

Министерство образования и науки Смоленской области
Отдел образования администрации муниципального образования
"Починковский район" Смоленской области
МБОУ Переснянская СШ

Принята на заседании
педагогического совета
от « 30 » 08 2024 г. Протокол № 1

Утверждаю:
Директор МБОУ Переснянская СШ
_____/С.В. Анашкина/
Приказ № 163-а от 30.08.2024 г.

Дополнительная общеобразовательная общеразвивающая программа
технической направленности

«Основы программирования на PYTHON»

**Программа реализуется в Центре образования естественно-научного и
технологического профилей «Точка роста»**

Возраст обучающихся: 13–15 лет

Срок реализации: 1 года

Автор-составитель:
Яковлева Анна Александровна,
Учитель информатики

Пересна
2024г.

Пояснительная записка

Рабочая программа кружка по информатике «Основы программирования на PYTHON» составлена в соответствии:

1. Федерального государственного образовательного стандарта основного общего образования
2. Основной образовательной программы основного общего образования
3. Учебного плана МБОУ Переснянской СШ

Актуальность программы. «Основы программирования на Python» отражает:

1. сущность информатики как научной дисциплины, изучающей закономерности протекания и возможности автоматизации информационных процессов в различных системах;
2. основные области применения информатики, прежде всего информационные технологии, управление и социальную сферу;
3. междисциплинарный характер информатики и информационной деятельности.

Информатика характеризуется всё возрастающим числом междисциплинарных связей, причём как на уровне понятийного аппарата, так информационных технологий как необходимого инструмента практически любой и на уровне инструментария. Современная школьная информатика оказывает существенное влияние на формирование мировоззрения школьника, его жизненную позицию, закладывает основы понимания принципов функционирования и использования деятельности и одного из наиболее значимых технологических достижений современной цивилизации. Многие предметные знания и способы деятельности, освоенные обучающимися при изучении информатики, находят применение как в рамках образовательного процесса при изучении других предметных областей, так и в иных жизненных ситуациях, становятся значимыми для формирования качеств личности, т. е. ориентированы на формирование метапредметных и личностных результатов обучения.

Курс отражает и расширяет содержание четырёх тематических разделов информатики на уровне основного общего образования:

- 1) цифровая грамотность;
- 2) теоретические основы информатики;
- 3) алгоритмы и программирование;
- 4) информационные технологии.

Программа «Основы программирования на Python», предназначенная для работы с учащимися 13-15 лет, дает возможность учащимся познакомиться с новыми направлениями развития средств информационных технологий и получить практические навыки.

Содержание программы. Программа курса «Основы программирования на Python» построена по модульному принципу. Все содержание программы разбито на 8 основных тем. Каждая тема содержит логически связанный учебный материал (теоретический материал, практические задания, задания для самостоятельной работы – выполнение индивидуального или группового проекта), который носит законченный характер. На последнем занятии каждой темы проводится конференция, на которой учащиеся представляют свои работы и обсуждают их.

Содержание обучения, представленное в программе курса, позволяет вести обучение школьников в режиме актуального познания. Практическая направленность курса на создание внешних образовательных продуктов — блок-схем, алгоритмов, программ — способствует выявлению фактов, которые невозможно объяснить на основе имеющихся у школьников знаний. Возникающие при этом познавательные переживания обуславливают сознательное отношение к изучению основных теоретических положений информатики.

Проявления трудолюбия, целеустремленности, возникающие при воплощении замыслов учащихся в рамках курса, стимулируют развитие индивидуально-личностных качеств школьников.

Активизация познавательного процесса позволяет учащимся более полно выражать свой творческий потенциал и реализовывать собственные идеи в изучаемой области знаний, создаёт

предпосылки по применению освоенных навыков программирования в других учебных курсах, а также способствует возникновению дальней мотивации, направленной на освоение профессий, связанных с разработкой программного обеспечения.

Курс служит средством внутрипрофильной специализации в области новых информационных технологий, что способствует созданию дополнительных условий для проявления индивидуальных образовательных интересов учащихся.

Тематическое и поурочное планирование и структура каждой темы осуществляет принцип от простого к сложному: изучение всех последующих тем обеспечено предыдущими темами или знаниями, полученными в базовом курсе информатики.

Направленность программы – техническая

Программа рассчитана на 68 часов

Форма организации образовательного процесса: очная. В связи с эпидемиологической обстановкой форма обучения может быть в дистанционном формате.

Вид занятий: лекции, практические и лабораторные работы, круглые столы, мастер-классы, мастерские, тематические занятия, выполнение самостоятельной работы, выставки, творческие отчеты (защита проектов).

Срок освоения программы – 1 год

Режим занятий: занятия проводятся 1 раза в неделю продолжительностью 3 часа в неделю, всего 102 часа в год.

Изменение взглядов на предмет информатики как науки, её место в системе научного знания требует существенных изменений в содержании образования по информатике. В связи с этим особую актуальность приобретают раскрытие личностных резервов учащихся и создание соответствующей среды.

Никакая система задач, какой бы хорошей она ни была, никакие тренинги памяти, внимания и т. п. не дают того эффекта, который возникает в случае, если учащиеся осознают необходимость решения тех или иных задач, если у них появляется острая необходимость к преодолению интеллектуальных трудностей, связанных с познанием, если они видят смысл в сотрудничестве с одноклассниками и учителем.

Отличительные особенности программы.

Материально-техническое и программное обеспечение нашей школы позволяет на практике познакомить школьников с основами компьютерных технологий, подготовить их к жизни и работе в условиях информационно развитого общества.

Ключевой особенностью курса является его направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и реализации алгоритма с помощью средств программирования.

В рамках предлагаемого курса изучение основ программирования на языке Python — это не столько средство подготовки к будущей профессиональной деятельности, сколько формирование новых общеинтеллектуальных умений и навыков: разделение задачи на этапы решения, построение алгоритма и др. Исключительно велика роль программирования для формирования мышления школьников, приёмов умственных действий, умения строить модели, самостоятельного нахождения и составления алгоритмов решения задач, умения чётко и лаконично реализовывать этапы решения задач. Использование этих возможностей для формирования общеинтеллектуальных и общеучебных умений школьников активизирует процесс индивидуально-личностного становления учащихся.

Общепедагогическая направленность занятий – гармонизация индивидуальных и социальных аспектов обучения по отношению к информационным технологиям. Умение составлять алгоритмы решения и навыки программирования являются элементами информационной компетенции — одной из ключевых компетенций современной школы. Умение находить решение, составлять алгоритм решения и реализовать его с помощью языков программирования — необходимое условие подготовки современных школьников. Особая роль отводится широко представленной в курсе системе рефлексивных заданий. Освоение рефлексии направлено на осознание учащимися того важного обстоятельства, что наряду с

разрабатываемыми ими продуктами в виде программ на компьютере рождается основополагающий образовательный продукт: освоенный инструментарий. Именно этот образовательный продукт станет базой для творческого самовыражения учащихся в форме различных программ.

Цели изучения курса:

- понять значение алгоритмизации как метода познания окружающего мира, принципы структурной алгоритмизации;
- овладеть базовыми понятиями теории алгоритмов;
- научиться разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python.

Задачи курса:

- познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
- научить составлять и читать блок-схемы;
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- изучить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками, кортежами, словарями, множествами);
- научить применять функции при написании программ на языке программирования Python;
- научить отлаживать и тестировать программы, делать выводы о работе этих программ.

Методы обучения

Отбор методов обучения обусловлен необходимостью формировать информационную и коммуникативную компетентности учащихся, реализовывать лично-ориентированное обучение, направлять их на самостоятельное решение разнообразных проблем, развивать исследовательские и творческие способности. Решение данных задач кроется в организации деятельностного подхода к обучению, в проблемном изложении материала учителем, в переходе от репродуктивного вида работ к самостоятельным, поисково-исследовательским видам деятельности. Поэтому основная методическая установка в данном курсе — обучение учащихся навыкам самостоятельной творческой деятельности.

В кружке могут заниматься дети с ОВЗ при наличии медицинской справки, а также одаренные дети, оказавшиеся в трудной жизненной ситуации.

Планируемые результаты

ЛИЧНОСТНЫЕ РЕЗУЛЬТАТЫ

Патриотическое воспитание:

- ценностное отношение к отечественному культурному, историческому и научному наследию;
- понимание значения информатики как науки в жизни современного общества.

Духовно-нравственное воспитание:

- ориентация на моральные ценности и нормы в ситуациях нравственного выбора;
- готовность оценивать своё поведение и поступки, а также поведение и поступки других людей с позиции нравственных и правовых норм с учётом осознания последствий поступков;
- активное неприятие асоциальных поступков, в том числе в Интернете.

Гражданское воспитание:

- представление о социальных нормах и правилах межличностных отношений в коллективе, в том числе в социальных сообществах;
- соблюдение правил безопасности, в том числе навыков безопасного поведения в интернет среде;
- ориентация на совместную деятельность при выполнении учебных и познавательных задач, создании учебных проектов;

- стремление оценивать своё поведение и поступки своих товарищей с позиции нравственных и правовых норм с учётом осознания последствий поступков.

Ценность научного познания:

- наличие представлений об информации, информационных процессах и информационных технологиях, соответствующих современному уровню развития науки и общественной практики;
- интерес к обучению и познанию;
- любознательность;
- стремление к самообразованию;
- овладение начальными навыками исследовательской деятельности, установка на осмысление опыта, наблюдений, поступков и стремление совершенствовать пути достижения индивидуального и коллективного благополучия;
- наличие базовых навыков самостоятельной работы с учебными текстами, справочной литературой, разнообразными средствами информационных технологий, а также умения самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учёбе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности.

Формирование культуры здоровья:

установка на здоровый образ жизни, в том числе и за счёт освоения и соблюдения требований безопасной эксплуатации средств ИКТ.

Трудовое воспитание:

интерес к практическому изучению профессий и труда в сферах деятельности, связанных с информатикой, программированием и информационными технологиями, основанными на достижениях науки информатики и научно-технического прогресса.

Экологическое воспитание:

наличие представлений о глобальном характере экологических проблем и путей их решения, в том числе с учётом возможностей ИКТ.

Адаптация обучающегося к изменяющимся условиям социальной среды:

освоение обучающимися социального опыта, основных социальных ролей, соответствующих ведущей деятельности возраста, норм и правил общественного поведения, форм социальной жизни в группах и сообществах, в том числе в виртуальном пространстве.

МЕТАПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ

Универсальные познавательные действия

Базовые логические действия:

- умение определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логические рассуждения, делать умозаключения (индуктивные, дедуктивные и по аналогии) и выводы;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;
- самостоятельно выбирать способ решения учебной задачи (сравнивать несколько вариантов решения, выбирать наиболее подходящий с учётом самостоятельно выделенных критериев).

Базовые исследовательские действия:

- формулировать вопросы, фиксирующие разрыв между реальным и желательным состоянием ситуации, объекта, и самостоятельно устанавливать искомое и данное;
- оценивать применимость и достоверность информации, полученной в ходе исследования;

- прогнозировать возможное дальнейшее развитие процессов, событий и их последствия в аналогичных или сходных ситуациях, а также выдвигать предположения об их развитии в новых условиях и контекстах.

Работа с информацией:

- выявлять дефицит информации, данных, необходимых для решения поставленной задачи;
- применять основные методы и инструменты при поиске и отборе информации из источников с учётом предложенной учебной задачи и заданных критериев;
- выбирать, анализировать, систематизировать и интерпретировать информацию различных видов и форм представления;
- выбирать оптимальную форму представления информации и иллюстрировать решаемые задачи несложными схемами, диаграммами, иными графическими объектами и их комбинациями;
- оценивать достоверность информации по критериям, предложенным учителем или сформулированным самостоятельно;
- запоминать и систематизировать информацию.

Универсальные коммуникативные действия

Общение:

- сопоставлять свои суждения с суждениями других участников диалога, обнаруживать различие и сходство позиций;
- публично представлять результаты выполненного опыта (исследования, проекта);
- выбирать формат выступления с учётом задач презентации и особенностей аудитории и в соответствии с ним составлять устные и письменные тексты с использованием иллюстративных материалов.

Совместная деятельность (сотрудничество):

- понимать и использовать преимущества командной и индивидуальной работы при решении конкретной проблемы, в том числе при создании информационного продукта;
- принимать цель совместной информационной деятельности по сбору, обработке, передаче, формализации информации; коллективно строить действия по её достижению: распределять роли, договариваться, обсуждать процесс и результат совместной работы;
- выполнять свою часть работы с информацией или информационным продуктом, достигая качественного результата по своему направлению и координируя свои действия с другими членами команды;
- оценивать качество своего вклада в общий информационный продукт по критериям, самостоятельно сформулированным участниками взаимодействия;
- сравнивать результаты с исходной задачей и вклад каждого члена команды в достижение результатов, разделять сферу ответственности и проявлять готовность к предоставлению отчёта перед группой.

Универсальные регулятивные действия

Самоорганизация:

выявлять в жизненных и учебных ситуациях проблемы, требующие решения;

составлять алгоритм решения задачи (или его часть), выбирать способ решения учебной задачи с учётом имеющихся ресурсов и собственных возможностей, аргументировать выбор варианта решения задачи;

составлять план действий (план реализации намеченного алгоритма решения), корректировать предложенный алгоритм с учётом получения новых знаний об изучаемом объекте.

Самоконтроль (рефлексия):

владеть способами самоконтроля, самомотивации и рефлексии;

учитывать контекст и предвидеть трудности, которые могут возникнуть при решении учебной задачи, адаптировать решение к меняющимся обстоятельствам;

вносить коррективы в деятельность на основе новых обстоятельств, изменившихся ситуаций, установленных ошибок, возникших трудностей;

оценивать соответствие результата цели и условиям.

Эмоциональный интеллект:

ставить себя на место другого человека, понимать мотивы и намерения другого.

Принятие себя и других:

осознавать невозможность контролировать всё вокруг даже в условиях открытого доступа к любым объёмам информации;

осознанно относиться к другому человеку, его мнению.

ПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ

По завершению данного курса учащиеся **должны знать:**

- соблюдать требования безопасности при работе на компьютере;
- объяснять, что такое информация, информационный процесс;
- перечислять виды информации;
- кодировать и декодировать сообщения по заданным правилам;
- переводить данные из одной единицы измерения информации в другую;
- характеризовать устройство компьютера;
- приводить примеры устройств для хранения и передачи информации;
- разбираться в структуре файловой системы;
- строить путь к файлу;
- объяснять, что такое алгоритм, язык программирования, программа;
- использовать переменные различных типов при написании программ на Python;
- использовать оператор присваивания при написании программ на Python;
- искать ошибки в программном коде на Python и исправлять их;
- дописывать программный код на Python;
- писать программный код на Python;
- использовать ветвления и циклы при написании программ на Python;
- анализировать блок-схемы и программы на Python;
- объяснять, что такое логическое выражение;
- вычислять значение логического выражения;
- записывать логическое выражение на Python;
- понимать структуру адресов веб-ресурсов;
- форматировать и редактировать текстовую информацию в Google Документах;
- создавать презентации в Google Презентациях.
- выделять основные этапы в истории развития информационных технологий и персонального компьютера;
- понимать принцип работы архитектуры Неймана;
- искать информацию в Интернете;
- форматировать и редактировать текстовую информацию в Google Документах;
- открывать доступ к презентации в Google Презентациях для совместной работы;
- писать программы на Python для рисования различных геометрических фигур, используя модуль Turtle;
- понимать различия локальных и глобальных переменных;
- решать задачи с использованием глобальных переменных на Python;
- строить таблицы истинности для логических выражений;
- строить логические схемы;
- понимать, что такое событие;
- использовать события при написании программ на Python;
- искать ошибки в программном коде на Python и исправлять их;
- дописывать программный код на Python;

- писать программный код на Python;
- писать свои функции на Python;
- разбивать задачи на подзадачи;
- анализировать блок-схемы и программы на Python.
- писать программы на Python по обработке числовых последовательностей;
- использовать списки и словари при написании программ на Python;
- искать ошибки в программном коде на Python и исправлять их;
- дописывать программный код на Python;
- писать программный код на Python;
- разбивать задачи на подзадачи;
- анализировать блок-схемы и программы на Python;
- разрабатывать веб-страницы, содержащие рисунки, списки и гиперссылки;
- защищать персональную информацию от несанкционированного доступа;
- предупреждать вовлечение себя и окружающих в деструктивные формы сетевой активности, такие как кибербуллинг.

Условия реализации программы. Использование оборудования центра естественно-научной и технологической направленностей «Точка роста»

Формы и методы отслеживания промежуточного результата: зачетные занятия, контрольные работы, участие в конкурсах и выставках, отчетные собрания.

Формы аттестации/контроля: Разрабатывают и защищают собственный проект или групповой проект и представляют его на итоговых занятиях.

Учебный план

Наименование тем	Количество часов	Практ. занятия
Тема 1. Знакомство с языком Python	10	10
Тема 2. Переменные и выражения	12	12
Тема 3. Условные предложения	12	12
Тема 4. Циклы	14	12
Тема 5. Функции	14	12
Тема 6. Строки - последовательности символов	14	12
Тема 7. Сложные типы данных	14	12
Тема 8. Стил программирования и отладка программ	12	10
ВСЕГО	102	92

Содержание учебного плана

Тема 1. Знакомство с языком Python

Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python. Комментарии.

Практическая работа 1.1. Установка программы Python Практическая работа 1.2. Режимы работы с Python Тест № 1. Знакомство с языком Python Учащиеся должны знать / понимать:

- понятие программы;
- структура программы на Python;
- режимы работы с Python.

Тема 2. Переменные и выражения

Типы данных . Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова. Выражения. Операции. Порядок выполнения операций. Математические функции. Композиция.

Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.

Практическая работа 2.1. Работа со справочной системой Практическая работа 2.2. Переменные Практическая работа 2.3. Выражения

Практическая работа 2.5. Задачи на элементарные действия с числами

Тест № 2. Выражения и операции.

Тема 3. Условные предложения

Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

Практическая работа 3.1. Логические выражения Практическая работа 3.2. "Условный оператор" Практическая работа 3.3. Множественное ветвление Практическая работа 3.4. "Условные операторы"

Самостоятельная работа № 1. Решение задач по теме "Условные операторы".

Зачетная работа № 1. "Составление программ с ветвлением".

Тест № 3. "Условные операторы".

Тема 4. Циклы

Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов.

Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом.

Практическая работа 4.1. "Числа Фибоначчи" Практическая работа 4.2. Решение задачи с циклом for. Практическая работа 4.3. Реализация циклических алгоритмов Практическая работа 4.4. Случайные числа Практическая работа 4.5. Решение задач с циклом. Самостоятельная работа № 2 "Составление программ с циклом"

Тест № 4. Циклы

Творческая работа № 1. "Циклы"

Тема 5. Функции

Создание функций. Параметры и аргументы. Локальные и глобальные переменные.

Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Примеры решения задач с использованием функций.

Рекурсивные функции. Вычисление факториала. Числа Фибоначчи.

Практическая работа 5.1. Создание функций

Практическая работа 5.2. Локальные переменные

Практическая работа 5.3. Решение задач с использованием функций

Практическая работа 5.4. Рекурсивные функции

Самостоятельная работа № 3 по теме "Функции"

Тест № 5. Функции

Тема 6. Строки - последовательности символов

Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки.

Срезы строк. Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи). Примеры решения задач со строками.

Практическая работа 6.1. Строки

Практическая работа 6.2. Решение задач со строками.

Тема 7. Сложные типы данных

Списки. Тип список (list). Индексы. Обход списка. Проверка вхождения в список.

Добавление в список. Суммирование или изменение списка. Операторы для списков.

Срезы списков. Удаление списка. Клонирование списков.

Списочные параметры.

Функция range. Списки: примеры решения задач.

Матрицы. Вложенные списки. Матрицы. Строки и списки. Генераторы списков в Python.

Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения. Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.

Множества в языке Python. Множества. Множественный тип данных. Описание множеств.

Операции, допустимые над множествами: объединение, пересечение, разность, включение.

Оператор определения принадлежности элемента множеству.

Практическая работа 7.1. Списки.

Практическая работа 7.2. Решение задач со списками.

Тест № 7. Списки

Тема 8. Стиль программирования и отладка программ

Стиль программирования. Отладка программ.

Зачет по курсу «Программирование на языке Python»

Календарный учебный график

Наименование тем	Количество часов всего	Практ. занятия
Тема 1. Знакомство с языком Python	10	10
Урок 1. Общие сведения о языке	2	2
Практическая работа 1.1. Установка программы Python		2
Урок 2. Режимы работы	2	2
Практическая работа 1.2. Режимы работы с Python		2
Тест № 1. Знакомство с языком Python		
Тема 2. Переменные и выражения	12	12
Урок 3. Переменные	2	
Практическая работа 2.1. Работа со справочной системой		2
Практическая работа 2.2. Переменные		2
Урок 4. Выражения	2	
Практическая работа 2.3. Выражения		2
Урок 5. Ввод и вывод	2	
		2
Урок 6. Задачи на элементарные действия с числами	2	
Практическая работа 2.5. Задачи на элементарные действия с числами		2
Тест № 2. Выражения и операции.		
Тема 3. Условные предложения	12	12
Урок 7. Логические выражения и операторы	2	
Практическая работа 3.1. Логические выражения		2
Урок 8. Условный оператор	2	
Практическая работа 3.2. "Условный оператор"		2
Урок 9. Множественное ветвление	2	
Практическая работа 3.3. Множественное ветвление		2
Урок 10. Реализация ветвления в языке Python	2	
Практическая работа 3.4. "Условные операторы"		2
Самостоятельная работа № 1 по теме "Условные операторы".		2
Урок 11. Зачетная работа № 1. "Составление программ с ветвлением".	2	
Тест № 3. "Условные операторы".		
Тема 4. Циклы	14	12
Урок 12. Оператор цикла с условием	2	
Практическая работа 4.1. "Числа Фибоначчи"		2
Урок 13. Оператор цикла for	2	
Практическая работа 4.2. Решение задачи с циклом for.		2
Урок 14. Вложенные циклы	2	
Практическая работа 4.3. Реализация циклических алгоритмов		2
Урок 15. Случайные числа	2	
Практическая работа 4.4. Случайные числа		2
Урок 16. Примеры решения задач с циклом	2	

Практическая работа 4.5. Решение задач с циклом.		2
Самостоятельная работа № 2 "Составление программ с циклом"		2
Тест № 4. Циклы		
Урок 17. Творческая работа № 1. "Циклы"		2
Тема 5. Функции	14	12
Урок 18. Создание функций	2	
Практическая работа 5.1. Создание функций		2
Урок 19. Локальные переменные	2	
Практическая работа 5.2. Локальные переменные		2
Урок 20. Примеры решения задач с использованием функций	2	
Практическая работа 5.3. Решение задач с использованием функций		2
Самостоятельная работа № 3 по теме "Функции"		2
Урок 21. Рекурсивные функции	2	
Практическая работа 5.4. Рекурсивные функции		2
Тест № 5. Функции		
Тема 6. Строки - последовательности символов	14	12
Урок 22. Строки	2	
Урок 23. Срезы строк	2	
Практическая работа 6.1. Строки		2
Урок 24. Примеры решения задач со строками	2	
Практическая работа 6.2. Решение задач со строками.		2
Тема 7. Сложные типы данных	14	12
Урок 25. Списки	2	
Урок 26. Срезы списков	2	
Практическая работа 7.1. Списки		2
Урок 27. Списки: примеры решения задач	2	
Практическая работа 7.2. Решение задач со списками		2
Урок 28. Матрицы	2	
Тест № 7. Списки	2	
Урок 29. Кортежи		
Урок 30. Введение в словари		
Урок 31. Множества в языке Python		
Тема 8. Стиль программирования и отладка программ	12	10
Урок 32. Стиль программирования	2	2
Урок 33. Отладка программ	2	
Урок 34. Зачет по курсу «Программирование на языке Python»		2
		2
Урок 35. Что дальше?	2	
ВСЕГО	102	92

Методическое обеспечение программы

Помодульные дидактические материалы, представленные на образовательной платформе (в том числе раздаточный материал и т. д.).

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ДЛЯ УЧИТЕЛЯ

Методические материалы.

Демонстрационные материалы по теме занятия.

Методическое видео с подробным разбором материалов, рекомендуемых для использования на занятии.

ЦИФРОВЫЕ ОБРАЗОВАТЕЛЬНЫЕ РЕСУРСЫ И РЕСУРСЫ ИНТЕРНЕТА

Образовательная платформа.

УЧЕБНОЕ ОБОРУДОВАНИЕ

Компьютер (стационарный компьютер, ноутбук, планшет).

УЧЕБНОЕ ОБОРУДОВАНИЕ ДЛЯ ПРОВЕДЕНИЯ

ЛАБОРАТОРНЫХ РАБОТ, ПРАКТИЧЕСКИХ РАБОТ И ДЕМОНСТРАЦИЙ

Мультимедийный проектор с экраном (интерактивной доской) или интерактивная панель.

Список литературы

1. Домашняя страница Python www.python.org . Справочные материалы, официальная документация.
2. Сайт проекта Интуит: Национальный открытый университет, курс «Введение в программирование на Python»,
<http://www.intuit.ru/studies/courses/12179/1172/info>.
3. Сайт проекта Интуит: Национальный открытый университет. Курс «Язык программирования Python» <http://www.intuit.ru/studies/courses/49/49/info>.
4. Сайт проекта Open Book Project openbookproject.net содержит серию практических примеров на Python Криса Мейерса.
5. Python. Подробный справочник Дэвида М. Бизли — книга со справочной информацией о языке Python и модулях стандартной библиотеки.
6. Python. Справочник Марка Лутца. Справочник по наиболее часто используемым функциям и модулям.