

**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тамбовский государственный университет имени Г.Р. Державина»**

Институт дополнительного образования и профессиональной переподготовки

УТВЕРЖДАЮ
Проректор по образованию
ФГБОУ ВО «Тамбовский государственный
университет имени Г.Р. Державина»


Я.Ю. Радюкова
« » 2024г.

**ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА
ПРОФЕССИОНАЛЬНОЙ ПЕРЕПОДГОТОВКИ**

«ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ PYTHON»

Вид профессиональной деятельности: Написание программного кода с использованием языков программирования, определения и манипулирования данными

Документ: диплом о профессиональной переподготовке с присвоением квалификации «Программист»

Объем: 432 часа / 12 зачетных единиц

Составитель:

Забродский И.А., ассистент кафедры математического моделирования и информационных технологий

Эксперты – Седова Н.В., к.п.н., доцент кафедры математического моделирования и информационных технологий;

Рабочая программа обсуждена на кафедре математического моделирования и информационных технологий 13.09. 2024 г. (протокол заседания №2).

I. ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1. Нормативные правовые основания разработки программы

Нормативную правовую основу разработки программы составляют:

1. Федеральный закон от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации»
2. Постановление Правительства Российской Федерации от 22 января 2013 г. № 23 «О Правилах разработки, утверждения и применения профессиональных стандартов»
3. Приказ Минтруда России от 12 апреля 2013 г. № 148н «Об утверждении уровней квалификаций в целях разработки проектов профессиональных стандартов»
4. Приказ Минобрнауки России от 1 июля 2013 г. № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам»
5. Приказ Минобрнауки России от 14.06.2013 № 464 (ред. от 28.08.2020) «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам среднего профессионального образования» (Зарегистрировано в Минюсте России 30.07.2013 N 29200);
6. Приказ Министерства здравоохранения и социального развития РФ от 26.08.2010 г. № 761 н. «Об утверждении Единого квалификационного справочника должностей руководителей, специалистов и служащих. Раздел «Квалификационные характеристики должностей работников образования»
7. Приказ Министерства труда и социальной защиты РФ от 08 октября 2015 года № 544н
8. Приказ Министерства труда и социальной защиты РФ от 05 августа 2016 года № 422н

Программа разработана с учетом профессионального стандарта (квалификационных требований): «Программист» (утв. приказом Министерства труда РФ от 20.07.2022 N 424н).

1.2. Требования к слушателям: наличие среднего профессионального образования или высшего образования

1.3. Формы освоения программы: очно, с применением дистанционных образовательных технологий.

1.4. Цель и планируемые результаты обучения

Цель: обеспечить подготовку в соответствии с государственным образовательным стандартом высшего профессионального образования, а приобретение системных знаний для выполнения нового вида профессиональной деятельности в сфере написания программного кода с использованием языков программирования, определения и манипулирования данными.

Характеристика профессиональной деятельности:

Слушатель готовится к выполнению деятельности, связанной созданием, поддержкой и доработкой различных цифровых программ.

Планируемые результаты обучения

Слушатель, освоивший данную программу, должен обладать:

профессиональными компетенциями, соответствующими виду (видам) профессиональной деятельности, на который (которые) ориентирована программа: применять языки программирования для решения профессиональных задач (ПК-1);

В результате освоения учебной дисциплины обучающийся должен **уметь**:

- применять язык программирования Python для написания программного кода;
- использовать возможности имеющейся технической и/или программной архитектуры;
- использовать среду программирования Python;
- использовать средства системы управления базами данных;

В результате освоения программы обучающийся должен **знать**:

- синтаксис языка программирования Python, особенности программирования на этом языке,
- стандартные библиотеки языка программирования;
- технологии программирования;
- компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними;
- особенности среды программирования Python;

Трудовая функция (по ПС)	Профессиональные компетенции	Практический опыт	Умения	Знания
1	2	3	4	5
Разработка и отладка программного кода	ПК-1: Применяет языки программирования для решения профессиональных задач. (Применяет язык программирования Python для решения профессиональных задач).	Владеть навыками программирования на языке Python; навыками работы с разными форматами файлов с данными; навыками поиска необходимых для программирования на языке Python данных; навыками сбора и обработки данных на языке программирования Python.	Уметь применять язык программирования Python для написания программного кода; использовать возможности имеющейся технической и/или программной архитектуры; использовать среду программирования Python.	Знать синтаксис языка программирования Python, особенности программирования на этом языке, стандартные библиотеки языка программирования; технологии программирования; компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними; особенности среды программирования Python.

1.5. Трудоемкость программы: 432 часа

II. УЧЕБНЫЙ ПЛАН

№ п/п	Наименование учебных курсов, дисциплин (модулей), практик	Формы промежуточной аттестации	Обязательные учебные занятия		Самостоятельная работа обучающегося	Всего (час.)
			Всего (час.)	в т.ч. лабораторные и практические занятия (час.)		
1	2	3	4	5	6	7
1	Модуль 1. Программирование на языке Python. Базовый курс	Зачет	36	22	36	72
1.1	Введение в Python и установка окружения		4	2	6	10
1.2	Переменные и типы данных		6	4	6	12
1.3	Операторы и выражения		6	4	6	12
1.4	Условные конструкции (if, else, elif)		6	4	6	12
1.5	Циклы в Python (for, while)		6	4	6	12
1.6	Функции и структуры программ		6	4	6	12
1.7	Промежуточная аттестация		2			2
2	Модуль 2. Программирование на языке Python. Продвинутый курс	Зачет	36	22	36	72
2.1	Списки и кортежи		4	2	6	10
2.2	Множества и словари		6	4	6	12
2.3	Срезы, генераторы списков и словарей		6	4	6	12
2.4	Работа с файлами (чтение и запись)		6	4	6	12
2.5	Исключения и их обработка		6	4	6	12
2.6	Итераторы и генераторы		6	4	6	12
2.7	Промежуточная аттестация		2			2
3.	Модуль 3. Программирование на языке Python. Разработка веб- приложений с использованием Flask	Зачет	36	22	36	72

3.1	Введение в Flask		4	2	6	10
3.2	Работа с формами и обработка POST-запросов		6	4	6	12
3.3	Добавление и удаление задач		6	4	6	12
3.4	Расширение функциональности: фильтрация и изменение статуса задач		6	4	6	12
3.5	Редактирование задач		6	4	6	12
3.6	Рефакторинг кода и отображение статистики		6	4	6	12
3.7	Промежуточная аттестация		2			2
4.	Модуль 4. Программирование на языке Python. ООП базовый курс	Зачет	36	22	36	72
4.1	Введение в объектно-ориентированное программирование		4	2	6	10
4.2	Атрибуты и методы классов		6	4	6	12
4.3	Конструктор класса (метод init)		6	4	6	12
4.4	Инкапсуляция и методы доступа (getter, setter)		6	4	6	12
4.5	Наследование классов		6	4	6	12
4.6	Полиморфизм и переопределение методов		6	4	6	12
4.7	Промежуточная аттестация		2			2
5	Модуль 5. Программирование на языке Python. ООП продвинутый курс	Зачет	36	22	36	72
5.1	Абстрактные классы и методы		4	2	6	10
5.2	Множественное наследование		6	4	6	12
5.3	Декораторы классов и методов		6	4	6	12
5.4	Статические и классовые методы		6	4	6	12
5.5	Специальные методы (магические методы), часть 1		6	4	6	12
5.6	Специальные методы		6	4	6	12

	(магические методы), часть 2					
5.7	Промежуточная аттестация		2			2
6	Модуль 6. Модули и библиотеки	Зачет	36	20	36	70
6.1	Введение в библиотеку math		4	2	6	10
6.2	Работа с библиотекой numpy		4	2	6	12
6.3	Введение в библиотеку pandas		6	4	6	12
6.4	Работа с HTTP- запросами с использованием библиотеки requests		6	4	6	12
6.5	Обработка и анализ данных с использованием pandas и numpy		6	4	6	12
6.6	Визуализация данных (с использованием matplotlib и pandas)		6	4	6	12
6.7	Промежуточная аттестация		2			2
7.	Итоговая аттестация	Экзамен	2			2
Всего по программе:			216	130	216	432

III. СОДЕРЖАНИЕ ПРОГРАММЫ

Содержание модуля «Программирование на языке Python. Базовый курс»

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем часов
1	2	3
Модуль 1		
Тема 1. Введение в Python и установка окружения	Содержание учебного материала Дается общее представление о Python, его особенностях и области применения. Рассматривается установка интерпретатора Python, настройка среды разработки (IDE) и запуск первых программ. Изучаются различные способы запуска скриптов через командную строку и интерактивную оболочку. Информационные (лекционные) занятия Практические занятия: Установка Python и настройка окружения разработки, запуск первой программы в различных режимах. Самостоятельная работа: Изучение документации Python, выполнение простых примеров с использованием разных подходов запуска программ.	Уровень освоения 1 2 2 6
Тема 2. Переменные и типы данных	Содержание учебного материала Изучаются основные типы данных Python, такие как числа, строки и логические значения. Рассматриваются способы объявления переменных, их использование и преобразование между типами. Анализируются встроенные функции для работы с типами данных и методы строк. Информационные (лекционные) занятия Практические занятия: Работа с различными типами данных, изучение преобразования типов и базовых операций с ними. Самостоятельная работа: Ознакомление с методами строк и числами с плавающей точкой, выполнение заданий на конкатенацию и округление.	Уровень освоения 2 2 4 6
Тема 3. Операторы и выражения	Содержание учебного материала Рассматриваются арифметические и логические операторы, их применение	Уровень освоения 2 12

	для создания выражений. Изучается порядок выполнения операторов, а также операторы сравнения. Анализируются особенности комбинирования выражений и использование скобок для управления приоритетом выполнения.		
	Информационные (лекционные) занятия		2
	Практические занятия: Применение арифметических и логических операторов для решения задач, связанных с вычислениями и условиями.		4
	Самостоятельная работа: Практика построения сложных логических выражений и решение задач, связанных с проверкой условий и сравнением значений.		6
Тема 4. Условные конструкции (if, else, elif)	Содержание учебного материала	Уровень освоения	12
	Изучаются условные операторы if, else и elif для создания логических ветвлений в программах. Рассматриваются принципы построения вложенных условий и работа с логическими операторами для создания сложных проверок.	2	
	Информационные (лекционные) занятия		2
	Практические занятия: Написание программ с использованием условных операторов, решение задач на проверку различных условий.		4
	Самостоятельная работа: Изучение использования вложенных условий и логических операторов для построения более сложных логических конструкций.		6
Тема 5. Циклы Python (for, while)	Содержание учебного материала	Уровень освоения	12
	Рассматриваются циклы for и while для выполнения повторяющихся операций. Изучаются операторы управления циклами (break, continue), а также вложенные циклы для выполнения многократных операций.	2	
	Информационные (лекционные) занятия		2
	Практические занятия, стажировка: Применение циклов for и while для решения задач, требующих повторения действий, использование операторов break и continue.		4
	Самостоятельная работа: Практика создания вложенных циклов и задач с подсчетом значений и генерацией последовательностей.		6

Тема 6. Функции и структуры программ	Содержание учебного материала		Уровень освоения	12
	Изучаются функции в Python, включая определение функций, передача аргументов и возврат значений. Рассматриваются встроенные функции и создание пользовательских функций для организации кода.		2	
	Информационные (лекционные) занятия			2
	Практические занятия, стажировка: Создание и использование функций для разделения программы на логические блоки, работа с параметрами и возвратом значений.			4
	Самостоятельная работа: Изучение использования встроенных и пользовательских функций для оптимизации кода и выполнения повторяющихся задач.			6
Промежуточная аттестация				2
	Всего:			72

Содержание модуля «Программирование на языке Python. Продвинутый курс»			
Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Уровень освоения	Объем часов
1	2		3
Модуль 2			
Тема 1. Списки и кортежи	Содержание учебного материала Рассматриваются списки и кортежи как основные структуры данных в Python. Изучаются методы добавления, удаления, сортировки элементов и другие встроенные функции. Анализируются отличия списков и кортежей, их применение в разных сценариях.	1	10
Тема 2. Множества и словари	Информационные (лекционные) занятия		2
	Практические занятия:		2
	Работа с операциями изменения и сортировки элементов в списках и кортежах.		6
	Самостоятельная работа:		
	Реализация задач по созданию и обработке вложенных списков и кортежей.		
	Содержание учебного материала	Уровень освоения	12
	Рассматриваются множества и словари как структуры для хранения	2	

Тема 3. Срезы, генераторы списков и словарей	уникальных данных и пар ключ-значение. Изучаются методы добавления, удаления элементов, пересечения и объединения множеств. Анализируются способы работы со словарями, включая доступ к значениям, обновление данных и проверку наличия ключей.		
	Информационные (лекционные) занятия		2
	Практические занятия: Создание и использование множеств и словарей для выполнения операций с данными, включая пересечение и фильтрацию.		4
	Самостоятельная работа: Реализация задач по обработке и обновлению больших словарей и множеств с использованием встроенных методов.		6
	Содержание учебного материала	Уровень освоения	12
	Рассматриваются срезы для извлечения подмножеств из коллекций данных.	2	
	Изучаются генераторы списков, множеств и словарей для создания коллекций с использованием циклов и условий. Рассматриваются примеры создания сложных структур данных через генераторы.		
	Информационные (лекционные) занятия		2
	Практические занятия: Применение срезов и генераторов для обработки данных, создание новых коллекций на основе существующих.		4
	Самостоятельная работа: Изучение вложенных генераторов и практика их использования для создания сложных структур данных.		6
Тема 4. Работа с файлами (чтение и запись)	Содержание учебного материала	Уровень освоения	12
	Изучаются методы работы с файлами, такие как открытие, чтение, запись и закрытие файлов. Рассматриваются режимы работы с файлами (текстовые и бинарные) и обработка исключений при работе с файлами. Анализируется создание и обработка больших файлов с использованием буферного чтения.	2	
	Информационные (лекционные) занятия		2
	Практические занятия: Чтение данных из файла, их обработка и запись результата в новый файл.		4

	Самостоятельная работа: Работа с различными форматами файлов, создание программ для анализа и преобразования данных в текстовых и бинарных файлах.		6
Тема 5. Исключения и их обработка	Содержание учебного материала		Уровень освоения 12
	Изучается механизм обработки исключений в Python с использованием конструкции try-except-finally. Рассматриваются типы встроенных исключений и методы их создания и перехвата. Уделяется внимание правильной обработке ошибок для создания более стабильных программ.	2	
	Информационные (лекционные) занятия		2
	Практические занятия: Создание программы с обработкой различных исключений, включая пользовательские ошибки.		4
	Самостоятельная работа: Написание программ, которые включают перехват исключений при работе с файлами и пользовательским вводом.		6
	Содержание учебного материала		Уровень освоения 12
Тема 6. Итераторы и генераторы	Рассматриваются итераторы как механизм последовательного перебора элементов коллекций. Изучаются генераторы для создания итераторов с использованием ключевого слова yield. Анализируются преимущества генераторов для экономии памяти при работе с большими объёмами данных.	2	
	Информационные (лекционные) занятия		2
	Практические занятия, стажировка: Создание итераторов и генераторов для обработки больших наборов данных и оптимизации использования памяти.		4
	Самостоятельная работа: Изучение и реализация пользовательских итераторов для работы с коллекциями данных.		6
			2
Промежуточная аттестация			2
	Всего:		72

Содержание модуля «Программирование на языке Python. Разработка веб-приложений с использованием Flask»

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем часов
-----------------------------	--	-------------

Тема 4. Расширение функциональности: фильтрация и изменение статуса задач	Самостоятельная работа: Практика работы с базами данных для хранения задач		6
	Содержание учебного материала	Уровень освоения	12
	Рассматриваются методы фильтрации данных на основе пользовательских запросов, а также изменение состояния данных, например, изменение статуса задачи. Изучаются принципы работы с динамическими запросами для отображения данных по критериям. Анализируются подходы к организации бизнес-логики приложения.	2	
	Информационные (лекционные) занятия		2
	Практические занятия, стажировка: Реализация системы фильтрации задач по статусу и возможность изменения состояния задач.		4
Тема 5. Редактирование задач	Самостоятельная работа: Работа над оптимизацией бизнес-логики для фильтрации данных и управление статусом задач.		6
	Содержание учебного материала	Уровень освоения	12
	Изучается процесс редактирования существующих записей (задач) через формы и POST-запросы. Рассматриваются методы извлечения данных для редактирования и обновления записей в базе данных. Анализируются подходы к проверке данных перед обновлением.	2	
	Информационные (лекционные) занятия		2
	Практические занятия, стажировка: Разработка функционала редактирования задач с обновлением данных в базе		4
Тема 6. Рефакторинг кода и отображение статистики	Самостоятельная работа: Реализация проверки и валидации данных при редактировании, создание формы для изменения параметров задач.		6
	Содержание учебного материала	Уровень освоения	12
	Рассматриваются методы рефакторинга кода для улучшения его структуры и читаемости. Изучаются инструменты для сбора статистики (например, количество задач, завершенных или активных) и её отображение в интерфейсе пользователя. Анализируются принципы разделения логики и отображения для улучшения архитектуры приложения.	2	

	Информационные (лекционные) занятия		2
	Практические занятия, стажировка: Проведение рефакторинга существующего кода и создание панели отображения статистики задач.		4
	Самостоятельная работа: Реализация дополнительных показателей статистики и внедрение новых элементов интерфейса для улучшения взаимодействия с пользователем.		6
			2
Промежуточная аттестация			
	Всего:		72

Содержание модуля «Программирование на языке Python. ООП базовый курс»

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем часов
1	2	3
Модуль 4		
Тема 1. Введение в объектно-ориентированное программирование	Содержание учебного материала Рассматриваются основные принципы ООП, такие как инкапсуляция, наследование и полиморфизм. Изучаются объекты и классы как ключевые элементы ООП. Анализируется структура классов и объектов, а также их взаимодействие в рамках программы Информационные (лекционные) занятия Практические занятия: Создание базового класса с атрибутами и методами, создание объекта этого класса и взаимодействие с его методами. Самостоятельная работа: Изучение примеров классов и объектов в реальных проектах, создание нескольких простых классов для закрепления концепции. Содержание учебного материала Изучаются атрибуты классов (переменные) и методы (функции внутри класса), их назначение и способы использования. Рассматривается различие между атрибутами экземпляра класса и статическими атрибутами. Анализируется	Уровень освоения 1 10 2 2 6 12 2
Тема 2. Атрибуты и методы классов		

	назначение методов, их вызов и работа с параметрами.		
	Информационные (лекционные) занятия		2
	Практические занятия: Определение класса с атрибутами и методами, которые взаимодействуют с этими атрибутами.		4
	Самостоятельная работа: Создание классов с различными типами атрибутов и методов для решения практических задач, таких как хранение информации о пользователе или товарах..		6
	Содержание учебного материала	Уровень освоения	12
Тема 3. Конструктор класса (метод init)	Рассматривается метод <code>init</code> как конструктор класса, используемый для инициализации объектов. Изучаются способы передачи аргументов в конструктор и их использование для создания уникальных объектов. Анализируются примеры использования конструктора для автоматической установки атрибутов при создании объекта.	2	
	Информационные (лекционные) занятия		2
	Практические занятия: Создание класса с конструктором, который принимает параметры и инициализирует атрибуты объекта.		4
	Самостоятельная работа: Реализация нескольких классов с конструкторами, принимающими разные типы данных, и тестирование их взаимодействия.		6
	Содержание учебного материала	Уровень освоения	12
Тема 4. Инкапсуляция и методы доступа (getter, setter)	Изучаются принципы инкапсуляции и ограничения доступа к атрибутам класса. Рассматриваются методы доступа, такие как <code>getter</code> и <code>setter</code> , для контроля доступа и изменения значений атрибутов. Анализируются примеры использования инкапсуляции для защиты данных внутри класса.	1	
	Информационные (лекционные) занятия		2
	Практические занятия: Создание класса с приватными атрибутами и методами доступа (<code>getter</code> , <code>setter</code>) для работы с ними.		4
	Самостоятельная работа: Разработка классов с использованием инкапсуляции для управления доступом к данным и их защиты от изменения извне.		6
	Содержание учебного материала	Уровень освоения	12

Тема 5. Наследование классов	Содержание учебного материала	Уровень освоения	12
	Рассматривается наследование как один из ключевых принципов ООП, позволяющий создавать новые классы на основе существующих. Изучаются механизмы расширения функциональности класса и переопределения методов. Анализируются примеры наследования и использование родительских методов и атрибутов в дочерних классах.	2	
	Информационные (лекционные) занятия		2
	Практические занятия: Создание базового класса и его дочернего класса с переопределёнными и добавленными методами и атрибутами.	4	
	Самостоятельная работа: Практика создания нескольких классов с наследованием, разработкой иерархии классов для моделирования реальных объектов (например, животные, транспортные средства).	6	
Тема 6. Полиморфизм и переопределение методов	Содержание учебного материала	Уровень освоения	12
	Рассматривается полиморфизм как возможность разных классов использовать методы с одинаковыми именами, но различной реализацией. Изучаются способы переопределения методов в дочерних классах. Анализируются примеры использования полиморфизма для создания гибких программных структур.	3	
	Информационные (лекционные) занятия		2
	Практические занятия: Практическое задание: Реализация полиморфизма через переопределение методов в наследуемых классах.	4	
	Самостоятельная работа: Создание программы с использованием нескольких классов и полиморфизма для демонстрации работы с одинаковыми методами в разных классах.	6	
Промежуточная аттестация			2
	Всего:		72

Содержание модуля «Программирование на языке Python. ООП продвинутый курс»

Наименование разделов и тем		Содержание учебного материала и формы организации деятельности обучающихся		Объем часов
1	Модуль 5	2		3
Тема 1. Абстрактные классы и методы	Содержание учебного материала		Уровень освоения	10
	Изучаются абстрактные классы, которые служат основой для других классов, но не могут быть использованы напрямую. Рассматриваются абстрактные методы, которые нужно реализовать в дочерних классах. Обсуждается модуль abc, который помогает создавать абстрактные классы.			
	Информационные (лекционные) занятия			2
	Практические занятия: Создание абстрактного класса и реализация его методов в дочерних классах.			2
	Самостоятельная работа: Разработка иерархии классов с использованием абстрактных методов.			6
Тема 2. Множественное наследование	Содержание учебного материала		Уровень освоения	12
	Рассматривается множественное наследование, когда класс наследуется от нескольких родительских классов. Обсуждаются возможные конфликты методов и порядок их разрешения с помощью механизма MRO (Method Resolution Order). Изучаются примеры использования множественного наследования.			
	Информационные (лекционные) занятия			2
	Практические занятия: Создание классов с множественным наследованием и анализ их работы.			4
	Самостоятельная работа: Реализация более сложных структур классов с множественным наследованием.			6
Тема 3. Декораторы классов и методов	Содержание учебного материала		Уровень освоения	12
	Изучаются декораторы для классов и методов, которые позволяют изменять их поведение. Рассматриваются примеры добавления дополнительной функциональности (например, логирования или проверки прав доступа) с помощью декораторов.			

Тема 4. Статические и классовые методы	Информационные (лекционные) занятия		2
	Практические занятия: Написание декоратора для изменения поведения метода класса.		4
	Самостоятельная работа: Разработка декораторов для улучшения функциональности классов.		6
	Содержание учебного материала		12
	Изучаются статические методы, которые не требуют экземпляра класса для вызова, и классовые методы, которые работают с самим классом. Обсуждаются различия между этими типами методов и их практическое применение.	Уровень освоения	
Тема 5. Специальные методы (магические методы), часть 1	Информационные (лекционные) занятия		2
	Практические занятия: Создание класса с примером статических и классовых методов.		4
	Самостоятельная работа: Разработка программы с использованием классовых и статических методов для решения реальной задачи.		6
	Содержание учебного материала		12
	Рассматриваются специальные методы, такие как <code>__str__</code> , <code>__repr__</code> , <code>__len__</code> , которые позволяют управлять поведением объектов класса. Изучаются способы их переопределения для создания более гибких классов.	Уровень освоения	
Тема 6. Специальные методы (магические методы), часть 2	Информационные (лекционные) занятия		2
	Практические занятия: Реализация класса с переопределением магических методов для поведения объектов.		4
	Самостоятельная работа: Изучение других магических методов и их применение в создании своих классов.		6
	Содержание учебного материала		12
	Рассматриваются дополнительные магические методы, такие как <code>__add__</code> , <code>__sub__</code> , <code>__eq__</code> , которые позволяют управлять арифметическими операциями и операторами сравнения между объектами. Анализируются примеры создания классов с перегруженными операторами для реализации кастомной логики работы с объектами.	Уровень освоения	
	Информационные (лекционные) занятия		2
	Практические занятия: Реализация класса с перегрузкой арифметических и логических операторов с использованием магических методов.		4
	Самостоятельная работа: Создание классов с перегрузкой различных операторов для кастомных объектов, изучение возможностей их использования в реальных задачах.		6

Промежуточная аттестация		2
	Всего:	72

Содержание модуля «Модули и библиотеки»

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем часов
1	2	3
Модуль 6		
Тема 1. Введение в библиотеку math	Содержание учебного материала	10
	Рассматривается библиотека math, которая предоставляет функции для выполнения математических операций. Изучаются основные функции, такие как вычисление квадратных корней, логарифмов, тригонометрические функции, а также работа с числами с плавающей точкой.	1
	Информационные (лекционные) занятия	2
	Практические занятия: Решение задач с использованием функций библиотеки math (например, вычисление корней, логарифмов, углов).	2
	Самостоятельная работа: Изучение документации библиотеки math и создание программы для выполнения математических расчётов.	6
Тема 2. Работа с библиотекой numpy	Содержание учебного материала	10
	Рассматривается библиотека numpy, используемая для работы с массивами и матрицами, а также выполнения быстрых математических операций. Изучаются структуры массивов, их создание, индексация и методы для выполнения математических операций на уровне элементов массива.	1
	Информационные (лекционные) занятия	2
	Практические занятия: Создание и обработка многомерных массивов с помощью библиотеки numpy, выполнение операций с элементами.	2
	Самостоятельная работа: Изучение различных методов работы с массивами и их оптимизация для выполнения сложных математических задач.	6
Тема 3. Введение в библиотеку	Содержание учебного материала	12
	Рассматривается библиотека pandas, которая используется для анализа и	1

pandas	обработки данных. Изучаются основные структуры данных: Series и DataFrame, их создание, модификация и фильтрация. Анализируются методы загрузки и сохранения данных из различных источников, таких как CSV и Excel.		
	Информационные (лекционные) занятия		2
	Практические занятия: Работа с DataFrame, выполнение фильтрации, сортировки и группировки данных.		4
	Самостоятельная работа: Загрузка данных из файла, их обработка и визуализация с помощью библиотеки pandas.		6
	Содержание учебного материала	Уровень освоения	12
Тема 4. Работа с HTTP-запросами с использованием библиотеки requests	Изучается библиотека requests, которая упрощает работу с HTTP-запросами. Рассматриваются методы отправки GET и POST запросов, а также работа с заголовками, параметрами и телом запроса. Анализируются примеры взаимодействия с внешними API через запросы и получение данных.	3	
	Информационные (лекционные) занятия		2
	Практические занятия: Отправка запросов к внешнему API и обработка ответов с использованием библиотеки requests.		4
	Самостоятельная работа: Изучение возможностей библиотеки для работы с более сложными запросами.		6
	Содержание учебного материала	Уровень освоения	12
Тема 5. Обработка и анализ данных с использованием pandas и numpy	Изучается совместное использование библиотек pandas и numpy для выполнения математических операций и анализа данных. Рассматриваются методы объединения данных, выполнения математических расчётов и статистического анализа с использованием этих библиотек.	3	
	Информационные (лекционные) занятия		2
	Практические занятия: Выполнение анализа данных с использованием методов numpy и pandas, фильтрация и расчёт статистики.		4
	Самостоятельная работа: Практика создания скриптов для обработки больших наборов данных и выполнение аналитических операций.		6
	Содержание учебного материала	Уровень освоения	12
Тема 6. Визуализация данных (с использованием matplotlib)	Рассматриваются инструменты для визуализации данных с помощью библиотеки matplotlib и методов библиотеки pandas. Изучаются методы	3	

matplotlib и pandas)	создания графиков, гистограмм, диаграмм, а также их настройка и вывод. Анализируются примеры визуализации больших данных для их лучшего понимания и анализа.		
	Информационные (лекционные) занятия		2
	Практические занятия, стажировка: Визуализация данных из DataFrame с помощью matplotlib, создание различных видов графиков и диаграмм.		4
	Самостоятельная работа: Разработка программы для анализа и визуализации данных с использованием библиотеки pandas и matplotlib.		6
			2
Промежуточная аттестация			
Итоговая аттестация			2
	Всего:		72

Для характеристики уровня освоения учебного материала используются следующие обозначения:

1. – ознакомительный (узнавание ранее изученных объектов, свойств);
2. – репродуктивный (выполнение деятельности по образцу, инструкции или под руководством)
3. – продуктивный (планирование и самостоятельное выполнение деятельности, решение проблемных задач)

ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

Требования к квалификации педагогических кадров, представителей предприятий и организаций, обеспечивающих реализацию образовательного процесса.

Промежуточная аттестация по конкретным модулям осуществляется на занятиях в виде собеседования. Освоение программы повышения квалификации завершается итоговой аттестацией в форме комиссионного просмотра. К итоговой аттестации допускается обучающийся, в полном объеме выполнивший учебный план по программе.

Состав комиссии состоит из не менее трех членов комиссии, привлеченных из числа преподавателей, деятельность которых связана с направленностью (профилем) реализуемой программы.

Оценка по результатам комиссионного просмотра формируется коллегиально аттестационной комиссией. При правильном выполнении 50% практических работ и более выставляется оценка «зачтено». Менее 50% - оценка «не зачтено». Лицам, успешно освоившим программу и прошедшим итоговую аттестацию, выдается удостоверение о повышении квалификации.

Реализация программы обеспечивается сотрудниками университета, а также лицами, привлекаемыми к реализации программы на условиях договора гражданско-правового характера. Доля научно-педагогических работников, имеющих образование, соответствующее профилю программы повышения квалификации, в общем числе научно-педагогических работников, реализующих программу, должна составлять не менее 50 процентов. Доля научно-педагогических из числа руководителей и работников организаций, деятельность которых связана с направленностью (реализуемой программы) (имеющих стаж работы в данной профессиональной области не менее 3 лет) в общем числе работников, реализующих программу повышения квалификации не менее 10 процентов.

Требования к материально-техническим условиям

Для реализации программы необходима одна учебная аудитория для проведения лекционных занятий с применением дистанционных образовательных технологий и компьютерный класс с установленным на компьютеры программным обеспечением графических редакторов.

Оборудование учебного кабинета и рабочих мест кабинета: аудитория должна быть оборудована учебной мебелью, доской.

Технические средства обучения: компьютер или ноутбук с выходом в Интернет, проектор, экран.

Программное обеспечение: PyCharmCommunityEdition

Требования к информационным и учебно-методическим условиям

Для реализации программы необходима обеспеченность основной литературой из расчета 50 экземпляров на каждые 100 слушателей. Для реализации программы необходим доступ обучающихся к ЭБС университета, а также обеспечение их справочными и другими учебно-методическими изданиями в печатном или электронном виде.

Перечень используемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основная литература:

1. Лубашева, Т. В., Железко, Б. А. Основы алгоритмизации и программирования : учебное пособие. - 2022-08-04; Основы алгоритмизации и программирования. - Минск: Республиканский институт профессионального образования (РИПО), 2016. - 379 с. - Текст : электронный // IPR BOOKS [сайт]. - URL: <http://www.iprbookshop.ru/67689.html2>.
2. Агафонов Е. Д., Ващенко Г. В. Прикладное программирование : учебное пособие. - Красноярск: Сибирский федеральный университет, 2015. - 112 с. - Текст : электронный // ЭБС «Университетская библиотека онлайн» [сайт]. - URL: <http://biblioclub.ru/index.php?page=book&id=4356403>.

3. Белоцерковская И. Е., Галина Н. В., Катаева Л. Ю. Алгоритмизация. Введение в язык программирования С++. - 2-е изд., испр.. - Москва: Национальный Открытый Университет «ИНТУИТ», 2016. - 197 с. - Текст : электронный // ЭБС «Университетская библиотека онлайн» [сайт]. - URL: <http://biblioclub.ru/index.php?page=book&id=4289354>.
4. Колокольников А. И., Таганов Л. С. Информатика: 630 тестов и теория : пособие. - Москва: Директ-Медиа, 2014. - 429 с. - Текст : электронный // ЭБС «Университетская библиотека онлайн» [сайт]. - URL: <http://biblioclub.ru/index.php?page=book&id=236489>
5. Инженерная и компьютерная графика : учебник и практикум для вузов / Р. Р. Анамова [и др.] ; под общей редакцией С. А. Леоновой, Н. В. Пшеничной. — Москва : Издательство Юрайт, 2021. — 246 с. — (Высшее образование). — ISBN 978-5-9916-8262-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/470037> (дата обращения: 28.08.2024).

Дополнительная литература:

1. Седжвик Р. Алгоритмы на С++. - 2-е изд., испр.. - Москва: Национальный Открытый Университет «ИНТУИТ», 2016. - 1773 с. - Текст : электронный // ЭБС «Университетская библиотека онлайн» [сайт]. - URL: <http://biblioclub.ru/index.php?page=book&id=4291642>.
2. Синецын, С. В., Хлытчиев, О. И. Основы разработки программного обеспечения на примере языка С. - 2021-01-23; Основы разработки программного обеспечения на примере языка С. - Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. - 211 с. - Текст : электронный // IPR BOOKS [сайт]. - URL: <http://www.iprbookshop.ru/73700.html>

Иные источники:

1. Научная электронная библиотека eLIBRARY.RU - <https://elibrary.ru/defaultx.asp>
2. Электронная библиотека Российской государственной библиотеки (РГБ) - <https://www.rsl.ru/>
3. КиберЛенинка: <https://cyberleninka.ru/> - Российская электронная библиотека с доступом к научным статьям, включая материалы по компьютерной графике.
4. Библиотека МГУ (Московского Государственного Университета): <https://www.lib.msu.ru/> - Библиотека МГУ имеет отдел научной литературы по компьютерным наукам, включая книги и журналы по компьютерной графике.
5. Библиотека СПбГУ (Санкт-Петербургского Государственного Университета): <https://www.lib.spbu.ru/> - Библиотека СПбГУ имеет отдел научной литературы по компьютерным наукам, включая книги и журналы по компьютерной графике.
6. Библиотека ИТМО (Санкт-Петербургского Национального Исследовательского Университета Информационных Технологий, Механики и Оптики): <https://www.itmo.ru/ru/library/> - Библиотека ИТМО имеет отдел научной литературы по компьютерным наукам, включая книги и журналы по компьютерной графике.
7. Библиотека МГТУ им. Н.Э. Баумана: <https://www.bmstu.ru/> - Библиотека МГТУ имеет отдел научной литературы по компьютерным наукам, включая книги и журналы по компьютерной графике.

Общие требования к организации образовательного процесса

Образовательный процесс осуществляется в соответствии с настоящей программой в соответствии локальными нормативными актами образовательной организации.

Продолжительность занятий устанавливается локальным нормативным актом образовательной организации. Занятия начинаются не ранее 9.00 часов утра и заканчиваются не позднее 21.00 часов. Занятия могут осуществляться в субботу.