

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего
образования
«Национальный исследовательский Томский государственный университет»



УТВЕРЖДАЮ

Директор по образовательной
деятельности

Е.В. Луков

(подпись)

«23» июня 2023 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«Data-science: программирование нейросетей (Machine Learning)»

технической направленности

Уровень сложности: «Базовый»

144 часа

СОГЛАСОВАНО:

Директор ЦРСК

Заместитель директора ИДО

М.В. Назарова

В.С. Дубровская

г. Томск - 2023



Пояснительная записка

Актуальность программы	Актуальность программы обусловлена потребностью общества в технически грамотных специалистах и полностью отвечает социальному заказу по подготовке квалифицированных кадров в области программирования, а также высоким интересом подростков к IT-сфере. Важнейшей характеристикой подрастающего поколения является активность в информационном пространстве, интернет-коммуникации.
Цель программы	Освоения языков программирования. Изучение методов синтеза нейронных сетей и их практического применения. Формирование умений решения технических (математических) задач на языке программирования. Программист Python способен самостоятельно установить окружение для работы со средой разработки на языке Python и создать скрипт, который берёт на вход данные из файла csv, и выдаёт обработанные данные в виде графика в файл изображение.
Планируемые результаты обучения (знания, умения, навыки)	Сформировать у обучающихся познавательный интерес к процессу сайтостроения; Сформировать знания и умения языка программирования Python, Сформировать умение применять программные средства для решения задач из различных предметных областей способствует развитию логического и комбинаторного мышления
Категория обучающихся по программе	Учащиеся 8 класса, Учащиеся 9 класса, Учащиеся 10 класса, Учащиеся 11 класса, Обучающиеся по программам среднего профессионального образования
Срок освоения программы	2 года
Формы и режим занятий	Очная форма с применением дистанционных образовательных технологий, в том числе, с применением средств электронного обучения
Форма обучения	Очная форма с применением дистанционных образовательных технологий, в том числе, с применением средств электронного обучения
Трудоемкость программы	144 ак.ч.
Примечание	Базовый уровень

Об организации

Наименование поля	Допустимые значения поля	Значение поля
ИНН организации, осуществляющей образовательную деятельность	10 арабских цифр	7018012970
Наименование организации	строка	Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский Томский государственный университет»
Логотип организации	изображение в формате jpeg разрешением не менее 100x100 пиксель	 <p>Национальный исследовательский Томский государственный университет</p>
Ссылка на логотип организации	URL на изображение, находящееся в сети интернет	https://drive.google.com/file/d/170J9BfWy1szAoJPBAKlaWFlgwrlJiqYT/view
Контакты ответственного за программу. ФИО	строка от 5 до 255 символов	Шарыпина Полина Андреевна
Контакты ответственного за программу. Должность	строка от 5 до 255 символов	Заместитель директора Центра совместных образовательных программ
Контакты ответственного за программу. Телефон	Формат +7(XXX)XXXXXX X	+7(952)8957687
Контакты ответственного за программу. E-mail	строка	sharypinapolina@gmail.com

Информация о программе

Наименование поля	Допустимые значения поля	Значение поля (примеры)
Название программы (курса)	строка	Data-science: программирование нейросетей (Machine Learning)
Описание программы	строка не менее 1000 не более 5000 символов	Программа формирует компетенций учащихся в области решения профессиональных задач по программированию с использованием языка Python, применения шаблонов проектирования на Python, работы с Python библиотеками, применения объектно-ориентированного и функционального программирования.
Аннотация	строка до 1000 символов	<p>Дополнительная общеобразовательная программа «Data-science: программирование нейросетей (Machine Learning)» дает возможность каждому ребенку получать дополнительное образование исходя из его интересов, склонностей и способностей, образовательных потребностей, осуществляемых за пределами федеральных государственных образовательных стандартов и федеральных государственных требований.</p> <p>По своему функциональному назначению программа является общеразвивающей и направлена на удовлетворение потребностей обучающихся в интеллектуальном, нравственном совершенствовании, в организации их свободного времени.</p> <p>Направленность общеобразовательной программы «Data-science: программирование нейросетей (Machine Learning)» техническая, так как она ориентирована на изучение основных графических компьютерных программ векторной и растровой графики, Machine Learning), Python в рамках их широкого использования, а также специальных профессиональных возможностей.</p>

		<p>Важнейшей характеристикой подрастающего поколения является активность в информационном пространстве, интернет-коммуникации.</p> <p>Обучающиеся приобретают необходимые навыки по программированию цифровых продуктов, что им помогает определиться с профессиональной сферой деятельности на будущее.</p> <p>Программа «Data-science: программирование нейросетей (Machine Learning)» модифицированная. Она составлена на основе типовых программ. Знания, умения и практические навыки решения актуальных задач, полученные на занятиях, готовят обучающихся к самостоятельной проектно-исследовательской деятельности с применением современных технологий.</p>
Цель программы	строка не менее 100 символов	<p>Освоения языков программирования. Изучение методов синтеза нейронных сетей и их практического применения.</p> <p>Формирование умений решения технических (математических) задач на языке программирования .</p> <p>Программист Python способен самостоятельно установить окружение для работы со средой разработки на языке Python и создать скрипт, который берёт на вход данные из файла csv, и выдаёт обработанные данные в виде графика в файл изображение.</p>
Актуальность	строка не менее 500 символов	<p>Актуальность программы обусловлена потребностью общества в технически грамотных специалистах и полностью отвечает социальному заказу по подготовке квалифицированных кадров в области программирования, а также высоким интересом подростков к IT-сфере.</p> <p>Важнейшей характеристикой подрастающего поколения является активность в информационном пространстве, интернет-коммуникации.</p>

Дополнительная информация	строка	Обучающиеся приобретают необходимые навыки по программированию цифровых продуктов, что им помогает определиться с профессиональной сферой деятельности на будущее.
Формат обучения	значение из: "Онлайн"	Онлайн
	"Оффлайн" "Смешанный"	
Уровень сложности	значение из: "Начальный" " "Базовый" "Продвинутый"	Базовый
Срок освоения образовательной программы	строка	2 года
Объем каждого модуля в ак.ч.	целое число	36
Объем часов в неделю в ак.ч.	целое число	4
Минимальное количество человек на одном потоке курса	целое число <100000	24
Количество уроков	целое число	116
Направленность программы	строка	(техническая) Современные языки программирования Программирование нейросетей (Machine Learning) - Python
Язык программирования	строка	Python
Образовательная программа не представлена для участия в иных федеральных проектах, направленных на дополнительное образование граждан, кроме федерального	строка, значения: "Представлена"/ "Не представлена"	Не представлена

проекта «Развитие кадрового потенциала ИТ-отрасли»		
Версия программы	строка, значение: "Первая версия образовательной программы, ранее не реализованная" "Ранее реализованная версия образовательной программы"	Ранее не реализованная версия образовательной программы
Категория обучающихся (возраст) по программе	строка не менее 10 символов	Учащиеся 8 класса, Учащиеся 9 класса, Учащиеся 10 класса, Учащиеся 11 класса, Обучающиеся по программам среднего профессионального образования
Описание планируемых результатов обучения	строка	Сформировать у обучающихся познавательный интерес к процессу сайтостроения; Сформировать знания и умения языка программирования Python, Сформировать умение применять программные средства для решения задач из различных предметных областей способствует развитию логического и комбинаторного мышления
Ссылка на лендинг	строка	https://it-school.tqu-dpo.ru/machinelearning
Ссылка на LMS	строка	https://odin.study/ru/
Страница обучения на курсе	строка	https://www.odin.study/ru/EducationalProgram/Info/7533
Итоговая аттестация. Количество академических часов	Целое число больше 0	2

Итоговая аттестация. Формы контроля	строка от 4 до 255 символов	Зачет (Итоговое практическое задание)
Итоговая аттестация. Диагностические инструменты	строка не менее 10 символов	<p>При оценке надо руководствоваться следующими критериями:</p> <ol style="list-style-type: none"> 1) полнота и правильность при создании проекта; 2) степень осознанности, понимания изученного; 3) оформление ответа. <p>Оценка «5» ставится, если:</p> <ol style="list-style-type: none"> 1) Написанная программа для демонстрации того, как работает управляемое обучение, выполнено без ошибок, полно изложен изученный материал; 2) учащийся понимает материал, может обосновать свои суждения и действия, хорошо применяет знания на практике, приводит необходимые примеры не только по учебному материалу, но и самостоятельно составленные; 3) в чек-листе излагает материал последовательно и правильно (в соответствии с шаблоном) с точки зрения преподавателя. <p>Оценка «4» ставится, если ученик выполнил проект, удовлетворяющий тем же требованиям, что и для оценки «5», но допускает 1—2 ошибки, которые сам же исправляет, и 1—2 недочета в последовательности оформления излагаемого материала в чек-листе.</p> <p>Оценка «3» ставится, если ученик освоил знания и понимает основы Создания игры "Пинг-понг" с помощью Python с GUI и Pygame, но:</p> <ol style="list-style-type: none"> 1) подготовил материал неполно и допускает неточности в последовательности выполнения заданий;

		<p>2) не может обосновать свои суждения и привести свои примеры;</p> <p>3) допускает ошибки в оформлении чек-листа.</p> <p>Оценка «2» ставится, если ученик обнаруживает незнание большей части изучаемого материала, допускает ошибки в написании кода, беспорядочно и неуверенно излагает материал.</p> <p>Оценка «2» отмечает такие недостатки в подготовке ученика, которые являются серьезным препятствием к успешному овладению последующим материалом.</p>
Итоговая аттестация. Показатели и критерии оценивания	строка не менее 50 символов	<p>Для допуска к итоговой аттестации слушателю необходимо набрать не менее 3 баллов за выполнение каждого практического задания из разделов курса.</p> <p>- итоговая работа, включающая разработку проекта в соответствии с установленными требованиями. Итоговая аттестация проходит в конце обучения в асинхронном формате.</p> <p>ПРОЕКТ- Написать программу для демонстрации того, как работает управляемое обучение.</p>
Итоговая аттестация. Шкала оценивания, нижнее значение	целое число	2
Итоговая аттестация. Шкала оценивания, верхнее значение	целое число, большее чем нижнее значение шкалы	5
Итоговая аттестация. Шкала оценивания, минимальный проходной балл	целое число в диапазоне шкалы	3

Промежуточная аттестация

Количество академических часов	строка не менее 10 символов	4 часа, текущая работа, включающая выполнение 4 практических заданий
Формы контроля	строка не менее 10 символов	Для допуска к итоговой аттестации слушателю необходимо набрать не менее 3 баллов за выполнение каждого практического задания из разделов курса.
Диагностические инструменты	строка не менее 10 символов	Оценка полноты выполнения работы. Оценка способности оперировать полученными знаниями и умениями при решении практической задачи. Проверка результата запуска программного решения на языке Python.
Показатели и критерии оценивания	строка не менее 10 символов	<p>В курсе будут оцениваться:</p> <ul style="list-style-type: none"> - текущая работа, включающая выполнение 4 практических задания. <p>При оценке надо руководствоваться следующими критериями:</p> <ol style="list-style-type: none"> 1) полнота и правильность выполнения заданий; 2) степень осознанности, понимания изученного; 3) оформление ответа. <p>Оценка «5» ставится, если:</p> <ol style="list-style-type: none"> 1) практическое задание выполнено без ошибок, полно изложен изученный материал; 2) учащийся понимает материал, может обосновать свои суждения и действия, хорошо применяет знания на практике, приводит необходимые примеры не только по учебному материалу, но и самостоятельно составленные. <p>Оценка «4» ставится, если ученик выполнил практическое задание, удовлетворяющее тем же требованиям, что и для оценки «5», но допускает 1—2 ошибки.</p> <p>Оценка «3» ставится, если ученик освоил знания и понимает основы практического задания, но:</p> <ol style="list-style-type: none"> 1) подготовил материал неполно и допускает неточности в последовательности выполнения заданий; 2) не может обосновать свои суждения и привести свои примеры. <p>Оценка «2» ставится, если ученик обнаруживает незнание большей части изучаемого материала, допускает ошибки в написании кода, беспорядочно и неуверенно излагает материал. Оценка «2» отмечает такие недостатки в подготовке ученика, которые являются серьезным препятствием к успешному овладению последующим материалом.</p>
Примеры заданий	строка не менее 10 символов	<p>Модуль 1. Практическое задание № 1. Функции</p> <p>Шаг 1. Создайте новый проект.</p> <p>Шаг 2. Создайте файл nums.txt, содержащий несколько чисел, записанных через пробел. Напишите программу, которая</p>

подсчитывает и выводит на экран общую сумму чисел, хранящихся в этом файле.

Шаг 3. Выложите файл в Odin.

Модуль 2. Практическое задание № 2. Решение задач машинного обучения

Шаг 1. Создайте проект

Шаг 2. Задача: Датасет содержит данные о кредитах на покупку электроники, которые были одобрены. Необходимо предсказать, выберет ли покупатель кредит от "ВАШ БАНК"

Шаг 3. Выложите файл в Odin

Модуль 3. Практическое задание № 3. Создание программы для анализа текста

Шаг 1. Создать программу на языке Python, которая будет выполнять следующие задачи:

Описание задания:

1. Ввод текста: Пользователь должен иметь возможность ввести произвольный текст, который будет анализироваться программой.

2. Разбиение текста на слова: Программа должна разбить введенный текст на отдельные слова. При этом необходимо учесть разделительные символы (пробелы, знаки препинания и т.д.) и удалить их из списка слов.

3. Подсчет слов: Программа должна подсчитать общее количество слов в тексте.

4. Подсчет частоты слов: Программа должна вывести наиболее часто встречающиеся слова в тексте и указать их частоту в процентах от общего количества слов.

Пример вывода программы:

Введите текст: Программирование - это увлекательно!

Общее количество слов: 4

Частота слов:

- "программирование": 25%

- "это": 25%

- "увлекательно": 25%

- "в": 25%

Рекомендации:

- Для разделения текста на слова вы можете использовать метод `split()` строки.

- Для подсчета слов вы можете использовать функцию `len()` для списка слов.

		<p>- Для подсчета частоты слов можно создать словарь, где ключами будут слова, а значениями - количество их появлений. После этого можно вычислить частоту слов, разделив количество появлений на общее количество слов и умножив на 100.</p> <p>Шаг 2. Выложите файл в Odin</p> <p>Модуль 4. Практическое задание № 4. Автоматический код загрузки для моделей MobileNet</p> <p>Шаг 1. Создайте текстовый документ</p> <p>Шаг 2. Найдите автоматический код загрузки для моделей MobileNet и вставьте в файл.</p> <p>Шаг 3. Какие бывают параметры URL версии моделей? Укажите также их в документе.</p> <p>Шаг 4. Выложите ссылку в Odin</p>
Шкала оценивания, нижнее значение	строка не менее 10 символов	2
Шкала оценивания, верхнее значение	строка не менее 10 символов	5
Шкала оценивания, минимальный проходной балл	строка не менее 10 символов	3

Преподаватели

Ф.И.О.	Наименование основного места работы	Должность	Высшее образование или среднее профессиональное образование по направлению «Образование и педагогические науки»	Высшее образование или среднее профессиональное образование по иному направлению соответствующим направленности ДОП	Ссылка на вебстраницы с портфолио	Информация о курсах повышения квалификации по профилю преподаваемой дисциплины (за последние 3 года)	Пройдена промежуточная аттестация не менее чем за два года обучения по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направленности ДОП	Отметка о получении согласия на обработку персональных данных
Сергеев Олег Владимирович	ГБОУ Школа имени В.В.Маяковского	Учитель информатики и робототехники	да	да	https://drive.google.com/drive/u/0/folders/1ic6GcR6XHp7fHm1qDGq-m-lkqu5tRXdE			да

Мезенцев Роман Александрович	КГАОУ ДО РМЦ ЦЦО IT-клуб	Педагог дополнительного образования; преподаватель лицейской академии и Яндекс (python, разработка приложений в руQT, pygame, создание ботов для telegram и discord).	да	да	Прошел обучение в АНО ДПО "Школа анализа данных" по программе "Введение в программирование на языке Python" Прошел обучение в АНО ДПО "Школа анализа данных" по программе "Решение прикладных задач на языке Python" Программа повышения квалификации "Основы технологии формирования гибких компетенций при обучении проектной деятельности"		да
------------------------------	--------------------------------------	---	----	----	---	--	----

Борисова Анна Дмитрие вна	Fogsde my.ru	Препода ватель Python- дисципли н	да	Практика на тему: "Классификатор сортов ириса на ЯП Python с применением алгоритма дерева принятия решений и его визуализация" https://colab.research.google.com/drive/1Ty2gy18TGM5aH9nj_w_577ar98cPB8Uv?usp=sharing	Практика на тему: "введение в машинное обучение и базовые настройки модели на языке программирования Python" https://colab.research.google.com/drive/1qgEP4wqhwnNIRc1qkXDTF32GFkr-YWZqI?usp=sharing	да
------------------------------------	-----------------	---	----	---	--	----

			Лекция и практика на тему "Конструирование и нормализация признаков"						
			https://colab.research.google.com/drive/1ad4K-NRwOc2_U0QAvzwpqFQRiHI0yzBO?usp=sharing						

I ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная программа «**Data-science: Программирование нейросетей (Machine Learning)**» реализуется Федеральным государственным автономным образовательным учреждением высшего образования «Национальный исследовательский Томский государственный университет» и дает возможность каждому ребенку получать дополнительное образование исходя из его интересов, склонностей и способностей, образовательных потребностей, осуществляемых за пределами федеральных государственных образовательных стандартов и федеральных государственных требований.

По своему функциональному назначению программа является общеразвивающей и направлена на удовлетворение потребностей обучающихся в интеллектуальном, нравственном совершенствовании, в организации их свободного времени.

Направленность общеобразовательной программы «**Data-science: программирование нейросетей (Machine Learning)**» техническая, так как она ориентирована на изучение основных графических компьютерных программ векторной и растровой графики, Machine Learning) - Python в рамках их широкого использования, а также специальных профессиональных возможностей.

Актуальность и педагогическая целесообразность программы

Актуальность программы обусловлена потребностью общества в технически грамотных специалистах и полностью отвечает социальному заказу по подготовке квалифицированных кадров в области программирования, а также высоким интересом подростков к IT-сфере. Важнейшей характеристикой подрастающего поколения является активность в информационном пространстве, интернет-коммуникации.

Обучающиеся приобретают необходимые навыки по программированию цифровых продуктов, что им помогает определиться с профессиональной сферой деятельности на будущее.

Программа «**Data-science: Программирование нейросетей (Machine Learning)**» модифицированная. Она составлена на основе типовых программ.

Знания, умения и практические навыки решения актуальных задач, полученные на занятиях, готовят обучающихся к самостоятельной проектно-исследовательской деятельности с применением современных технологий.

Цель и основные задачи программы

Целью освоения языков программирования является изучение методов синтеза нейронных сетей и их практического применения.

Требования к уровню знаний, полученных в результате обучения:

Учащиеся должны знать:

- основные типы алгоритмов, иметь представление о структуре программы, основы программирования на языках высокого уровня;
- базовые алгоритмические конструкции;

- содержание этапов разработки программы: алгоритмизация-кодирование-отладка тестирование;
- дополнительные возможности языка Python; для выражения различных алгоритмических ситуаций; алгоритмы и программы на языке Паскаль Python решения нестандартных задач, задач повышенной сложности в математической области; исходные данные и результаты, как строить алгоритмы методом последовательного уточнения (сверху вниз), изображать эти алгоритмы в виде блок-схем; дополнительные средства языка Python; основы постановки задач в области информационных систем.

Учащиеся должны уметь:

- записывать основные языки программирования для решения задач из области математики;
- использовать основные алгоритмические приемы при решении математических задач;
- решать нестандартные задачи и задачи повышенной сложности; анализировать текст чужих программ, находить в них неточности, оптимизировать алгоритм, создавать собственные варианты решения.

Основные задачи:

Обучающие:

- 1) познакомить учащихся с программой Python»;
- 2) познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
- 3) научить читать и составлять блок-схемы;
- 4) сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- 5) изучить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками , кортежами, словарями, множествами)
- 6) научить применять функции при написании программ на языке программирования.

Развивающие:

- формировать у учащихся познавательный интерес к процессу программирования;
- развивать воображение, конструкторское мышление учащихся;
- формировать видение возможностей использования приобретенного на занятиях в учебной и повседневной деятельности;
- расширение знаний по предмету и умение применять программные средства для решения задач из различных предметных областей способствует развитию логического и комбинаторного мышления.

Воспитательные:

- воспитывать уважение и соблюдение авторских прав к работам окружающих;
- формировать положительный психологический климат в среде обучающихся.
- привить навыки культуры общения, деловых качеств, таких как ответственность, самостоятельность, активность, аккуратность.

Педагогические принципы, определяющие теоретические подходы к построению образовательного процесса

Реализация программы «**Data-science: Программирование нейросетей (Machine Learning)**» основывается на принципах научности, последовательности, системности, связи теории с практикой, доступности.

В целях раскрытия педагогического и развивающего потенциала учебно-воспитательного процесса по программе акцент в ней делается на следующих принципах:

Принцип прочности предполагает стремление к тому, чтобы приобретенные знания, умения и навыки обучающихся были прочно закреплены, в первую очередь, практически.

Для этого необходимо их участие в соревнованиях, где полученные умения и навыки можно проверить на практике, в выездных мероприятиях, а также постоянное поддержание интереса обучающихся к занятиям.

Принцип проектности предполагает последовательную ориентацию всей деятельности педагога на подготовку и выведение ребенка в самостоятельное проектное действие, развертываемое в логике замысел – реализация – рефлексия. В ходе проектирования перед человеком всегда стоит задача представить себе еще не существующее, но то, что он хочет, чтобы появилось в результате его активности.

Принцип активности предполагает необходимость строить учебный процесс таким образом, чтобы учащиеся не просто воспринимали предлагаемый материал, но и стремились закрепить полученные знания, умения и навыки, анализировали ошибки и достижения свои и товарищей, самостоятельно решали поставленные задачи.

Предполагаемые результаты

По окончании реализации программы учащийся будет:

- 1) Знать основные алгоритмические конструкции программ;
- 2) Уметь писать на языке Python, простейшие программы связанные с числовыми и символьными данными;
- 3) Уметь применять свои знания на практике при решении технических (математических задач)

Основные характеристики образовательного процесса

Программа предполагает участие детей в возрасте от 13 до 18 лет.

Уровень освоения программы базовый, что предполагает освоение обучающимися специализированных знаний, обеспечение трансляции общей и целостной картины тематического содержания программы.

Форма обучения: очная форма с применением дистанционных образовательных технологий.

Срок реализации программы - 2 года.

Режим работы: занятия проводятся 2-3 раза в неделю по 2 академических часа, длительность одного академического часа – 45 минут.

Продолжительность образовательного процесса: начало занятий 01 октября 2023 г., завершение 31 мая 2024 г.

Календарный учебный график формируется непосредственно при реализации общеобразовательной программы «Data-science Программирование нейросетей (Machine Learning)».

Первый модуль с 01.10.2023 - 30.11.2023

Второй модуль: с 01.12.2023 - 31.01.2024

Третий модуль: с 01.02.2024 - 31.03.2024

Четвертый модуль: с 01.04.2024 - 31.05.2024

Занятия проводятся 2-3 раза в неделю по 2 академических часа, длительность одного академического часа – 45 минут.

Объем учебных часов по программе 144 часа.

Ожидаемые результаты освоения программы

После завершения обучения по программе у обучающихся будут сформированы:

- критическое и системное мышление – развитие мышления, обуславливающего совершение обоснованного выбора;
- умения работать с информацией и медиа средствами – умение находить, оценивать и создавать информацию в разных формах и на различных типах медиа оборудования;
- умение ставить и решать проблемы;
- направленность на саморазвитие;
- поиск и размещение соответствующих ресурсов;
- перенос информации из одной области знаний в другую;
- социальную ответственность – умение действовать в интересах большого сообщества;
- этично себя вести в личном, рабочем и общественном контекстах;
- использовать уже изученный материал для работы над проблемными ситуациями по работе и обработке данных;
- самостоятельно действовать по составленному плану, сверяясь с ним и целью деятельности, исправляя ошибки.
- самостоятельно оценивать степень успешности своей образовательной деятельности;
- самостоятельно отбирать, сопоставлять и проверять информацию, полученную из различных источников для решения задач (проблем) и создавать базы данных;
- применять современные информационные технологии, обеспечивающие доступ к необходимым профильным базам, банкам данных, источникам информации по теме исследования.

Данная программа адаптирована для занятий с учащимися среднего и старшего возраста, с различной степенью подготовки, включает теоретические и практические задания.

Входные требования к обучающимся: тестирование - базовое владение персональным компьютером.

Для занятий -наличие ПК, камеры, микрофона, стабильного доступа в интернет.

Вступительное тестирование должно отвечать следующим функциональным характеристикам: иметь не менее 25 вопросов. Вступительное тестирование должно состоять из мотивационных вопросов и заданий в области математики/информатики/программирования для определения уровня обучения. Вступительное тестирование должно оценивать мотивацию, предрасположенность и знания школьника для профессионального развития в сфере информационных технологий и получения соответствующего обучения.

Календарный учебный график формируется непосредственно при реализации программы общеобразовательной программы **«Data-science: Программирование нейросетей (Machine Learning)»**. Календарный учебный график представлен в форме расписания занятий при наборе группы на обучение.

СОДЕРЖАНИЕ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ

СОДЕРЖАНИЕ УЧЕБНОГО (ТЕМАТИЧЕСКОГО) ПЛАНА

«Data-science: программирование нейросетей (Machine Learning)»

№ п/п	Наименование модулей (курсов)	Общая трудоемкост ь, ч.	Всего контактных часов			Контактные часы			СРС, ч.	Формы контроля
			Общее количество Лекций и ПЗ	ДЗ	Лекции	Практические занятия	Зачет			
I	Модуль 1. Python. Введение в программирование	36	29	1	9	21	6	Зачет		
1.1	Тема 1.1. Знакомство с языком программирования Python	1	1		1					
1.2	Тема 1.2. PyChart Community. Основы работы	2	1		1		1			
1.3	Тема 1.3. Типы данных. Переменные	1	1		1					
1.4	Тема 1.4. Ввод и вывод данных	2	1			1	1			
1.5	Тема 1.5. Логические выражения и	1	1		1					

операторы																		
1.6	Тема 1.6. Ветвление. Условный оператор	4	3					1			2					1		
1.7	Тема 1.7. Исключения и их обработка в Python	2	2								2							
1.8	Тема 1.8. Циклы в программировании. Цикл while	4	3					1			2					1		
1.9	Тема 1.9. Функции в программировании	1	1					1										
1.10	Тема 1.10. Локальные и глобальные переменные	2	2								2							
1.11	Тема 1.11. Возврат значений из функций. Оператор return	2	1								1					1		
1.12	Тема 1.12. Параметры и аргументы функции	1	1					1										
1.13	Тема 1.13. Случайные числа в Python – random, randint и randrange	2	2								2							
1.14	Тема 1.14. Списки	2	2								2							
1.15	Тема 1.15. Цикл for	4	3					1			2					1		
1.16	Тема 1.16. Функция enumerate	1	1								1							
1.17	Тема 1.17. Строки	1	1								1							
1.18	Тема 1.18. Словари	1	1								1							
1.19	Тема 1.19. Функции	2	1					1			2							зачет

	Модуль 2. Машинное обучение на Python	36	23	1	9	21	6	Зачет
2								
2.1	Тема 2.1 Начало работы. Введение	3	2		2		1	
2.2	Тема 2.2 Основные инструменты и библиотеки	5	3		1	3	1	
2.3	Тема 2.3 Пример классификации сортов ириса	4	3		1	3		
2.4	Тема 2.4 Методы машинного обучения	4	3		1	3		
2.5	Тема 2.5 Предварительная обработка данных	5	3		1	3	1	
2.6	Тема 2.6 Типы данных и конструирование признаков	5	3		1	3	1	
2.7	Тема 2.7 Работа с текстовыми данными	5	3		1	3	1	
2.8	Тема 2.8 Общий подход к решению задач машинного обучения	5	3	1	1	3	1	зачет
3	Модуль 3. Продвинутое программирование на языке Python	36	28	1	10	20	6	Зачет
	<i>Раздел 1. Расширенное изучение Python</i>	<i>20</i>	<i>16</i>		<i>6</i>	<i>11</i>	<i>3</i>	
3.1	Тема 3.1. Расширенное изучение Python	1	1		1			
3.2	Тема 3.2 Лексическая структура	1	1			1		
3.3	Тема 3.3 Типы данных и значения	1	1		1			

3.4	Тема 3.4 Переменные	4	3		1	2	1	
3.5	Тема 3.5 Выражения и операторы	1	1			1		
3.6	Тема 3.6 Инструкции	2	2		1	1		
3.7	Тема 3.7 Структуры данных: списки, кортежи, множества и словари	1	1			1		
3.8	Тема 3.8 Функции	3	2		1	1	1	
3.9	Тема 3.9 Классы и объекты	3	2		1	1	1	
3.10	Тема 3.10 Модули и пакеты	1	1			1		
3.11	Тема 3.11 Исключения и их обработка	2	1			2		
	Раздел 2. Python в веб-браузерах	16	12	1	4	9	3	
3.12	Тема 3.12 Регулярные выражения	2	2		1	1		
3.13	Тема 3.13 Работа с файлами	1	1			1		
3.14	Тема 3.14 Ввод и вывод данных	2	1			1	1	
3.15	Тема 3.15 Итераторы и генераторы	3	2		1	1	1	
3.16	Тема 3.16 Декораторы	1	1			1		
3.17	Тема 3.17 Многопоточность	1	1			1		
3.18	Тема 3.18 Веб-скрапинг	3	2		1	1	1	
3.19	Тема 3.19 Тестирование и отладка	1	1			1		
3.20	Тема 3.20 Виртуальное окружение и управление зависимостями	2	1	1	1	1		зачет

4	Модуль 4. Python: Продвинутое искусство машинного обучения	36	27	3	9	21	6	Зачет
4.1	Тема 4.1 Глубокое обучение и нейронные сети	2	2		1	1		
4.2	Тема 4.2 Обработка изображений в машинном обучении	3	3		1	2		
4.3	Тема 4.3 Обучение без учителя	4	3		1	2	1	
4.4	Тема 4.4 Обучение с подкреплением	4	3		1	2	1	
4.5	Тема 4.5 Рекомендательные системы	4	3		1	2	1	
4.6	Тема 4.6 Анализ данных и визуализация	4	3		1	2	1	
4.7	Тема 4.7 Высоконагруженные системы и машинное обучение	3	3		1	2		
4.8	Тема 4.8 Обработка полученных знаний	2	1			1	1	
4.9	Тема 4.9 Производительность и оптимизация моделей машинного обучения	4	3		1	2	1	
4.10	Тема 4.10 Практика на Python	4	3	1	1	3		Зачет
5	Итоговая аттестация	2		2		2		Итоговая проектная работа

Итого	144	107	6	37	83	24
--------------	------------	------------	----------	-----------	-----------	-----------

II ОЦЕНКА КАЧЕСТВА ОСВОЕНИЯ ПРОГРАММЫ (формы аттестации, оценочные и методические материалы)

Условия прохождения курса

В курсе будут оцениваться:

- текущая работа, включающая выполнение 4 практических задания.

Каждое задание будет оцениваться по шкале:

Шкала оценивания:

Нижнее значение 2

Верхнее значение 5

Минимальный проходной балл для успешной сдачи 3

При оценке надо руководствоваться следующими критериями:

- 1) полнота и правильность выполнения заданий;
- 2) степень осознанности, понимания изученного;
- 3) оформление ответа.

Оценка «5» ставится, если:

- 1) практическое задание выполнено без ошибок, полно изложен изученный материал;
- 2) учащийся понимает материал, может обосновать свои суждения и действия, хорошо применяет знания на практике, приводит необходимые примеры не только по учебному материалу, но и самостоятельно составленные.

Оценка «4» ставится, если ученик выполнил практическое задание, удовлетворяющее тем же требованиям, что и для оценки «5», но допускает 1—2 ошибки.

Оценка «3» ставится, если ученик освоил знания и понимает основы практического задания, но:

- 1) подготовил материал неполно и допускает неточности в последовательности выполнения заданий;
- 2) не может обосновать свои суждения и привести свои примеры.

Оценка «2» ставится, если ученик обнаруживает незнание большей части изучаемого материала, допускает ошибки в написании кода, беспорядочно и неуверенно излагает материал.

Оценка «2» отмечает такие недостатки в подготовке ученика, которые являются серьезным препятствием к успешному овладению последующим материалом.

Для допуска к итоговой аттестации слушателю необходимо набрать не менее 3 баллов за выполнение практических заданий из разделов курса.

- итоговая работа, включающая разработку проекта в соответствии с установленными требованиями. Итоговая аттестация проходит в конце обучения в асинхронном формате. Максимальное количество баллов, которое слушатели могут получить за итоговую аттестацию – 5

Для успешного завершения обучения по курсу необходимо, чтобы 100 % заданий в рамках текущей работы были зачтены (оценены преподавателем на 3, 4 или 5 баллов), итоговая

аттестационная работа была зачтена (оценены преподавателем на 3, 4 или 5 баллов)

В таблице представлен перечень заданий, которые будут проводиться на практике в режиме онлайн с преподавателем в период обучения на курсе.

Таблица 1

	Мероприятия промежуточного контроля	Количество баллов
1.	Практическое задание № 1. Функции	2-5
2.	Практическое задание № 2. Решение задач машинного обучения	2-5
3.	Практическое задание № 3. Создание программы для анализа текста	2-5
4.	Практическое задание № 4. Автоматический код загрузки для моделей MobileNet	2-5

В таблице представлен перечень заданий, которые необходимо выполнить в период обучения на курсе.

Текущий контроль уровня усвоения материала осуществляется по окончании изучения каждой темы – выполнением практических заданий.

Итоговый контроль проходит в конце курса (всех модулей обучения) – в форме зачета. Формы проведения аттестации: зачетная работа.

Оценочные материалы для текущего контроля успеваемости:

Модуль 1. Практическое задание № 1. Функции

Шаг 1. Создайте новый проект.

Шаг 2. Создайте файл `nums.txt`, содержащий несколько чисел, записанных через пробел. Напишите программу, которая подсчитывает и выводит на экран общую сумму чисел, хранящихся в этом файле.

Шаг 3. Выложите файл в Odin.

Модуль 2. Практическое задание № 2. Решение задач машинного обучения

Шаг 1. Создайте проект

Шаг 2. Задача: Датасет содержит данные о кредитах на покупку электроники, которые были одобрены. Необходимо предсказать, выберет ли покупатель кредит от “ВАШ БАНК”

Шаг 3. Выложите файл в Odin

Модуль 3. Практическое задание № 3. Создание программы для анализа текста

Шаг 1. Создать программу на языке Python, которая будет выполнять следующие задачи:
Описание задания:

1. Ввод текста: Пользователь должен иметь возможность ввести произвольный текст, который будет анализироваться программой.

2. Разбиение текста на слова: Программа должна разбить введенный текст на отдельные слова. При этом необходимо учесть разделительные символы (пробелы, знаки препинания и т.д.) и удалить их из списка слов.

3. Подсчет слов: Программа должна подсчитать общее количество слов в тексте.

4. Подсчет частоты слов: Программа должна вывести наиболее часто встречающиеся слова в тексте и указать их частоту в процентах от общего количества слов.

Пример вывода программы:

Введите текст: Программирование - это увлекательно!

Общее количество слов: 4

Частота слов:

- "программирование": 25%

- "это": 25%

- "увлекательно": 25%

- "в": 25%

Рекомендации:

- Для разделения текста на слова вы можете использовать метод `split()` строки.

- Для подсчета слов вы можете использовать функцию `len()` для списка слов.

- Для подсчета частоты слов можно создать словарь, где ключами будут слова, а значениями - количество их появлений. После этого можно вычислить частоту слов, разделив количество появлений на общее количество слов и умножив на 100.

Шаг 2. Выложите файл в Odin

Модуль 4. Практическое задание № 4. Автоматический код загрузки для моделей MobileNet

Шаг 1. Создайте текстовый документ

Шаг 2. Найдите автоматический код загрузки для моделей MobileNet и вставьте в файл.

Шаг 3. Какие бывают параметры URL версии моделей? Укажите также их в документе.

Шаг 4. Выложите ссылку в Odin

Для допуска к итоговой аттестации слушатели должны выполнить все задания и получить не менее 3 баллов за каждое из них.

Итоговая аттестация

В качестве итоговой работы слушателям предлагается разработать проект по теме "Data-science: Программирование нейросетей (Machine Learning)".

Проект представляет собой "Data-science: Программирование нейросетей (Machine Learning)". Сделать чек лист по проделанной работе в документе word. Расписать в свободной форме.

Максимальное количество баллов, которое слушатели могут получить за итоговую аттестацию – 5 (минимальное количество баллов за итоговую аттестацию – 3).

Задание

Написать программу для демонстрации того, как работает управляемое обучение. Для этого мы будем использовать библиотеку Sklearn и язык Python. Sklearn - это библиотека машинного обучения для языка программирования Python, которая предоставляет множество возможностей, таких как многоступенчатый анализ, регрессия и алгоритмы кластеризации. Кроме того, Sklearn хорошо взаимодействует с библиотеками NumPy и SciPy.

Этапы работы

1. Установка Sklearn

Инструкция по установке Sklearn предлагает очень простой способ установки для

различных платформ. Для работы библиотеки требуется несколько зависимостей:

Python (≥ 2.7 or ≥ 3.3),

NumPy (≥ 1.82)

SciPy ($\geq 0.13.3$)

Если эти зависимости уже установлены, то можно установить Sklearn, просто выполнив команду:

```
pip install -U scikit-learn
```

Более простым способом является установка Anaconda. Данный пакет сам установит все зависимости, так что вам не придется устанавливать их по одной.

Чтобы проверить, что Sklearn работает корректно, просто импортируйте эту библиотеку в интерпретаторе языка Python:

```
import sklearn
```

Если это не вызвало ошибок, значит все готово к работе.

2. Задача

После того, как мы разобрались с установкой, давайте вернемся к нашей задаче. Допустим, мы хотим научиться различать животных. Для этого мы создадим алгоритм, который сможет определить, является ли то или иное животное лошадью или курицей.

Сперва нам необходимо собрать исходные данные для каждого вида животного. Некоторые исходные данные представлены в таблице ниже.

Полученные нами исходные данные содержат некоторые основные характеристики и их значения для наших двух животных. Чем больше исходных данных, тем более точными и менее предвзятыми будут результаты.

Основываясь на имеющихся данных, мы можем написать алгоритм и обучить его определять животное на основе изученных данных и классифицировать его как лошадь или курицу. Теперь необходимо написать алгоритм, который будет выполнять поставленную задачу.

3. Процесс выполнения задания

1. Импортируем модуль tree из Sklearn
2. Определим набор характеристик, по которым будем классифицировать животных
3. Определим результат, который будет давать каждый набор значений. Пусть курица будет представлена результатом 0, а лошадь - результатом 1
4. Определяем классификатор, который будет основываться на схеме принятия решения
5. Передаем наши данные классификатору
6. Определить животное ростом 18 сантиметров, весом 0.6 кг и температурой 41 градус
7. Определить животное ростом 94 см, весом 600 кг и температурой 37.5 градусов

4. Итог работы

Вы должны получить алгоритм, который изучает все характеристики и названия двух животных. Знания об этих данных далее используются при идентификации новых животных.

Условия реализации программы:

Материально – техническое обеспечение:

- Стационарное рабочее место ученика (компьютер);
- Интерпритатор ;

Информационное обеспечение:

- Интернет;
- Питонтьютер.

КАДРОВЫЕ УСЛОВИЯ

Руководитель программы:

Сергеев Олег Владимирович, Российский государственный социальный университет, Москва
Факультет: Информационные технологии (бакалавриат) Специальность: Педагогическое образование (информатика) Сентябрь 2011 - Июль 2015
Российский государственный социальный университет, Москва Факультет: Информационные технологии (магистратура)
Специальность: Педагогическое образование (информатика) Сентябрь 2016 - Июль 2019
Стаж работы – более 7 лет.

Сертификаты/ курсы:

Разработка веб-приложений с использованием Javascript ФГАОУ ВО "Национальный исследовательский университет "Высшая школа экономики". Московский институт электроники и математики им. А. Н. Тихонова 2019 год.

Программирование веб-сайтов и веб-приложений на HTML и CSS ФГАОУ ВО "Национальный исследовательский университет "Высшая школа экономики". Московский институт электроники и математики им. А. Н. Тихонова 2018 год.

Методика и использование 3d-прототипирования и моделирования на учебных занятиях в инженерном классе Академия «Просвящение» 2017 год.

«Подготовка технических специалистов, оказывающих информационно-техническую помощь руководителю и организаторам пункта проведения экзамена при проведении государственной итоговой аттестации по образовательным программам основного общего образования» ГАОУ ДПО МЦКО 2017 год.

"ИК технологии при обучении робототехнике" Городской методический центр Департамента образования города Москвы 2016 год
Дизайнер-верстальщик РГСУ 2013 год

III. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ МОДУЛЯ (организационно-педагогические)

Материально-технические условия реализации программы: Материально-технические условия реализации программы

Обучение по программе реализовано в режиме синхронно-асинхронной работы учащихся в электронной среде. Для проведения синхронных занятий применяется программа видеоконференцсвязи Zoom. Дополнительно для организации работы учащиеся могут использоваться Google-сервисы. Асинхронная работа учащихся реализуется на базе электронного курса на платформе электронного обучения LMS Odin.

Требования к оборудованию

Сеть: скорость соединения от 2 Мб/с.

Оборудование для синхронных занятий: персональный компьютер (рекомендуется) / мобильный телефон / планшет; наушники, микрофон и камера (обязательно).

Для работы на платформе электронного обучения LMS Odin рекомендуется использовать персональный компьютер.

Для успешной реализации Программы необходимо, чтобы рабочее место обучающегося и преподавателя включали в себя:

- компьютеры, обеспечивающие возможность работы с мультимедийным контентом: воспроизведение видеоизображений, качественный стереозвук в наушниках, речевой ввод с микрофона и др.

периферийное оборудование:

- принтер (черно/белой печати, формата А4);
- устройства для ввода визуальной информации (сканер, цифровой фотоаппарат, web-камера и пр.);
- устройства создания графической информации (графический планшет), используемые для создания и редактирования графических объектов, ввода рукописного текста;
- акустические колонки;
- оборудование, обеспечивающее подключение к сети Интернет (комплект оборудования для подключения к сети Интернет, сервер).

программное обеспечение компьютера:

- операционная система семейства MacOS или Windows;
- программа Python;
- файловый менеджер (в составе операционной системы или др.);
- почтовый клиент (в составе операционных систем или др.);
- браузер (в составе операционных систем или др.);
- мультимедиа проигрыватель (в составе операционной системы или др.);
- антивирусная программа;
- программа-архиватор;
- программа-переводчик;
- программа интерактивного общения;
- текстовый редактор;
- растровый графический редактор;
- звуковой редактор;
- редактор Web-страниц.

Учебно-методическое и информационное обеспечение программы: Методические рекомендации и пособия по изучению курса.

Программа реализуется в формате смешанного обучения. Синхронные занятия включают интерактивные лекции и практические занятия, предусматривающие групповую и индивидуальную работу учащихся. Для организации асинхронной работы учащихся используются записи синхронных занятий, презентации, конспекты лекций, практические задания и тестовые вопросы, размещаемые в электронном курсе на платформе электронного обучения LMS Odin.

Содержание комплекта учебно-методических материалов.

Для изучения программы используется электронный курс в LMS Odin. Электронный курс включает информационно-организационные материалы по программе (учебно-тематический план, календарный график работы по программе, информацию о преподавателях программы, чат для объявлений и вопросов, сведения о результатах обучения), набор видеолекций, презентации к лекциям, набор ссылок на внешние ресурсы, систему заданий с инструкциями, списки основной и дополнительной литературы.

Литература

Список литературы, использованной при написании программы

1. Банкрашков А. Программирование для детей на языке Python. – Москва: АСТ, 2017.
2. Бриггс Дж. Python для детей. Самоучитель по программированию. – Москва: Манн, 2018.
3. Бэрри П. Изучаем программирование на Python. – Москва: Эксмо, 2017.
4. ВандерПлас Дж. Python для сложных задач. Наука о данных и машинное обучение. – Санкт-Петербург: Питер, 2018.
5. Васильев А. Python на примерах. Практический курс. Москва: Наука и техника, 2018.
6. Вордэрман К. Программирование на Python. Иллюстрированное руководство для детей. – Москва: Манн, Иванов и Фербер, 2018.
7. Гэддис Т. Начинаем программировать на Python. – Санкт-Петербург: БВХ-Петербург, 2019.
8. Доусон М. Програмируем на Python. – Санкт-Петербург: Прогресс книга, 2019.
9. Жуков Р. Язык программирования Python: практикум. Учебное пособие. – Москва: Инфра-М, 2020.
10. Златопольский Д.М. Основы программирования на языке Python. – Москва: ДМК Пресс, 2018.
11. Лутц М. Изучаем Python. В 2-х т. – Москва: Вильямс, 2019.
12. Любанович Б. Простой Python. Современный стиль программирования. – Санкт-Петербург: Питер, 2019.
13. МакГрат М. Программирование на Python для начинающих. Москва: Эксмо, 2015.
14. Маккини У. Python и анализ данных. – Москва: ДМК Пресс, 2018.
15. Мартелли А., Рейвенсक्रофт А., Холден С. Python. Справочник. Полное описание языка. – Москва: Диалектика, 2018.
16. Мэтиз Э. Изучаем Python. Программирование игр, визуализация данных, веб-приложения. – Санкт-Петербург: Питер, 2020.
17. Мюллер Дж.П. Python для чайников. – Москва: Вильямс, 2019.
18. Персиваль Г. Python. Разработка на основе тестирования. – Москва: ДМК Пресс, 2018.
19. Прадик Дж. Искусственный интеллект с примерами на Python. Создание приложений искусственного интеллекта с помощью Python для взаимодействия с окружающим миром. – Москва: Вильямс, 2019.
20. Пэйн Б. Python для детей и родителей. Играй и программируй. – Москва, Эксмо, 2018.
21. Рамальо Л. Python к вершинам мастерства. – Москва: ДМК Пресс, 2016.
22. Свейгарт Э. Автоматизация рутинных задач с помощью Python. Практическое руководство для начинающих. – Москва: Вильямс, 2018.
23. Свейгарт Э. Учим Python, делая крутые игры. – Москва: Бомбора, 2018.
24. Содем Я.Э. Программирование компьютерного зрения на языке Python. – Москва: ДМК Пресс, 2016.
25. Томашевский П. Привет, Python. Моя первая книга по программированию. – Москва: Наука и техника, 2018.
26. Харрисон М. Как устроен Python. – Санкт-Петербург: Питер, 2019

Компетенции

Наименование компетенции	Тип компетенции	Знания, соответствующие компетенции	Умения, соответствующие компетенции	Владение инструментами, соответствующие компетенции
строка, не менее 10 символов	строка, значение из "общекультурные", "общепрофессиональные", "профессиональные"	строка не менее 50 символов, перечень знаний	строка не менее 50 символов, перечень умений	строка не менее 50 символов, перечень инструментов
ПК-1 Создание программ на языке Python	общепрофессиональные	<ul style="list-style-type: none"> - Принципы работы информационных технологий - Принципы работы алгоритмов - Существующие структуры данных - Принципы написания 	<ul style="list-style-type: none"> - ПК - Устанавливать Python, на Писать циклы, условия, на Python - Подключать сторонние библиотеки - Читать и создавать json, csv, xml файлы 	<ul style="list-style-type: none"> - Средствами разработки: PyCharm, а также онлайн эмулятором Python - Командной строкой (cmd) Методами машинного обучения предварительной обработкой данных подходом к решению задач машинного обучения

		программ на Python - Классы, функции в Python		
--	--	---	--	--

Модули

Наименование поля	Допустимые значения полей	Значение полей	Значение полей	Значение полей
Порядковый номер модуля	целое число	1	2	3
Название модуля	1.	Модуль 1. Python. Введение в программирование	Модуль 2. Машинное обучение на Python	Модуль 3. Продвинутое программирование на языке Python
				Модуль 4. Python: Продвинутое искусство машинного обучения

Описание модуля	<p>2.</p> <p>Модуль посвящен одному из бурно развивающихся и популярных в настоящее время сценарных языков программирования - Python. Язык Python позволяет быстро создавать как прототипы программных систем, так и сами программные системы, помогает в интеграции программного обеспечения для решения производственных задач. Python имеет богатую стандартную библиотеку и большое количество модулей расширения практически для всех нужд отрасли информационных технологий. Благодаря ясному синтаксису изучение языка не</p>	<p>Область компьютерных наук, с помощью которой компьютерные системы могут придавать смысл. Задача в том, чтобы разобратся во всех данных. Предприятия и организации пытаются справиться с этим путем создания интеллектуальных систем с использованием концепций и методологий из Machine learning. Машинное обучение — самая захватывающая область информатики. Лучший способ начать использовать Python для машинного обучения — это разобрать готовый проект по машинному обучению и затем его сделать самому с нуля:</p>	<p>Python является инструментом, с помощью которого изучаются различные области знаний. Python — это удобные структуры данных, методы программирования и отладки компактных эффективных программ. Разработчики называют его лучшим языком для машинного обучения, потому что он сравнительно простой, гибкий и имеет хорошие фреймворки для работы с искусственным интеллектом. Модуль посвящен изучению более глубоким тем языка программирования Python.</p>	<p>Машинное обучение сейчас является одной из самых востребованных областей в программировании. Алгоритмы машинного обучения могут выполнять задачи, которые раньше казались недоступными компьютерам. С помощью Python пишутся математические основы для построения нейронных сетей. Изучая данный модуль, вы поймете, чем отличается обучение с подкреплением от обучения без учителя, научитесь определять, какие</p>
-----------------	--	---	--	--

		<p>составляет большой проблемы. Написанные на нем программы получают структурированными по форме, и в них легко проследить логику работы.</p>	<ul style="list-style-type: none"> ● Это заставит вас установить и запустить Python интерпретатор. ● Это даст вам понимание с высоты птичьего полета о том, как выглядит небольшой проект. ● Это даст вам уверенность, чтобы перейти к собственным задачам и проектам. 		<p>технологии использовать для определенных наборов данных, научитесь использовать эти технологии на практике и писать код на Python для решения задач.</p>
--	--	---	---	--	---

Аттестация по итогам модуля. Количество ак. часов	целое число	1	1	1	1	1
Аттестация по итогам модуля. Формы контроля	строка не менее 4 символов	Практическое задание №1 Функции	Практическое задание №2 Решение задач машинного обучения	Практическое задание №3 Создание программы для анализа текста	Практическое задание №4. Автоматический код загрузки для моделей MobileNet	2-5
Аттестация по итогам модуля. Шкала оценивания, нижнее значение	целое число	2-5	2-5	2-5	2-5	2-5
Аттестация по итогам модуля. Шкала оценивания, верхнее значение	целое число	2-5	2-5	2-5	2-5	2-5
Аттестация по итогам модуля. Шкала оценивания, минимальный проходной балл для успешной сдачи	целое число в диапазоне шкалы	3	3	3	3	3

РАБОЧАЯ ПРОГРАММА

«Data-science: Программирование нейросетей (Machine Learning)»

I. АННОТАЦИЯ

Программа «Data-science: Программирование нейросетей (Machine Learning)» Программа направлена на формирование профессиональных практических навыков в области цифровых технологий актуальных в 2023-2024 году.

Программа направлена на формирование компетенций в соответствии с трудовыми функциями программиста.

В результате обучения выпускник программы будет способен:

- Разбираться, как устроены нейронные сети;
- Разбираться с различными областями применения нейронных сетей.;
- Программировать на Python, работать с библиотеками, с помощью которых сможете создать свою первую модель машинного обучения.;
- осуществлять самостоятельный поиск и изучение технической документации.

Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
Модуль 1. Python. Введение в программирование Модуль посвящен одному из бурно развивающихся и популярных в настоящее время	Тема 1.1. Знакомство с языком программирования Python (1 ч.)	Как писать программы на Python	теоретические занятия	(1 ч.)
	Тема 1.2. PyChart Community.	Создание проекта в PyChart Community, первоначальная настройка среды и некоторые принципы работы.	теоретические занятия	(1 ч.)

<p>сценарных языков программирования - Python. Язык Python позволяет быстро создавать как прототипы программных систем, так и сами программные системы, помогает в интеграции программного обеспечения для решения производственных задач. Python имеет богатую стандартную библиотеку и большое количество модулей расширения</p>	<p>Основы работы (2 ч.)</p>	<p>Изучить дополнительный материал по теме 1.2. PyCharm Community. Основы работы https://www.jetbrains.com/ru-ru/pycharm/learn/</p>	<p>самостоятельная работа</p>	<p>(1 ч.)</p>
<p>создавать как прототипы программных систем, так и сами программные системы, помогает в интеграции программного обеспечения для решения производственных задач. Python имеет богатую стандартную библиотеку и большое количество модулей расширения</p>	<p>Тема 1.3. Типы данных. Переменные (1 ч.)</p>	<p>Данные и их типы. Целые числа (int). Числа с плавающей точкой (float). Строки (str).</p>	<p>теоретические занятия</p>	<p>(1 ч.)</p>
<p>создавать как прототипы программных систем, так и сами программные системы, помогает в интеграции программного обеспечения для решения производственных задач. Python имеет богатую стандартную библиотеку и большое количество модулей расширения</p>	<p>Тема 1.4. Ввод и вывод данных (2 ч.)</p>	<p><i>Задание к теме 1.4</i> Вывод данных. Функция print(). Ввод данных. Функция input()</p>	<p>практические занятия</p>	<p>(1 ч.)</p>
<p>создавать как прототипы программных систем, так и сами программные системы, помогает в интеграции программного обеспечения для решения производственных задач. Python имеет богатую стандартную библиотеку и большое количество модулей расширения</p>	<p>Тема 1.5. Логические выражения и операторы (1 ч.)</p>	<p>Изучить дополнительный материал по теме 1.4. Ввод и вывод данных https://pythontutor.ru/lessons/inout_and_arithmetic_operations/</p>	<p>самостоятельная работа</p>	<p>(1 ч.)</p>
<p>создавать как прототипы программных систем, так и сами программные системы, помогает в интеграции программного обеспечения для решения производственных задач. Python имеет богатую стандартную библиотеку и большое количество модулей расширения</p>	<p>Тема 1.6. Ветвление. Условный оператор (4 ч.)</p>	<p>Логические операторы. Сложные логические выражения.</p>	<p>теоретические занятия</p>	<p>(1 ч.)</p>
<p>создавать как прототипы программных систем, так и сами программные системы, помогает в интеграции программного обеспечения для решения производственных задач. Python имеет богатую стандартную библиотеку и большое количество модулей расширения</p>	<p>Тема 1.6. Ветвление. Условный оператор (4 ч.)</p>	<p>Условные операторы if, elif, else</p>	<p>теоретические занятия</p>	<p>(1 ч.)</p>
<p>создавать как прототипы программных систем, так и сами программные системы, помогает в интеграции программного обеспечения для решения производственных задач. Python имеет богатую стандартную библиотеку и большое количество модулей расширения</p>	<p>Тема 1.6. Ветвление. Условный оператор (4 ч.)</p>	<p><i>Задание к теме 1.6</i> Ветвление. Условный оператор</p>	<p>практические занятия</p>	<p>(2 ч.)</p>

<p>изучение языка не составляет большой проблемы. Написанные на нем программы получают структурированные по форме, и в них легко проследить логику работы.</p>		Изучить дополнительный материал по теме 1.6. Ветвление. Условный оператор https://metanit.com/python/tutorial/2.6.php	самостоятельная работа	(1 ч.)
	Тема 1.7. Исключения и их обработка в Python (2 ч.)	<i>Задание к теме 1.7</i> Исключения и их обработка в Python	практические занятия	(2 ч.)
	Тема 1.8. Циклы в программировании. Цикл while (4 ч.)	Цикл While, принципы работы структурного программирования	теоретические занятия	(1 ч.)
		<i>Задание к теме 1.8</i> Циклы в программировании. Цикл while	практические занятия	(2 ч.)
		Изучить дополнительный материал по теме 1.8. Циклы в программировании. Цикл while https://metanit.com/python/tutorial/2.7.php	самостоятельная работа	(1 ч.)
	Тема 1.9. Функции в программировании (1 ч.)	Определение функции. Оператор def. Вызов функции.	теоретические занятия	(1 ч.)
	Тема 1.10. Локальные и глобальные переменные (2 ч.)	<i>Задание к теме 1.9</i> Локальные и глобальные переменные	практические занятия	(2 ч.)

<p>Тема 1.11. Возврат значений из функции. Оператор return (2 ч.)</p>	<p><i>Задание к теме 1.11</i> Возврат значений из функции. Оператор return</p> <p>Изучить дополнительный материал по теме 1.11. Возврат значений из функции. Оператор return https://metanit.com/python/tutorial/2.16.php</p>	<p>практические занятия самостоятельная работа</p>	<p>(1 ч.) (1 ч.)</p>
<p>Тема 1.12. Параметры и аргументы функции (1 ч.)</p>	<p>Произвольное количество аргументов.</p>	<p>теоретические занятия</p>	<p>(1 ч.)</p>
<p>Тема 1.13. Случайные числа в Python – random, randint и randrange (2 ч.)</p>	<p><i>Задание к теме 1.13</i> Случайные числа в Python – random, randint и randrange</p>	<p>практические занятия</p>	<p>(2 ч.)</p>
<p>Тема 1.14. Списки (2 ч.)</p>	<p><i>Задание к теме 1.14</i> Списки</p>	<p>практические занятия</p>	<p>(2 ч.)</p>
<p>Тема 1.15. Цикл for (4 ч.)</p>	<p>Функция range(). Цикл for и range().</p>	<p>теоретические занятия</p>	<p>(1 ч.)</p>
	<p><i>Задание к теме 1.15</i> Цикл for</p>	<p>практические занятия</p>	<p>(2 ч.)</p>

	Изучить дополнительный материал по теме 1.15. Цикл for https://metanit.com/python/tutorial/2.7.php	самостоятельная работа	(1 ч.)
Тема 1.16. Функция enumerate (1 ч.)	Задание к теме 1.16 Функция enumerate	практические занятия	(1 ч.)
Тема 1.17. Строки (1 ч.)	Задание к теме 1.17 Строки	практические занятия	(1 ч.)
Тема 1.18. Словари (1 ч.)	Задание к теме 1.18 Словари	практические занятия	(1 ч.)
Тема 1.19. Функции (2 ч.)	Практическое задание №1 Функции	практические занятия	(1 ч.) (1 асс. ч.)
		Объем в ак.ч.	Объем в %
		теоретические занятия	9

ИТОГО:		практические занятия	20	55%
		самостоятельная работа	6	17%
		аттестация	1	
		Всего:	36	
Модуль 2. Машинное обучение на Python Область компьютерных наук, с помощью которой компьютерные системы могут придавать смысл. Задача в том, чтобы разобратся во всех данных. Предприятия и организации пытаются	Тема 2.1 Начало работы. Введение (3 ч.)	Введение в Машинное обучение на Python Изучить дополнительный материал по теме 2.1 Начало работы. Введение https://www.codecamp.ru/blog/machine-learning-python-tutorial/	теоретические занятия (2 ч.) самостоятельная работа (1 ч.)	
	Тема 2.2 Основные инструменты и библиотеки (5 ч.)	Инструменты и библиотеки в Python Задание к теме 2.2 Инструменты и библиотеки	теоретические занятия (1 ч.) практические занятия (3 ч.)	
		Изучить дополнительный материал по теме 2.2 Основные инструменты и библиотеки https://www.codecamp.ru/blog/machine-learning-python-tutorial/	самостоятельная работа (1 ч.)	

<p>справиться с этим путем создания интеллектуальных систем с использованием концепций и методологий из Machine learning. Машинное обучение — самая захватывающая область информатики. Лучший способ начать использовать Python для машинного обучения — это разобрать готовый проект по машинному обучению и затем его сделать самому с нуля:</p> <ul style="list-style-type: none"> ● Это заставит вас установить 	<p>Тема 2.3 Пример классификации и сортов ириса (4 ч.)</p>	<p>Классификация сортов ириса</p> <p><i>Задание к теме 2.3</i> Пример классификации сортов ириса</p>	<p>теоретические занятия</p> <p>практические занятия</p>	<p>(1 ч.)</p> <p>(3 ч.)</p>
	<p>Тема 2.4 Методы машинного обучения (4 ч.)</p>	<p>Методы машинного обучения в Python</p>	<p>теоретические занятия</p>	<p>(1 ч.)</p>
	<p>Тема 2.5 Предварительная обработка данных (5 ч.)</p>	<p>Предварительная обработка данных в Python</p> <p><i>Задание к теме 2.5</i> Предварительная обработка данных в Python</p>	<p>теоретические занятия</p> <p>практические занятия</p>	<p>(1 ч.)</p> <p>(3 ч.)</p>
	<p>Тема 2.6 Типы данных и конструирование признаков</p>	<p>Типы данных и конструирование признаков в Python</p> <p><i>Задание к теме 2.6.</i> Типы данных и конструирование признаков в Python</p>	<p>самостоятельная работа</p>	<p>(1 ч.)</p>
			<p>теоретические занятия</p> <p>практические занятия</p>	<p>(1 ч.)</p> <p>(3 ч.)</p>

<p>и запустить Python интерпретатор.</p> <ul style="list-style-type: none"> • Это даст вам понимание с высоты птичьего полета о том, как выглядит небольшой проект. • Это даст вам уверенность, чтобы перейти к собственным задачам и проектам. 	<p>(5 ч.)</p>	<p>Изучить дополнительный материал по теме 2.6 Типы данных и конструирование признаков https://habr.com/ru/company/skillfactory/blog/510420/</p>	<p>самостоятельная работа</p> <p>(1 ч.)</p>
<p>Тема 2.7 Работа с текстовыми данными (5 ч.)</p>	<p>Работа с текстовыми данными в Python</p> <p><i>Задание к теме 2.7</i> Работа с текстовыми данными в Python</p>	<p>Работа с текстовыми данными в Python</p>	<p>теоретические занятия</p> <p>(1 ч.)</p> <p>практические занятия</p> <p>(3 ч.)</p> <p>самостоятельная работа</p> <p>(1 ч.)</p>
<p>Тема 2.8 Общий подход к решению задач машинного обучения (5 ч.)</p>	<p>Изучение общего подхода к решению задач машинного обучения</p> <p>Практическое задание № 2 Решение задач машинного обучения</p>	<p>Изучить дополнительный материал по теме 2.7 Работа с текстовыми данными https://www.8host.com/blog/rabota-s-tekstovymi-fajlami-v-python-3/</p>	<p>теоретические занятия</p> <p>(1 ч.)</p> <p>практические занятия</p> <p>(2 ч.)</p> <p>(1 асс. ч.)</p> <p>самостоятельная работа</p> <p>(1 ч.)</p>

		ИТОГО:		теоретические занятия	9	
					20	55%
					6	17%
				аттестация	1	
				Всего:	36	
Модуль 3. Продвинутое программирование на языке Python Python является инструментом, с помощью которого изучаются различные области знаний. Python — это удобные структуры данных, методы программирования	Тема 3.1. Расширенное изучение Python (1 ч.)	Изучение Python (расширенно)	теоретические занятия (1 ч.)			
	Тема 3.2 Лексическая структура (1 ч.)	<i>Задание к теме 3.2</i> Лексическая структура	практические занятия (1 ч.)			
	Тема 3.3 Типы данных и значения (1 ч.)	Типы данных и значения в Python	теоретические занятия (1 ч.)			
	Тема 3.4	Переменные в Python	теоретические занятия (1 ч.)			

<p>и отладки компактных эффективных программ. Разработчики называют его лучшим языком для машинного обучения, потому что он сравнительно простой, гибкий и имеет хорошие фреймворки для работы с искусственным интеллектом. Модуль посвящен изучению более глубоким тем языка программирования Python.</p>	<p>Переменные (4 ч.)</p>	<p><i>Задание к теме 3.4</i> Переменные в Python</p> <p>Изучить дополнительный материал по теме 3.4 Переменные https://neurohive.io/ru/tutorial/primer-reshenija-realnoj-zadachi-po-mashinnomu-obucheniju-na-python/</p>	<p>практические занятия</p>	<p>(2 ч.)</p>	
	<p>Тема 3.5 Выражение и операторы (1 ч.)</p>	<p><i>Задание к теме 3.5</i> Выражение и операторы в Python</p>	<p>практические занятия</p>	<p>(1 ч.)</p>	
	<p>Тема 3.6 Инструкции (2 ч.)</p>	<p>Инструкции в Python</p> <p><i>Задание к теме 3.6</i> Инструкции в Python</p>	<p>теоретические занятия</p> <p>практические занятия</p>	<p>(1 ч.)</p> <p>(1 ч.)</p>	
	<p>Тема 3.7 Структуры данных: списки, кортежи, множества и словари (1 ч.)</p>	<p><i>Задание к теме 3.7</i> Структуры данных: списки, кортежи, множества и словари</p>	<p>практические занятия</p>	<p>(1 ч.)</p>	
	<p>Тема 3.8</p>	<p>Функции в Python</p>	<p>теоретические занятия</p>	<p>(1 ч.)</p>	

Функции (3 ч.)	<p><i>Задание к теме 3.8</i> Функции в Python</p> <p>Изучить дополнительный материал по теме 3.8 Функции https://pythonworld.ru/tipy-dannyx-v-python/vse-o-funkciyax-i-ix-argumentax.html</p>	практические занятия	(1 ч.)
Тема 3.9 Классы и объекты (3 ч.)	<p>Классы и объекты в Python</p> <p><i>Задание к теме 3.9</i> Классы и объекты в Python</p> <p>Изучить дополнительный материал по теме 3.9 Классы и объекты https://metanit.com/python/tutorial/7.1.php</p>	теоретические занятия	(1 ч.)
Тема 3.10 Модули и пакеты (1 ч.)	<p><i>Задание к теме 3.10</i> Модули и пакеты в Python</p>	практические занятия	(1 ч.)
Тема 3.11 Исключения и их обработка (2 ч.)	<p><i>Задание к теме 3.11</i> Исключения и их обработка</p>	практические занятия	(2 ч.)
Тема 3.12	Регулярные выражения	теоретические занятия	(1 ч.)

Регулярные выражения (2 ч.)	<i>Задание к теме 3.12</i> Регулярные выражения	практические занятия	(1 ч.)
Тема 3.13 Работа с файлами (1 ч.)	<i>Задание к теме 3.13</i> Работа с файлами	практические занятия	(1 ч.)
Тема 3.14 Ввод и вывод данных (2 ч.)	<i>Задание к теме 3.14</i> Ввод и вывод данных Изучить дополнительный материал по теме 3.14 Ввод и вывод данных https://younglinux.info/python/input	практические занятия самостоятельная работа	(1 ч.) (1 ч.)
Тема 3.15 Итераторы и генераторы (3 ч.)	Что такое итераторы и генераторы в Python	теоретические занятия	(1 ч.)
	<i>Задание к теме 3.15</i> Итераторы и генераторы	практические занятия	(1 ч.)
	Изучить дополнительный материал по теме 3.15 Итераторы и генераторы https://habr.com/ru/articles/337314/	самостоятельная работа	(1 ч.)
Тема 3.16 Декораторы (1 ч.)	<i>Задание к теме 3.16</i> Что такое декораторы в Python	практические занятия	(1 ч.)
Тема 3.17	<i>Задание к теме 3.17</i>	практические занятия	(1 ч.)

Многопоточно сть (1 ч.)	Многопоточность			
Тема 3.18 Веб-скрапинг (3 ч.)	Что такое Веб-скрапинг в Python	теоретические занятия	(1 ч.)	
	<i>Задание к теме 3.18</i> Веб-скрапинг	практические занятия	(1 ч.)	
Тема 3.19 Тестирование и отладка (1 ч.)	Изучить дополнительный материал по теме 3.18 Веб-скрапинг https://ru-brightdata.com/blog/how-to-scrapping-with-python	самостоятельная работа	(1 ч.)	
	<i>Задание к теме 3.19</i> Тестирование и отладка в Python	практические занятия	(1 ч.)	
Тема 3.20 Виртуальное окружение и управление зависимостям и (2 ч.)	Виртуальное окружение и управление зависимостями в Python	теоретические занятия	(1 ч.)	
	Практическое задание № 3 Создание программы для анализа текста	практические занятия	(1 асс. ч.)	
ИТОГО:		теоретические занятия	10	
		практические занятия	19	55%

Модуль 4. Python: Продвинутое искусство машинного обучения Машинное обучение сейчас является одной из самых востребованных областей в программировании. Алгоритмы машинного обучения могут выполнять задачи, которые раньше казались	Тема 4.1 Глубокое обучение и нейронные сети (2 ч.)	Глубокое обучение. Нейронные сети <i>Задание к теме 4.1</i> Глубокое обучение. Нейронные сети	теоретические занятия практические занятия теоретические занятия практические занятия	(1 ч.) (1 ч.) (1 ч.) (2 ч.)	6	17%	
	Тема 4.2 Обработка изображений в машинном обучении (3 ч.)	Обработка изображений в машинном обучении <i>Задание к теме 4.2</i> Обработка изображений в машинном обучении	теоретические занятия практические занятия	(1 ч.) (2 ч.)			
	Тема 4.3 Обучение без учителя (4 ч.)	Обучение без учителя в Python <i>Задание к теме 4.3</i> Обучение без учителя Изучить дополнительный материал по теме 4.3 Обучение без учителя	теоретические занятия практические занятия самостоятельная работа	(1 ч.) (2 ч.) (1 ч.)	36	1	
	Всего:						
	аттестация						
	самостоятельная работа						

<p>недоступными компьютерам.</p> <p>С помощью Python пишутся математические основы для построения нейронных сетей. Изучая данный модуль, вы поймете, чем отличается обучение с подкреплением от обучения без учителя, научитесь определять, какие технологии использовать для определенных наборов данных, научитесь использовать эти технологии на практике и писать код на Python для решения задач.</p>	<p>Тема 4.4</p> <p>Обучение с подкреплением (4 ч.)</p>	<p>https://www.youtube.com/live/O5uGRvgI0vQ?feature=share</p> <p>Обучение с подкреплением в Python</p> <p><i>Задание к теме 4.4</i></p> <p>Обучение с подкреплением</p> <p>Изучить дополнительный материал по теме 4.4</p> <p>Обучение с подкреплением</p> <p>https://habr.com/ru/company/guyvds/blog/450314/</p>	<p>теоретические занятия (1 ч.)</p> <p>практические занятия (2 ч.)</p> <p>самостоятельная работа (1 ч.)</p>		
	<p>Тема 4.5</p> <p>Рекомендательные системы (4 ч.)</p>	<p>Рекомендательные системы в Python</p> <p><i>Задание к теме 4.5</i></p> <p>Рекомендательные системы в Python</p> <p>Изучить дополнительный материал по теме 4.5</p> <p>Рекомендательные системы</p> <p>https://habr.com/ru/amp/publications/727228/</p>	<p>теоретические занятия (1 ч.)</p> <p>практические занятия (2 ч.)</p> <p>самостоятельная работа (1 ч.)</p>		
	<p>Тема 4.6</p> <p>Анализ данных и визуализация машинного обучения (4 ч.)</p>	<p>Анализ данных и визуализация</p> <p><i>Задание к теме 4.6</i></p> <p>Анализ данных и визуализация</p> <p>Изучить дополнительный материал по теме 4.6</p> <p>Анализ данных и визуализация</p>	<p>теоретические занятия (1 ч.)</p> <p>практические занятия (2 ч.)</p> <p>самостоятельная работа (1 ч.)</p>		

	https://habr.com/ru/articles/264339/		
Тема 4.7 Высоконагруженные системы и машинное обучение (3 ч.)	Высоконагруженные системы	теоретические занятия	(1 ч.)
	<i>Задание к теме 4.7</i> Высоконагруженные системы	практические занятия	(2 ч.)
Тема 4.8 Обработка полученных знаний (2 ч.)	<i>Задание к теме 4.8</i> Обработка полученных знаний	практические занятия	(1 ч.)
	Изучить дополнительный материал по теме 4.8 Обработка полученных знаний https://waksoft.susu.ru/2021/04/03/kak-sozdat-klassifikator-izobrazhenij-na-python-s-pomoshhyu-tensorflow-2-i-keras/	самостоятельная работа	(1 ч.)
Тема 4.9 Производительность и оптимизация моделей машинного обучения (4 ч.)	Производительность и оптимизация моделей машинного обучения	теоретические занятия	(1 ч.)
	<i>Задание к теме 4.9</i> Производительность и оптимизация моделей машинного обучения	практические занятия	(2 ч.)
	Изучить дополнительный материал по теме 4.9 Производительность и оптимизация моделей машинного обучения	самостоятельная работа	(1 ч.)

	https://www.specialist.ru/course/python32		
Тема 4.10 Практика на Python (4 ч.)	Практика. Программирование на Python	теоретические занятия	(1 ч.)
	Практическое задание № 4 Автоматический код загрузки для моделей MobileNet	практические занятия	(2 ч.) (1 асс. ч.)
Итоговая аттестация		Итоговая проектная работа	(2 асс. ч.)
	ИТОГО:		
		теоретические занятия	9
		практические занятия	18
		самостоятельная работа	6
		промежуточная аттестация	1
		Итоговая аттестация	2
		Всего:	36

Пошаговое руководство к выполнению заданий

Задание к теме 1.4

Вывод данных. Функция `print()`. Ввод данных. Функция `input()` (1 ч.)

1. Откройте ссылку https://pythontutor.ru/lessons/inout_and_arithmetic_operations/
2. Самостоятельно изучите материал по выводу данных. Функция `print()`. Ввод данных. Функция `input()`
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 1.6

Ветвление. Условный оператор (2 ч.)

1. Откройте ссылку <https://pythontutor.ru/lessons/ifelse/>
2. Самостоятельно изучите материал по ветвлению и условным операторам.
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 1.7

Исключения и их обработка в Python (2 ч.)

1. Откройте ссылку <https://metanit.com/python/tutorial/2.11.php>
2. Самостоятельно изучите материал по исключениям и их обработкам
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 1.8

Циклы в программировании. Цикл `while` (2 ч.)

1. Откройте ссылку <https://pythontutor.ru/lessons/while/>
2. Самостоятельно изучите материал по циклу `While`.
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 1.9

Локальные и глобальные переменные (2 ч.)

1. Откройте ссылку <https://codechick.io/tutorials/python/global-local-nonlocal-variables>
2. Самостоятельно изучите материал по локальным и глобальным переменным.
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 1.11

Возврат значений из функции. Оператор return (1 ч.)

1. Откройте ссылку <https://metanit.com/python/tutorial/2.16.php>
2. Самостоятельно изучите материал по оператору return
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 1.13

Случайные числа в Python – random, randint и randrange (2 ч.)

1. Откройте ссылку <https://metanit.com/python/tutorial/6.1.php>
2. Самостоятельно изучите материал по модулям random, randint и randrange
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 1.14

Списки (2 ч.)

1. Откройте ссылку <https://pythontutor.ru/lessons/lists/>
2. Самостоятельно изучите материал по спискам
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 1.15

Цикл for (2 ч.)

1. Откройте ссылку https://pythontutor.ru/lessons/for_loop/
2. Самостоятельно изучите материал по циклу for
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 1.16

Функция enumerate (1 ч.)

1. Откройте ссылку <https://digitology.tech/posts/python-funkciya-enumerate/>
2. Самостоятельно изучите материал по функции enumerate
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 1.17

Строки (1 ч.)

1. Откройте ссылку <https://pythontutor.ru/lessons/str/>
2. Самостоятельно изучите материал по строкам в Python
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 1.18

Словари (1 ч.)

1. Откройте ссылку <https://pythontutor.ru/lessons/dicts/>
2. Самостоятельно изучите материал по словарям в Python
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 2.2

Инструменты и библиотеки (3 ч.)

1. Откройте ссылку <https://pythonist.ru/top-8-bibliotek-python-dlya-mashinnogo-obucheniya-i-iskusstvennogo-intellekta/>
2. Самостоятельно изучите материал по инструментам и библиотекам
3. Установить библиотеку tensorflow на свой ПК. Самостоятельно найдите пакет Python pip

Задание к теме 2.3

Пример классификации сортов ириса (3 ч.)

1. Откройте ссылку <https://vk.com/@-176174488-postroenie-prosteishei-modeli-mashinnogo-obucheniya-na-osnov>
2. Самостоятельно изучите материал по классификации сортов ириса
3. Попробуйте выполнить работу, которая указана в материале под пунктом 1.

Задание к теме 2.4

Методы машинного обучения (3 ч.)

1. Откройте ссылку <https://trager.ru/translations/hitchhikers-guide-to-ml/>
2. Самостоятельно изучите материал по методам машинного обучения
3. Перейдите по ссылке <https://github.com/cdeweyx/Hitchhikers-Guide-Machine-Learning>
4. Попробуйте запустить проект указанный в пункте 3

Задание к теме 2.5

Предварительная обработка данных (3 ч.)

1. Откройте ссылку <https://machinelearningmastery.ru/data-preprocessing-in-python-b52b652e37d5/>
2. Самостоятельно изучите материал по предварительной обработке данных
3. Попробуйте импортировать библиотеки, которые указаны в материале к пункту 1

Задание к теме 2.6.

Типы данных и конструирование признаков (3 ч.)

1. Откройте ссылку <https://learn.microsoft.com/ru-ru/azure/machine-learning/how-to-configure-auto-features#feature-engineering-and-featurization>
2. Самостоятельно изучите материал по типам данных и конструированию признаков
3. Установите последнюю версию пакета `azureml-train-automl` в локальной среде.

Задание к теме 2.7

Работа с текстовыми данными (3 ч.)

1. Откройте ссылку <https://foxford.ru/wiki/informatika/rabota-s-tekstovymi-faylami-v-python>
2. Самостоятельно изучите материал по работе с текстовыми данными
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте простую программу по теме задания

Задание к теме 3.2

Лексическая структура (1 ч.)

1. Откройте ссылку <https://is42-2018.susu.ru/gaifulinavr/2020/11/29/leksika-programmy-python/>
2. Самостоятельно изучите информацию по лексической структуре языка Python
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Составьте три простые программы на свободную тему

Задание к теме 3.4

Переменные (2 ч.)

1. Откройте ссылку <https://metanit.com/python/tutorial/2.2.php>
2. Самостоятельно изучите информацию по переменным и типам данных в Python
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler

4. Составьте простую программу по теме задания

Задание к теме 3.5

Выражение и операторы (1 ч.)

1. Откройте ссылку <https://metanit.com/python/tutorial/2.4.php>
2. Самостоятельно изучите информацию.
3. Перейдите по ссылке на онлайн компилятор <https://metanit.com/python/tutorial/2.6.php>
4. Попрактикуйтесь на тему “*Выражение и операторы*”

Задание к теме 3.6

Инструкции (1 ч.)

1. Откройте ссылку <https://web.archive.org/web/20220906210456/https://python.ru.com/uroki/instrukcii-i-vyrazhenija-dlja-nachinajushhih>
2. Пройдите тренажер по теме “*Инструкции*”
3. Дополнительно изучите материал по теме <https://python.ru.com/uroki/instrukcii-i-vyrazhenija-dlja-nachinajushhih>

Задание к теме 3.7

Структуры данных: списки, кортежи, множества и словари (1 ч.)

1. Откройте ссылки
<https://metanit.com/python/tutorial/3.1.php>
<https://metanit.com/python/tutorial/3.2.php>
<https://metanit.com/python/tutorial/3.3.php>
<https://metanit.com/python/tutorial/3.4.php>
2. Пройдите тренажер по теме “*Структуры данных: списки, кортежи, множества и словари*”

Задание к теме 3.8

Функции (1 ч.)

1. Откройте ссылку <https://metanit.com/python/tutorial/2.8.php>
2. Пройдите тренажер по теме “Функции”

Задание к теме 3.9

Классы и объекты (1 ч.)

1. Откройте ссылку <https://metanit.com/python/tutorial/7.1.php>
2. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
3. Попрактикуйтесь на тему “Классы и объекты”

Задание к теме 3.10

Модули и пакеты (1 ч.)

1. Откройте ссылки
<https://metanit.com/python/tutorial/2.10.php>
https://ru.hexlet.io/courses/python-basics/lessons/packages/theory_unit
2. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
3. Попрактикуйтесь на тему “Модули и пакеты”

Модуль 3. Практическая работа № 8 Практическая работа к теме 3.11

Шаг 1. Создайте проект

Шаг 2. Решите задачу: Дана строка `'ahb acb aeb acsb adcb aхсb'`. Напишите регулярку, которая найдет строки `ahb`, `acb`, `аеb` по шаблону: буква `'a'`, любой символ, буква `'b'`.

Шаг 3. Выложите файл в Odin

Задание к теме 3.12

Регулярные выражения (1 ч.)

1. Откройте ссылку <https://habr.com/ru/articles/349860/>
2. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
3. Попрактикуйтесь на тему “Регулярные выражения”

Задание к теме 3.13

Работа с файлами (1 ч.)

1. Откройте ссылку <https://metanit.com/python/tutorial/4.1.php>
<https://metanit.com/python/tutorial/4.2.php>
2. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
3. Попрактикуйтесь на тему “Работа с файлами”

Задание к теме 3.14

Ввод и вывод данных (1 ч.)

1. Откройте ссылку <https://metanit.com/python/tutorial/2.14.php>
2. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
3. Попрактикуйтесь на тему “Ввод и вывод данных”

Задание к теме 3.15

Итераторы и генераторы (1 ч.)

1. Перейдите по ссылке на онлайн компилятор <https://habr.com/ru/articles/653121/>
<https://habr.com/ru/articles/320288/>
2. На сайте выберите Popular Templates
3. Внутри есть готовый файл с CSS, поэкспериментируйте с файлов и посмотрите на результат

Задание к теме 3.16

Декораторы (1 ч.)

1. Откройте ссылку и изучите материал <https://habr.com/ru/articles/141411/>
2. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
3. Попрактикуйтесь на тему “Декораторы”

Задание к теме 3.17

Многопоточность (1 ч.)

1. Откройте ссылку и изучите материал <https://habr.com/ru/articles/484446/>
2. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
3. Попрактикуйтесь на тему “Многопоточность”

Задание к теме 3.18

Веб-скрапинг (1 ч.)

1. Откройте ссылку и изучите материал <https://habr.com/ru/companies/vdsina/articles/537174/https://habr.com/ru/companies/vdsina/articles/537174/>
2. Скачайте SublimeText с официального сайта <https://www.sublimetext.com/3>
3. Попрактикуйтесь в создании XML файлов, как показано в уроке

Задание к теме 3.19

Тестирование и отладка (1 ч.)

1. Откройте ссылку, найдите главу 22 и изучите материал <https://habr.com/ru/companies/otus/articles/433358/>

2. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
3. Попробуйте по ссылке на тему “Тестирование и отладка”

Задание к теме 4.1

Глубокое обучение и нейронные сети (1 ч.)

1. Откройте ссылку <https://habr.com/ru/articles/148407/>
2. Самостоятельно изучите материал по теме
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Попробуйте вводить код в программу

Задание к теме 4.2

Обработка изображений в машинном обучении (2 ч.)

1. Откройте ссылку <https://habr.com/ru/companies/otus/articles/695462/>
2. Самостоятельно изучите материал по моделям в TensorFlow.js
3. Установите TensorFlow.js <https://www.tensorflow.org/install>

Задание к теме 4.3

Обучение без учителя (2 ч.)

1. Откройте ссылку <https://proglib.io/p/unsupervised-ml-with-python>
2. Самостоятельно изучите материал по TensorFlow.js с Node.js
3. Попробуйте попрактиковаться в выполнении работы из пункта 1

Задание к теме 4.4

Обучение с подкреплением (2 ч.)

1. Откройте ссылку <https://habr.com/ru/companies/piter/articles/434738/>
2. Самостоятельно изучите материал по установке библиотек TensorFlow

3. Загрузите библиотеку TensorFlow

Задание к теме 4.5

Рекомендательные системы (2 ч.)

1. Откройте ссылку <https://habr.com/ru/companies/piter/articles/350346/>
2. Самостоятельно изучите материал по загрузке моделей TensorFlow
3. Перейдите по ссылке <https://github.com/MihailSalnikov/NERjs>
4. Попробуйте самостоятельно запустить проект из пункта 3

Задание к теме 4.6

Анализ данных и визуализация машинного обучения (2 ч.)

1. Откройте ссылку <https://habr.com/ru/companies/ods/articles/323210/>
2. Самостоятельно изучите материал по загрузке моделей из файловой системы
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Попрактикуйтесь на тему “Анализ данных и визуализация машинного обучения”

Задание к теме 4.7

Высоконагруженные системы и машинное обучение (2 ч.)

1. Откройте ссылку <https://habr.com/ru/articles/451164/>
2. Самостоятельно изучите материал по загрузке моделей из файловой системы
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Попрактикуйтесь на тему “Высоконагруженные системы и машинное обучение”

Задание к теме 4.8

Обработка полученных знаний (1 ч.)

1. Откройте ссылку <https://habr.com/ru/articles/671116/>
2. Самостоятельно изучите материал по классификации изображений

3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Попробуйте

Задание к теме 4.9

Производительность и оптимизация моделей машинного обучения (2 ч.)

1. Откройте ссылку <https://habr.com/ru/companies/otus/articles/698370/>
2. Самостоятельно изучите материал
3. Перейдите по ссылке на онлайн компилятор https://www.onlinegdb.com/online_python_compiler
4. Попробуйте на тему “Производительность и оптимизация моделей машинного обучения”

Календарный график

Первый модуль с 01.10.2023 - 30.11.2023

Второй модуль: с 01.12.2023 - 31.01.2024

Третий модуль: с 01.02.2024 - 31.03.2024

Четвертый модуль: с 01.04.2024 - 31.05.2024

Занятия проводятся 2-3 раза в неделю по 2 академических часа, длительность одного академического часа – 45 минут.

Календарно-тематическое планирование

№	Тема и № модуля	Тема занятия	Кол-во занятий*	Кол-во часов	Дата
1	Модуль 1.	Тема 1.1. Знакомство с языком программирования Python	1	1	2.10.23

2	Python. Введение в программирование 01.10.2023 - 30.11.2023	Тема 1.2. PyChart Community. Основы работы	1	2	5.10.23
3		Тема 1.3. Типы данных. Переменные	1	1	5.10.23
4		Тема 1.4. Ввод и вывод данных	1	2	9.10.23
5		Тема 1.5. Логические выражения и операторы	1	1	9.10.23
6		Тема 1.6. Ветвление. Условный оператор	3	4	12.10.23-14.10.23
7		Тема 1.7. Исключения и их обработка в Python	2	2	16.10.23
8		Тема 1.8. Циклы в программировании. Цикл while	3	4	19.10.23-21.10.23
9		Тема 1.9. Функции в программировании	1	1	23.10.23
10		Тема 1.10. Локальные и глобальные переменные	2	2	25.10.23
11		Тема 1.11. Возврат значений из функции. Оператор return	1	2	27.10.23
12		Тема 1.12. Параметры и аргументы функции	1	1	7.11.23

13	Тема 1.13. Случайные числа в Python – random, randint и randrange	2	2	9.11.23
14	Тема 1.14. Списки	2	2	13.11.23
15	Тема 1.15. Цикл for	3	4	16.11.23-20.11.23
16	Тема 1.16. Функция enumerate	1	1	23.11.23
17	Тема 1.17. Строки	1	1	23.11.23
18	Тема 1.18. Словари	1	1	27.11.23
19	Тема 1.19. Функции	1	2	27.11.23
20	Модуль 2. Машинное обучение на Python	2	3	1.12.23
21	Тема 2.1 Начало работы. Введение	3	5	4.12.23-7.12.23
22	Тема 2.2 Основные инструменты и библиотеки	3	4	11.12.23-15.12.23
23	Тема 2.3 Пример классификации сортов ириса	3	4	18.12.23-22.12.23
	Тема 2.4 Методы машинного обучения	3	4	

24		Тема 2.5 Предварительная обработка данных	3	5	25.12.23-27.12.23
25		Тема 2.6 Типы данных и конструирование признаков	3	5	9.01.24-12.01.24
26		Тема 2.7 Работа с текстовыми данными	3	5	15.01.24-19.01.24
27		Тема 2.8 Общий подход к решению задач машинного обучения	3	5	22.01.24-26.01.24
28	Модуль 3. Продвинутое программирование на языке Python 01.02.2024 - 31.03.2024	Тема 3.1. Расширенное изучение Python	1	1	1.02.24
29		Тема 3.2 Лексическая структура	1	1	1.02.24
30		Тема 3.3 Типы данных и значения	1	1	3.02.24
31		Тема 3.4 Переменные	3	4	5.02.24-7.02.24
32		Тема 3.5 Выражения и операторы	1	1	9.02.24
33		Тема 3.6 Инструкции	2	2	12.02.24
34		Тема 3.7 Структуры данных: списки, кортежи, множества и словари	1	1	15.02.24

35	Тема 3.8 Функции	2	3	17.02.24
36	Тема 3.9 Классы и объекты	2	3	19.02.24
37	Тема 3.10 Модули и пакеты	1	1	22.02.24
38	Тема 3.11 Исключения и их обработка	1	2	22.02.24
39	Тема 3.12 Регулярные выражения	2	2	26.02.24
40	Тема 3.13 Работа с файлами	1	1	28.02.24
41	Тема 3.14 Ввод и вывод данных	2	2	2.03.24
42	Тема 3.15 Итераторы и генераторы	3	3	4.03.24-6.03.24
43	Тема 3.16 Декораторы	1	1	7.03.24
44	Тема 3.17 Многопоточность	1	1	11.03.24
45	Тема 3.18 Веб-скрапинг	3	3	13-03.24-16.03.24

46		Тема 3.19 Тестирование и отладка	1	1	18.03.24
47		Тема 3.20 Виртуальное окружение и управление зависимостями	2	2	21.03.24
48	Модуль 4. Python: Продвинутое искусство машинного обучения	Тема 4.1 Глубокое обучение и нейронные сети	2	2	2.04.24
49		Тема 4.2 Обработка изображений в машинном обучении	3	3	3.04.24-6.04.24
50	01.04.2024 - 31.05.2024	Тема 4.3 Обучение без учителя	4	4	8.04.24-12.04.24
51		Тема 4.4 Обучение с подкреплением	4	4	15.04.24-19.04.24
52		Тема 4.5 Рекомендательные системы	4	4	22.04.24-26.04.24
53		Тема 4.6 Анализ данных и визуализация	4	4	29.04.24-4.05.24
54		Тема 4.7 Высоконагруженные системы и машинное обучение	3	3	6.05.24-8.04.24

55	Тема 4.8	Обработка полученных знаний	2	2	11.05.24
56	Тема 4.9	Производительность и оптимизация моделей машинного обучения	4	4	13.05.24-17.05.24
57	Тема 4.10	Практика на Python	4	4	20.05.24-23.05.24
58	Итоговая аттестация			2	25.05.24

*количество занятий не включают часы, отведенные на самостоятельное изучение, и часы, отведенные на прохождение аттестации

Учебно-методические материалы

<i>Наименование поля</i>	<i>Допустимые значения полей</i>	<i>Значение полей</i>	<i>Значение полей</i>	<i>Значение полей</i>
<i>Порядковый номер модуля</i>	<i>строка не менее 10 символов</i>	<i>1</i>	<i>2</i>	<i>3</i>
				<i>4</i>

Методы, формы и технологии	<i>строка не менее 10 символов</i>	Синхронные занятия включают интерактивные лекции и практические занятия, предусматривающие групповую и индивидуальную работу учащихся. Для организации асинхронной работы учащихся используются записи синхронных занятий, презентации, конспекты лекций, практические задания и тестовые вопросы, размещаемые в электронном курсе на платформе электронного обучения LMS Odin.
Методические разработки	<i>строка не менее 10 символов</i>	Для изучения программы используется электронный курс в LMS Odin.
Материалы модуля	<i>строка не менее 10 символов</i>	Электронный курс включает информационно-организационные материалы по программе (учебно-тематический план, календарный график работы по программе, информацию о преподавателях программы, чат для объявлений и вопросов, сведения о результатах обучения), набор видеолекций, презентации к лекциям, набор ссылок на внешние ресурсы, систему заданий с инструкциями, списки основной и дополнительной литературы.
Учебная литература	<i>строка не менее 10 символов</i>	<p>Литература:</p> <ol style="list-style-type: none"> 1. Банкрашков А. Программирование для детей на языке Python. – Москва: АСТ, 2017. 2. Бриггс Дж. Python для детей. Самоучитель по программированию. – Москва: Манн, 2018. 3. Бэрри П. Изучаем программирование на Python. – Москва: Эксмо, 2017.

4. ВандерПлас Дж. Python для сложных задач. Наука о данных и машинное обучение. – Санкт-Петербург: Питер, 2018.
5. Васильев А. Python на примерах. Практический курс. Москва: Наука и техника, 2018.
6. Вордэрман К. Программирование на Python. Иллюстрированное руководство для детей. – Москва: Манн, Иванов и Фербер, 2018.
7. Гэддис Т. Начинаем программировать на Python. – Санкт-Петербург: БВХ-Петербург, 2019.
8. Доусон М. Программируем на Python. – Санкт-Петербург: Прогресс книга, 2019.
9. Жуков Р. Язык программирования Python: практикум. Учебное пособие. – Москва: Инфра-М, 2020.
10. Злагопольский Д.М. Основы программирования на языке Python. – Москва: ДМК Пресс, 2018.
11. Лутц М. Изучаем Python. В 2-х т. – Москва: Вильямс, 2019.
12. Любанович Б. Простой Python. Современный стиль программирования. – Санкт-Петербург: Питер, 2019.
13. МакГрат М. Программирование на Python для начинающих. Москва: Эксмо, 2015.
14. Маккини У. Python и анализ данных. – Москва: ДМК Пресс, 2018.
15. Мартелли А., Рейвенскрофт А., Холден С. Python. Справочник. Полное описание языка. – Москва: Диалектика, 2018.
16. Мэттиз Э. Изучаем Python. Программирование игр, визуализация данных, веб-приложения. – Санкт-Петербург: Питер, 2020.
17. Мюллер Дж.П. Python для чайников. – Москва: Вильямс, 2019.
18. Персиваль Г. Python. Разработка на основе тестирования. – Москва: ДМК Пресс, 2018.
19. Прадик Дж. Искусственный интеллект с примерами на Python. Создание приложений искусственного интеллекта с помощью Python для взаимодействия с окружающим миром. – Москва: Вильямс, 2019.

20. Пэйн Б. Python для детей и родителей. Игры и программируй. – Москва, Эксмо, 2018.
21. Рамальо Л. Python к вершинам мастерства. – Москва: ДМК Пресс, 2016.
22. Свейгарт Э. Автоматизация рутинных задач с помощью Python. Практическое руководство для начинающих. – Москва: Вильямс, 2018.
23. Свейгарт Э. Учим Python, делая крутые игры. – Москва: Бомбора, 2018.
24. Солем Я.Э. Программирование компьютерного зрения на языке Python. – Москва: ДМК Пресс, 2016.
25. Томашевский П. Привет, Python. Моя первая книга по программированию. - Москва: Наука и техника, 2018.
26. Харрисон М. Как устроен Python. – Санкт-Петербург: Питер, 2019

Материально-технические условия реализации программы

<i>Наименование поля</i>	<i>Допустимые значения полей</i>	<i>Значение полей</i>	<i>Значение полей</i>	<i>Значение полей</i>
<i>Порядковый номер модуля</i>	<i>строка не менее 10 символов</i>	<i>1</i>	<i>2</i>	<i>3</i>
<i>Наименование требуемого оборудования</i>	<i>строка не менее 2 символов</i>	<p>- персональный компьютер (рекомендуется) / мобильный телефон / планшет; наушники, микрофон и камера (обязательно); -сеть: скорость соединения от 2 Мб/с.</p>		
<i>Наименование требуемого программного обеспечения</i>	<i>строка не менее 2 символов</i>	<p>операционная система семейства MacOS или Windows;</p> <ul style="list-style-type: none"> ● программа Python; ● файловый менеджер (в составе операционной системы или др.); ● почтовый клиент (в составе операционных систем или др.); ● браузер (в составе операционных систем или др.); ● мультимедиа проигрыватель (в составе операционной системы или др.); ● антивирусная программа; ● программа-архиватор; ● программа-переводчик; ● программа интерактивного общения; ● текстовый редактор; ● растровый графический редактор; ● звуковой редактор; 		

		<ul style="list-style-type: none"> • редактор Web-страниц.
<p>Электронные информационные ресурсы</p>	<p><i>строка не менее 10 символов</i></p>	<p>Курс включает информационно-организационные материалы по программе (учебно-тематический план, календарный график работы по программе, информацию о преподавателях программы, чат для объявлений и вопросов, сведения о результатах обучения), набор видеолекций, презентации к лекциям, набор ссылок на внешние ресурсы, систему заданий с инструкциями, списки основной и дополнительной литературы.</p>
<p>Электронные образовательные ресурсы</p>	<p><i>строка не менее 10 символов</i></p>	<p>Курс включает набор ссылок на внешние ресурсы.</p>