

Муниципальное бюджетное общеобразовательное учреждение
«Средняя общеобразовательная школа п. Опытный»
Цивильского муниципального округа Чувашской Республики

«УТВЕРЖДАЮ»

Директор школы:

_____/Фадеева Г.Г./
Приказ № 203 от 30 августа 2023г.

«СОГЛАСОВАНО»

Заместитель директора по ОД
_____/ Хвостенкова О.А.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ПРОГРАММА ТЕХНИЧЕСКОЙ НАПРАВЛЕННОСТИ

«Программирование»

Возраст обучающихся: 13-17 лет

Автор – составитель: Федоров Сергей Юрьевич, педагог дополнительного
образования

Срок реализации: 2023-2024 учебный год

Пояснительная записка

Направленность программы - техническая

Рабочая программа «Программирование» составлена в соответствии с требованиями Федерального государственного образовательного стандарта основного общего образования, Основной образовательной программой основного общего образования Федерального государственного образовательного стандарта среднего общего образования, Основной образовательной программой среднего общего образования МБОУ «СОШ п. Опытный».

Актуальность, педагогическая целесообразность

В настоящее время прослеживается острый кризис квалифицированных кадров в различных сферах рынка труда. Работодатели, хоть и обращают внимание на область предметных знаний, но все же отдают предпочтения универсальным навыкам. Подобная позиция связана с постоянно изменяющимися условиями труда, технологическим прогрессом. Подобные универсальные навыки (способность работать в команде, многозадачность, творческий подход, критическое мышление, продуктивное мышление, нацеленность на результат) вырабатываются в процессе профессионального становления.

Характерной чертой развития общества на протяжении последних десятилетий является его все более расширяющаяся информатизация. Отражением и следствием этой тенденции явилась потребность в подготовке подрастающего поколения к вступлению в информационное общество, любая профессиональная деятельность в котором будет связана с информатикой и информационными технологиями. Изучение данного курса имеет важное значение для развития мышления подростков. В современной психологии отмечается значительное влияние изучения информатики и использования компьютеров в обучении на развитие у школьников теоретического, творческого мышления, а также формирование нового типа мышления, так называемого операционного мышления, направленного на выбор оптимальных решений; открывает новые возможности для овладения такими современными методами научного познания, как формализация, моделирование, компьютерный эксперимент и т.д. Алгоритмическое мышление является необходимой частью научного взгляда на мир. В то же время оно включает и некоторые общие мыслительные навыки, полезные и в более широком контексте.

Направленность дополнительной образовательной программы — техническая. Она заключается в развитии технического и творческого мышления у обучающихся среднего школьного возраста через разработку алгоритмов в цифровой среде, взаимодействие с приложениями, сервисами и инструментами вне зависимости от платформы или интерфейса, формирование навыка определения различных видов ошибок (логическая, синтаксическая, ошибка среды и ошибка взаимодействия) и принципах их исправления в процессе разработки с помощью процедур отладки. Также развитие у детей умения составлять план деятельности, стремления к познавательным активностям, исследовательских, прикладных способностей, формирование навыков сотрудничества и коллективной работы в цифровой среде.

Развитие навыков программирования является уникальным способом формирования интереса учащихся к техническим направлениям деятельности, а также совершенствования алгоритмического, креативного мышления и навыков проектной деятельности.

Программа составлена с учетом задач национального проекта «Цифровая экономика» (федерального проекта «Кадры для цифровой экономики») и национального проекта «Образование» (федерального проекта «Молодые профессионалы»): создание условий для формирования у учащихся основ цифровых компетенций, необходимых для дальнейшего участия в разработке и поддержке инфраструктур передачи, обработки и хранения данных, а также цифровых продуктов для граждан, бизнеса и власти; разработка, внедрение цифровых технологий и платформенных решений.

Python – это язык программирования общего назначения, распространяемый с открытыми исходными текстами. Он оптимизирован для создания качественного программного обеспечения. Язык Python используется сотнями тысяч разработчиков по всему миру в таких областях, как создание веб-сценариев, системное программирование, создание пользовательских интерфейсов,

настройка программных продуктов под пользователя, численное программирование и в других. ...

Программа «Программирование на Python» представляет собой углубленный курс по программированию, дающий представление о понятиях структурного программирования (данных, операциях, переменных, ветвлениях в программе, циклах и функциях, массивах, файлах). Программа направлена на развитие логического и пространственного мышления обучающегося, способствует раскрытию творческого потенциала личности, приобретению практических умений и навыков в области компьютерных технологий, способствует интеллектуальному развитию ребенка.

Наш выбор Python для преподавания призван помочь обучающимся сделать первые шаги по одному из современных и перспективных путей развития IT-индустрии.

Курс служит средством внутривидеальной специализации в области новых информационных технологий, что способствует созданию дополнительных условий для проявления индивидуальных образовательных интересов учащихся, их дальнейшей профессиональной ориентации.

Данная программа относится к углубленному уровню, так как направлена на овладение знаниями и развитие навыков, она дает возможность ребенку погрузиться в атмосферу дополнительного образования, попробовать себя в новом лично значимом виде деятельности. Родителям дает возможность разобраться с логикой дополнительного образования и наметить линию индивидуального развития своего ребенка.

Программа базируется на решении кейсов и проектной деятельности. Для повышения качества образования данные кейсы основываются на реальных ситуациях и проблемах потенциальных работодателей. Это дает возможность ранней профессиональной ориентации обучающихся.

По форме организации содержания и процесса педагогической деятельности программа «Программирование на Python» интегрированная, так как объединяет в целое отдельные образовательные области на основе единства математики и информатики. Главной целью является приобретение обучающимся теоретических знаний и практических навыков и компетенций, достаточных для дальнейшего успешного обучения в системе непрерывного образования: школа – допобразование – ВУЗ – предприятие. Сущностью программы можно определить как развитие мышления и формирование мировоззрения в условиях преподавания интегрированного курса информатики, математики, программирования обучающихся среднего и старшего звена. Изучение всех этих предметов должно быть не целью, а средством познания мира, давать возможность обучающимся проникать в сущность изучаемых проблем. В теории интеграции в данном случае вкладывается понятие взаимосвязи, взаимопроникновения математики и информатики, что предполагает качественное изменение в параметрах нового объекта. В школьном курсе информатики вопросы программирования рассматриваются лишь в ознакомительном плане и на это выделяется недостаточное количество часов, как следствие – формальное восприятие обучающимися основ современного программирования. Образовательная программа «Программирование на Python» направлена на устранение данного пробела.

Новизна. Отличительной особенностью курса является его направленность на формирование у учащихся навыков поиска собственного решения поставленной практической задачи, представленную в виде адаптированного кейса из реальной жизни, составления алгоритма решения и его реализации с помощью средств программирования. Для этого, учащиеся проходят через следующие этапы: определяют и составляют последовательность действий, ведущую к цели, и записывают ее с помощью формального языка. Полученные знания учащиеся применяют при создании собственных проектов, которые защищаются перед другими учениками, педагогами и родителями.

Использование компьютерных технологий в работе с детьми среднего школьного возраста является стремительно развивающейся методикой в образовании во всем мире. С ее помощью можно более эффективно решать образовательные задачи, которые будут способствовать качественному улучшению обучения ребенка в школе.

Цель:

Способствовать формированию алгоритмического и критического мышления, навыков проектной деятельности через использование языка программирования Python как инструмента для реализации идей в области своих интересов.

Задачи:

- Познакомить с базовыми понятиями основ программирования.
- Познакомить с особенностями синтаксиса языка программирования Python.
- Познакомить с принципами объектно-ориентированного программирования.
- Сформировать навык правильного оформления кода.
- Сформировать навык использования итеративного подхода при решении различных задач.
- Формировать навык решения задач и разработки проектов с помощью Python.
- Формировать навык разработки игр с помощью языка программирования Python.
- Формировать навык тестирования и оптимизации созданных проектов.

Программа предназначена для учеников 5-11 классов, имеющих базовый уровень компьютерной грамотности.

Сроки реализации – 2022-2023 учебный год.

Формы и режим занятий. Занятия проводятся один раз в неделю, после уроков. 34 часа в год.

В рамках внеурочной деятельности предусматриваются следующие методы организации учебно-познавательной деятельности, позволяющие повысить эффективность обучения по курсу:

- практическое занятие;
- занятие – соревнование;
- консультация;
- самостоятельная работа.

Ожидаемые результаты и способы их проверки.

К концу обучения на начальном этапе будет обеспечена готовность обучающихся к продолжению образования, достигнут необходимый уровень их развития.

Учащийся научится:

- основам и принципам программирования;
- составлять основные алгоритмические конструкции;
- знание основ и овладение практическими базисными знаниями программирования в Python;
- знание основ и овладение практическими базисными навыками разработки игр в PyGame.

учащиеся должны знать: основы и принципы программирования на Python;

учащиеся должны уметь: решать академические и практические задачи по программированию, создавать программный код с нуля.

Основной формой подведения итогов дополнительной общеразвивающей программы «Язык программирования Python» является решение задач, проектная деятельность (разработка и отладка программ на языке программирования Python).

Формы аттестации: выполнение промежуточных групповых и индивидуальных проектов

Защита итогового проекта проходит в форме представления обучающимся индивидуального проекта по своему выбору, ответов на вопросы преподавателя. Обсуждения с учащимися достоинств и недостатков проекта.

Критерии оценивания итогового проекта: самостоятельность выполнения,

- законченность работы,
- соответствие выбранной тематике,
- оригинальность и качество решения
- проект уникален, и продемонстрировано творческое мышление участников

- проект хорошо продуман и имеет концепцию
- сложность
 - трудоемкость, многообразие используемых функций
- авторы продемонстрировали свою компетентность, сумели четко и ясно объяснить, как их проект работает.

Ожидаемые результаты и способы их проверки

Личностными результатами освоения выпускниками средней школы курса программирования являются:

- 1) бережное отношение к компьютерной технике как неотъемлемой части настоящего времени как основного помощника в быту;
- 2) потребность сохранять чистоту рабочего места и техники;
- 3) осознание применимости информационных технологий в народном хозяйстве и социально-экономической структуре;
- 4) осознание роли информационной технологии как главного атрибута XXI века;
- 5) сформированность мировоззрения, соответствующего современному уровню развития информационных технологий;
- 6) потребность саморазвития, в том числе логического мышления, понимание алгоритмов в информационных процессах;
- 7) готовность к самостоятельной творческой и ответственной деятельности;
- 8) готовность и способность вести диалог с другими людьми; сформированность навыков сотрудничества;
- 9) эстетическое отношение к языкам программирования, осознание их выразительных возможностей;
- 10) нравственное сознание и поведение на основе общечеловеческих ценностей.

Метапредметными результатами освоения выпускниками средней школы курса программирования являются:

- 1) умение эффективно общаться в процессе совместной деятельности со всеми её участниками, не допускать конфликтов;
- 2) владение навыками познавательной, учебно-исследовательской и проектной деятельности; использование различных методов познания; владение логическими операциями анализа, синтеза, сравнения;
- 3) способность к самостоятельному поиску информации, в том числе умение пользоваться справками программ и интернет поиском;
- 4) умение критически оценивать и интерпретировать информацию, получаемую из различных источников;
- 5) владение всеми видами компьютерной деятельности: машинописью, чтением и редактированием;
- 6) умение правильно построить алгоритм и создавать программы разных типов и применимости с учётом языков программирования и их особенностей (Turbo Pascal, Visual basic, Python и т.д.);
- 7) свободное владение письменной формой записи программ, циклом и структурой;
- 8) умение определять цели деятельности и планировать её, контролировать и корректировать деятельность;
- 9) умение оценивать свою и чужую работу с эстетических и нравственных позиций;
- 10) умение выбирать стратегию поведения, позволяющую достичь максимального эффекта.

Предметные результаты В результате изучения элективного курса «Программирование» обучающиеся научатся:

- определять результат выполнения алгоритма при заданных исходных данных; узнавать изученные алгоритмы обработки чисел и числовых последовательностей; создавать на их основе несложные программы анализа данных; читать и понимать несложные программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня;

- выполнять пошагово (с использованием компьютера или вручную) несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных;
- создавать на алгоритмическом языке программы для решения типовых задач базового уровня из различных предметных областей с использованием основных алгоритмических конструкций;
- использовать готовые прикладные компьютерные программы в соответствии с типом решаемых задач и по выбранной специализации;
- использовать компьютерно-математические модели для анализа соответствующих объектов и процессов, в том числе оценивать числовые параметры моделируемых объектов и процессов, а также интерпретировать результаты, получаемые в ходе моделирования реальных процессов; представлять результаты математического моделирования в наглядном виде, готовить полученные данные для публикации;
- правильно составлять текстовые документы в соответствии с эстетическими нормами и оптимальным количеством необходимого текста;
- работать с таблицами, обрабатывать большие массивы данных и проводить математические операции больших объемов;
- презентовать работу, используя соответствующие редакторы, не перегружать лишней информацией и правильно составлять структуру материала;
- разрабатывать программы, составляя этапы решения задач и проектирования их каркаса и подпрограмм;
- работе со всемирной сетью, настройкой связи и подключения, Выпускник получит возможность научиться:
 - понимать важность дискретизации данных; использовать знания о постановках задач поиска и сортировки; их роли при решении задач анализа данных;
 - использовать навыки и опыт разработки программ в выбранной среде программирования, включая тестирование и отладку программ; использовать основные управляющие конструкции последовательного программирования и библиотеки прикладных программ; выполнять созданные программы;
 - разрабатывать и использовать компьютерно-математические модели; оценивать числовые параметры моделируемых объектов и процессов; интерпретировать результаты, получаемые в ходе моделирования реальных процессов; анализировать готовые модели на предмет соответствия реальному объекту или процессу;
 - применять базы данных и справочные системы при решении задач, возникающих в ходе учебной деятельности и вне ее; создавать учебные многотабличные базы данных;
 - критически оценивать информацию, полученную из сети Интернет.

Учебно-тематический план

№ п/п	Наименование раздела, темы	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
1.	Раздел 1. Введение в Python	2	1	1	Тест по модулю
	1.1. Инструкции и структура программы.	1	1	0	Устный опрос
	1.2. Установка Python. Ввод и вывод данных.	1	0	1	Решение задач
2.	Раздел 2. Типы данных и операции	6	3	3	Тест по модулю
	2.1. Переменные. Операторы.	1	1	1	Тест по теме
	2.2. Типы данных.	4	2	2	Решение задач

	Преобразование типов. Различия типов данных. Базовые функции (abs, round, int, math)				
3.	Раздел 3. Инструкции и синтаксис	7	3	4	Тест по модулю
	3.1. Условный оператор if, Альтернативный условный оператор elif, else.	2	1	1	Решение задач
	3.2. Циклы while. Цикл for. Обработка включений. Функция random. Случайные числа.	5	2	3	Решение задач с использованием циклов
4.	Раздел 4. Функции и модули в программировании	8	3	5	Тест по модулю
	4.1. Встроенные и пользовательские функции. Создание функций. Функции, возвращающие результат.	3	1	2	Тест по теме
	4.2. Строки, последовательность символов. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки. Срезы строк. Сравнение строк.	5	2	3	Решение задач с использованием строк
5.	Раздел 5. Сложные типы данных	8	4	4	Тест по модулю
	5.1. Списки. Срезы списков. Решение задач со списками	2	1	1	Решение задач
	5.2. Матрицы	2	1	1	Решение задач
	5.3. Словари	2	1	1	Решение задач
	5.4. Множества в языке Python	2	1	1	Решение задач
	Раздел 6. Написание и отладка программ. Самостоятельная работа	3	0	3	Защита проекта
	Итого	34	14	20	

Содержание разделов программы

Тема 1 (1 час). Инструкции и структура программы..

Теория. Техника безопасности. История создания. Области применения и перспективы.

Инструкции и структура программы. Основы Алгоритмизации. Достоинства языка.

Тема 2 (1 час). Установка Python. Ввод и вывод данных.

Практика. Установка Python. Доступ к документации. Основы ввода и вывода данных. Первая программа на языке Python. Основы Алгоритмизации. Решение задач для развития логики и

Тема 3 (2 часа). Переменные. Операторы.

Теория. 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.

Практика. Решение задач на элементарные действия с числами. Создание программы простейший калькулятор.

Тема 4 (4 часа). Типы данных. Преобразование типов. Различия типов данных. Базовые функции (abs, round, int, math).

Теория. Типы данных. Преобразование типов. Различия типов данных. Базовые функции (abs, round,

Практика. Решение математических задач с использованием функции import math.

Тема 5 (2 часа). Условный оператор if, Альтернативный условный оператор elif, else.

Теория. 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

Практика. Практическое закрепление знаний по условным операторам. Создание программ.

Тема 6 (5 часов). Циклы while. Цикл for. Обработка включений. Функция random. Случайные числа.

Теория. 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.

Практика. Числа Фибоначчи. Решение задачи с циклом for. Создание игры угадай число. Повторение пройденного.

Тема 7 (3 часа). Встроенные и пользовательские функции. Создание функций. Функции, возвращающие результат.

Теория. 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.

Практика. Создание игры русская рулетка. Создание всех ранее созданных программ с использованием функций.

Тема 8 (5 часов). Строки, последовательность символов.

Теория. 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

Практика. Написание программ. Работа со строками.

Тема 9 (2 часа). Списки. Срезы списков. Решение задач со списками.

Теория. 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.

Практика. Написание программ. Работа со списками.

Тема 10 (2 часа). Матрицы.

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

Практика. Создание программ. Практическое закрепление полученных знаний.

Тема 11 (2 часа). Кортежи.

Теория. Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения.

Практика. Создание программ. Практическое закрепление полученных знаний.

о
с
т
е
й

Тема 12 (1 час). Словари.

Теория. Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.

Практика. Создание программ. Практическое закрепление полученных знаний.

Тема 13 (2 часа). Множества.

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

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

Оператор определения принадлежности элемента множеству.

Практика. Создание программ. Практическое закрепление полученных знаний.

Тема 14 (2 часа). Защита итогового проекта.

Теория.-

Практика. Выполнение и защита итоговой работы .

Календарно – тематическое планирование

№ п/п	Дата проведения	Тема занятия	Основные виды деятельности	Формы занятий	Мероприятия за рамками учебного плана
1	2.09	1.1. Инструкции и структура программы.	Установка Python. Доступ к документации. Основы ввода и вывода данных. Первая программа на языке Python. Основы Алгоритмизации. Решение задач для развития логики и понимания работы алгоритма.	Лекция Практическое занятие	
2	9.09	1.2. Установка Python. Ввод и вывод данных.	Установка Python. Доступ к документации. Основы ввода и вывода данных. Первая программа на языке Python. Основы Алгоритмизации. Решение задач для развития логики и понимания работы алгоритма.	Лекция Практическое занятие	

3 - 4	30 .0 9 7. 10	2.1. Пер еме нны е. Опе рато ры.	Создание программ. Практическое закрепление полученных знаний.	Пра кти чес кое зан ятие	
5 - 8	14 .1 0 21 .1 0 28 .1 0 11 .1 1	2.2 Тип ы данн ых. Пре обра зова ние типо в. Разл ичия типо в данн ых. Базо вые функ ции (abs, roun d, int, math)	Создание программ. Практическое закрепление полученных знаний.	Пра кти чес кое зан ятие	Уч аст ие в кон курс е
9 - 1 0	18 .1 1 25 .1 1	3.1. Усл овн ый опер атор if, Аль терн атив ный усло вны й опер атор	Создание программ. Практическое закрепление полученных знаний.	Ле кц ия Пра кти чес кое зан ятие	

		elif, else.			
1 1 - 1 5	2. 12 09 .1 2 16 .1 2 23 .1 2 30 .1 2	3.2. Цик лы while. Цик л for. Обр абот ка вкл юче ний. Фун кция rand om. Слу чайн ые числ а.	Создание программ. Практическое закрепление полученных знаний.		Ле кц ия Пр акт ич еск ое зан ят ие
1 6 - 1 8	13 .0 1 20 .0 1 27 .0 1	4.1. Встр оенн ые и поль зова тель ские фун кци и. Соз дани е фун кци й. Фун кци и, возв ращ ающ ие резу льта т.	Создание программ. Практическое закрепление полученных знаний.		Ле кц ия Пр акт ич еск ое зан ят ие

1 9 - 2 2	3. 02 10 .0 2 27 .0 2	4.2. Стр оки, посл едов ател ьнос ть сим воло в. Дос туп по инде ксу. Дли на стро ки и отри цате льн ые инде ксы. Пре обра зова ние типо в. При мен ение цик ла для обхо да стро ки. Срез ы стро к. Сра внен ие стро к.	Создание программ. Практическое закрепление полученных знаний.	Ле кц ия Пр акт ич еск ое зан ят ие	Пр оек т
-----------------------	--	---	--	---	----------------

2 3 - 2 4	24 .0 2 3. 03	5.1. Спи ски. Срез ы спис ков. Реш ение зада ч со спис кам и	Создание программ. Практическое закрепление полученных знаний.	Ле кц ия Пр акт ич еск ое зан ят ие	
2 5 - 2 6	10 .0 3 17 .0 3	5.2. Мат риц ы	Создание программ. Практическое закрепление полученных знаний.	Ле кц ия Пр акт ич еск ое зан ят ие	Уч аст ие в ко нк урс е
2 7 - 2 8	24 .0 3 14 .0 4	5.3. Сло вари	Создание программ. Практическое закрепление полученных знаний.	Пр акт ич еск ое зан ят ие	
2 9 - 3 0	21 .0 4 28 .0 4	5.4. Мно жест ва в язык е Pyt hon	Создание программ. Практическое закрепление полученных знаний.	Ле кц ия Пр акт ич еск ое зан ят ие	Со рев но ван ия
3 1 - 3 2	5. 05 12 .0 5	5.5. Кор теж и.	Создание программ. Практическое закрепление полученных знаний.	Пр акт ич еск ое зан ят ие	

3 3 - 3 4	19 .0 5 26 .0 5	Защ ита итог овог о прое кта	Защищают проект, участвуют в дискуссии	Ит ого вая ко нф ере нц ия
-----------------------	--------------------------------	--	--	---

Методическое обеспечение

Название темы	Формы проведения занятий	Методы и приемы	Оснащение	Форма подведения итогов
Введение в python	Учебное занятие Лекция Практическая работа	Репродуктивные: словесные (лекция); наглядные (демонстрация, иллюстрация); практические (работа с компьютерной программой); самостоятельная работа (работа по образцу, по алгоритму). Проблемно-развивающие методы: - создание проблемных ситуаций и разрешение их. Игровые методы	Ноутбуки с установленным программным обеспечением, проектор, экран, авторская презентация	Беседа
Типы данных и операции			Ноутбуки с установленным программным обеспечением, проектор, экран, авторская презентация	Опрос, презентация мини проекта
Инструкции и синтаксис.			Ноутбуки с установленным программным обеспечением, проектор, экран, авторская презентация	Презентация мини проекта
Функции и модули в программировании			Ноутбуки с установленным программным обеспечением, проектор, экран, авторская презентация	Презентация мини проекта
Сложные типы данных.			Ноутбуки с установленным программным обеспечением, проектор, экран, авторская презентация	Презентация мини проекта

Написание программ			Ноутбуки с установленным программным обеспечением, проектор, экран, авторская презентация	Презентация мини проекта
Отладка программ			Ноутбуки с установленным программным обеспечением, проектор, экран, авторская презентация	Презентация проекта
Защита итогового проекта				

Оценочные средства (контрольно-измерительные материалы)

№	Виды работ	Источники
1	Защита проектов	<p>Защита итогового проекта проходит в форме представления обучающимся индивидуального проекта по своему выбору, ответов на вопросы преподавателя. Обсуждения с учащимися достоинств и недостатков проекта.</p> <p>Критерии оценивания итогового проекта:</p> <ul style="list-style-type: none"> - самостоятельность выполнения, - законченность работы, - соответствие выбранной тематике, - оригинальность и качество решения - проект уникален, и продемонстрировано творческое мышление участников - проект хорошо продуман и имеет сюжет / концепцию - сложность - трудоемкость, многообразие используемых функций - авторы продемонстрировали свою компетентность, сумели четко и ясно объяснить, как их проект работает.

Список литературы для учителя.

Основная:

1. Клейнберг Дж. Алгоритмы: разработка и применение. СПб: Питер, 2016. - 800 с.
2. Златопольский Д. М. Основы программирования на языке Python. – М.: ДМК Пресс, 2017. – 284 с.: ил.
3. Бхаргава А. Грокаем алгоритмы. Иллюстрированное пособие для программистов и любопытствующих. - СПб.: Питер, 2017. - 288 с.

Дополнительная:

1. Луц М. Изучаем Python. СПб: Симво-плюс, 2011. - 1280 с.
2. Паронджанов В.Д. Учись писать, читать и понимать алгоритмы. М: Ямб, 2012. – 520 с.
4. Златопольский Д.М. Сборник задач по программированию. СПб: БХВ-Петербург, 2011. - 295 с.

5. ООП на Python: концепции, принципы и примеры реализации [Электронный ресурс]. – Режим доступа: <https://proglib.io/p/python-oop/>