsoft Windows	Сублицензионный договор АО «СофтЛайн Трейд» № /131 от 10.07.2020. Период действия договора и лицензий бессрочный. Лицензионное соглашение Microsoft - Open Value Subscription для решений Education Solutions №V8265046
2	Microsoft Office	
5	Антивирус Касперского отечественного производства	Сублицензионный договор АО «СофтЛайн Трейд» №Tr000947217 от 04.12.2025 срок действия 10.02.2026 - 11.02.2028. Тип лицензии - проприетарная (17E0-251210-090335-980-xxxx)

Перечень свободно распространяемого программного обеспечения

1. Браузеры Yandex, Google Chrome;
2. IDLE Python – интегрированная среда разработки и обучения на языке Python

В соответствии с Положением о создании специальных условий для инвалидов и лиц с ОВЗ информационно-технологическая база образовательного процесса предусматривает использование материально-технических средств с учетом различных нозологий инвалидов и лиц с ОВЗ.

11. ЛИТЕРАТУРА

11.1. Основная литература

№ п/п	Библиографическое описание	Тип издания	Количество в библиотеке
1.	Гуриков, С. Р. Основы алгоритмизации и программирования на Python : учеб. пособие / С. Р. Гуриков. - Москва : Инфра-М, 2025. - 342 с. - ISBN 978-5-16-102278-8 (online). - ЭБС Знаниум. URL: https://znanium.ru/catalog/document?id=453296	учебное пособие	ЭБС
2.	Шевченко, Л. Г. Программирование на Python в среде IDLE : учеб. пособие / Л. Г. Шевченко, Т. В. Дружинина. - Новосибирск : Изд-во НГТУ, 2020. - 194 с. - ISBN 978-5-7782-4215-9. - ЭБС Знаниум. - URL: https://znanium.ru/catalog/document?id=396958	учебное пособие	ЭБС

11.2 Дополнительная литература

1. Жуков, Р. А. Язык программирования Python : практикум : учеб. пособие / Р. А. Жуков. - Москва : Инфра-М, 2026. - 215 с. - ISBN 978-5-16-107207-3 (online). - ЭБС Знаниум. - URL: <https://znanium.ru/catalog/document?id=477700>
2. Карякин, М. И. Технологии программирования и компьютерный практикум на языке Python : учеб. пособие / М. И. Карякин, К. А. Ватульян, Р. М. Мнухин. - Ростов-на-Дону; Таганрог : Издательство Южного федерального университета, 2022. - 240 с. - ISBN 978-5-9275-4108-9 (online). - ЭБС Знаниум. - URL: <https://znanium.ru/catalog/document?pid=2057604>
3. Комлев, Н. Ю. Объектно Ориентированное Программирование. Хорошая книга для Хороших Людей / Н. Ю. Комлев. - Москва : СОЛОН-Пресс, 2022. - 297 с. - ISBN 978-5-91359-276-7. - ЭБС Знаниум. - URL: <https://znanium.ru/catalog/document?id=456749>
4. Колдаев, В. Д. Структуры и алгоритмы обработки данных : учеб. пособие / В. Д. Колдаев. - Москва : Инфра-М, 2021. - 296 с. - ISBN 978-5-16-101275-8 (online). - ЭБС Знаниум. - URL: <https://znanium.ru/catalog/document?id=398591>
5. Шишкин, В. В. Разработка логических компьютерных игр с графическим интерфейсом в среде Питон : учебное пособие / В. В. Шишкин, Д. С. Афонин. - Ульяновск : УлГТУ, 2023. - 88 с. - ISBN 978-5-9795-2339-2. - Цифровая библиотека IPRsmart . - URL: <https://www.iprbookshop.ru/149291.htm>

Периодические издания:

1. Открытые системы. СУБД : журнал. – УБД ИВИС. – URL: <https://eivis.ru/browse/publication/64072>
2. Программные продукты и системы: международный научно-практический журнал. – ЭБС Знаниум. - URL : <https://znanium.ru/catalog/magazines/issues?ref=f9bfbfd0e-239e-11e4-99c7-90b11c31de4c>

11.3. Современные профессиональные базы данных, информационные справочные системы, электронные библиотечные системы:

1. Znanium: электронно-библиотечная система. – URL: <http://znanium.ru>.
2. eLIBRARY.RU: научная электронная библиотека. – URL: <https://elibrary.ru/>.
3. Цифровая библиотека IPRsmart. – URL: <https://www.iprbookshop.ru/>.
4. Консультант плюс: справочно-правовая система. – URL: T:\consultantplus\cons.exe.
5. УБД ИВИС. – URL : <https://eivis.ru/basic/details>.
6. Электронная библиотека TAY. – URL: <http://83.234.207.58/MarcWeb2/Default.asp>.

12. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Успешное освоение дисциплины «Технология разработки объектно-ориентированных приложений на Python» предполагает посещение лекционных и практических занятий, самостоятельную работу над изучением дополнительного теоретического материала и возможностей различных операционных систем. Рекомендуется перед каждым занятием повторять материал предыдущей лекции, изучать документы, источники основной и дополнительной литературы по пройденной теме, структурировать изученный. Такая работа с учебным материалом позволит закрепить знания, более эффективно подготовиться к устным опросам и сдаче зачёта с оценкой по дисциплине, а также выявить «проблемные» места в знаниях и подготовить вопросы к преподавателю.

При выполнении практических заданий по темам 3-6 обучающимся следует обратиться к материалам и опыту программирования на различных языках. Первоначальные практические навыки работы программирования на Python обучающиеся получают на практических занятиях в аудитории с преподавателем. Для самостоятельной отработки этих навыков рекомендуется использовать бесплатные IDLE, либо воспользоваться аудиториями Академии, специально отведенными для самостоятельной работы. Обучающиеся могут повторять уже знакомые алгоритмы выполнения работ по программированию на Python и анализировать полученные результаты.

Все задания и теоретические материалы размещаются в локальной сети и электронной

информационно-образовательной среде Академии.

Выполненные практические работы сдаются преподавателю непосредственно в аудитории либо одним из следующих способов: сохранение в электронной информационно-образовательной среде, отправка преподавателю на почтовый ящик. При отправке преподавателю выполненной работы, по почте обучающемуся следует обеспечить личную идентификацию. Как правило, в теме или тексте письма указывается курс, ФИО обучающегося, дисциплина, тема, по которой выполнена работы. Некоторые практические задания не могут быть сделаны только в рамках выделенного объема контактной работы (в аудитории) и «доделываются» в часы самостоятельной работы. Сдача таких работ на проверку осуществляется теми же самыми способами, что и по окончании практических занятий. Результаты проверки выполненных работ доводятся до сведения обучающегося во время аудиторных занятий и/или в часы КСР, а также размещаются в электронной информационно-образовательной среде.

Консультации по выполнению практических и самостоятельных работ, обсуждение отметок и допущенных ошибок осуществляется во время КСР на кафедре прикладной информатики или в аудитории по расписанию. Консультации могут осуществляться также посредством асинхронного (почта, ЭИОС) и синхронного (zoom) коммуникационного взаимодействия по предварительной договоренности с преподавателем.

Формой промежуточного контроля по дисциплине выступает зачет с оценкой, который проводится в форме устного ответа на теоретический вопрос и выполнения практического задания. Критерии допуска к зачету с оценкой и условия его сдачи озвучиваются преподавателем на первых занятиях по дисциплине.

13. ЛИСТ СОГЛАСОВАНИЯ

Составил:

Н.А. Иванова, ст.преп.


(подпись)

Заведующий кафедрой

Н.Б. Стрекалова, д.п.н., доцент


(подпись)

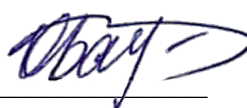
Заведующий выпускающей кафедрой

Н.Б. Стрекалова, д.п.н., доцент


(подпись)

Директор БИК

О.В. Балакина


(подпись)

Начальник ООУП

С.В. Фирсова


(подпись)