

МУНИЦИПАЛЬНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
«УДУГУЧИНСКАЯ СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА»

РАССМОТРЕНО На заседании ШМО <u>Виталис</u> Протокол № 1 от 23.08.2023 г. Руководитель: Пасынкова О.Л.	СОГЛАСОВАНО Зам. директора по УВР МОУ «Удугучинская СОШ» <u>Н.М. Вострикова</u> «24» августа 2023 г.	Утверждено Директор МОУ «Удугучинская СОШ» <u>Е.В. Рыбакова</u> Приказ № 83 от 24.08.2023 г.
---	--	--

**Дополнительная общеобразовательная
общеразвивающая программа**
Программирование на Python

Направление техническое
Составитель Максимова Л.Ю. I
ФИО категория

Программа рассчитана на детей: 13-17 лет

Срок реализации программы 1 год

Количество часов: 1 час (в нед.) 34 часа (всего)

с. Удугучин 2023 г.

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1.1. Направленность программы

Настоящая программа имеет *техническую направленность*, рассчитана на учащихся 8-11 классов и предназначена для развития навыков алгоритмического мышления. Курс последовательно и целенаправленно вовлекает учащегося в процесс самостоятельного и осмысленного составления законченных алгоритмов и программ, вырабатывает необходимые составляющие алгоритмической и программистской грамотности.

1.2. Актуальность программы

Язык программирования Python на сегодняшний день является наиболее популярной высокоуровневой средой международного олимпиадного программирования. Python –мощный и простой для изучения язык программирования. В нем представлены проработанные высокоуровневые структуры данных и простой, но эффективный подход к объектно-ориентированному программированию.

Python – серьёзный язык программирования. В то же время учащиеся в полной мере могут раскрыть свои творческие таланты, так как с его помощью можно легко создавать игры и другие приложения.

Активизация познавательного процесса позволяет учащимся более полно выразить свой творческий потенциал и реализовывать собственные идеи в изучаемой области знаний, создаёт предпосылки по применению информационных компетенций в других учебных курсах, а также способствует возникновению дальней мотивации, направленной на освоение профессий, связанных с разработкой программного обеспечения.

1.3. Отличительные особенности программы

1. Проектный подход. В процессе обучения происходит воспитание культуры проектной деятельности, раскрываются и осваиваются основные шаги по разработке и созданию проекта.

2. Межпредметность. В курсе прослеживается тесная взаимосвязь с математикой, физикой, географией, русским языком, музыкой и другими предметами школьного цикла. Знания, полученные на других предметах, логичным образом могут быть использованы при разработке проектов.

3. Вариативность. Учащиеся с достаточной степенью свободы и самостоятельности могут выбирать темы проектов.

4. Коммуникация. В курсе предусмотрена работа в командах, парах, использование возможностей сетевого сообщества для взаимодействия. Обязательное условие - публичная презентация и защита проектов.

1.4. Нормативно-правовые основания

- Законом РФ «Об образовании в Российской Федерации» (от 29 декабря 2012 года № 273-ФЗ);
- Приказом Минпросвещения Российской Федерации от 27.07.2022 № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»
- Постановлением Главного государственного санитарного врача РФ от 28.09.2020 г. №28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- Письмом Минобрнауки РФ от 14.12.2015г. №09-3564 «О внеурочной деятельности и реализации дополнительных общеобразовательных программ» (вместе с «Методическими рекомендациями по организации внеурочной деятельности и реализации дополнительных общеобразовательных программ»);

Адресат программы

Программа рассчитана на обучающихся 15 – 17 лет.

1.5. Объем программы

Срок реализации программы – 1 год (34 часов).

1.6. Организация образовательного процесса

Режим занятий: 1 занятие в неделю, общее количество часов в год – 34.

Продолжительность занятия – 40 минут.

Распределение учебного времени занятия определяется в соответствии с возрастом детей и требованиями СанПиН.

При организации образовательного процесса используются фронтальные, групповые и индивидуальные формы работы.

2. ЦЕЛЬ И ЗАДАЧИ ПРОГРАММЫ

Цель: способствовать формированию творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python..

Задачи:

Обучающие:

- обучение основам алгоритмизации и программирования, приобщении к проектно-творческой деятельности;
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- объяснить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных;
- научить применять функции при написании программ на языке программирования Python;
- научить отлаживать и тестировать программы, делать выводы о работе этих программ.

Воспитательные:

- воспитывать интерес к занятиям информатикой;
- воспитывать культуру общения между учащимися;
- воспитывать культуру безопасного труда при работе за компьютером;
- воспитывать культуру работы в глобальной сети;
- воспитание целеустремленности и результативности в процессе решении учебных задач;

Развивающие:

- развивать познавательный интерес школьников;
- развивать творческое воображение, математическое и образное мышление учащихся;
- развивать умение работать с компьютерными программами и дополнительными источниками информации;
- развивать навыки планирования проекта, умение работать в группе

3. СОДЕРЖАНИЕ ПРОГРАММЫ

3.1. Учебно-тематическое планирование

№ п/п	Название разделов, тем	Количество часов			Формы и методы организации занятий	Формы аттестации (контроля)
		Всего	Теор	Практ		
1.	Введение. Инструктаж по ТБ. Знакомство с правилами работы за компьютером.	1	0,5	0,5	Инструктаж Демонстрация /показ/ Работа за компьютером	Опрос
2.	Основы языка программирования Python	18	9	9		
2.1.	Введение, типы данных, синтаксис	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.2.	Арифметические операции	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.3.	Логика. Логические операторы	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.4.	Условный оператор. Сложные условия	2	1	1	Объяснение нового материала Демонстрация /показ/	Тестирование Решение практических

					Практическая работа	задач
2.5.	Цикл for	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.6.	Цикл while	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.7.	Строковый тип данных	2	1	1	Объяснение нового Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.8.	Списки и индексация в них	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.9.	Функции, лямбды	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
3.	Алгоритмы	4	2	2		
3.1	Арифметические алгоритмы	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических

						задач
3.2	Сортировки. Понятие асимптотической сложности	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
4.	Структуры данных	6	2	4		
4.1.	Многомерные массивы	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
4.2.	Кортежи, множества, словари	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
4.3.	Работа со словарями Стек, очередь, дек	2	1	1	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
5.	Разработка и защита проекта (кейса)	5	1	4		
5.1.	Алгоритм создания проекта. Подбор материалов для проекта	2	1	1	Объяснение нового материала	План проекта

					Демонстрация /показ	
5.2.	Создание программы с использованием подготовленных материалов. Тестирование и отладка проекта.	2		2	Демонстрация /показ/ Практическая работа	Проект
5.3.	Защита проекта.	1		1	Демонстрация и защита собственного проекта Публичное выступление	Творческий проект
	Итого	34	14,5	19,5		

3.2. Содержание учебно-тематического плана

1. Введение. Инструктаж по ТБ. Знакомство с правилами работы за компьютером. (1 час)

Теория: Знакомство с программой творческого объединения. Правила поведения на занятии. Техника безопасности при работе на компьютере.

Практика: Практическая работа за компьютером.

2. Основы языка программирования Python (18 часов)

2.1. Введение, типы данных, синтаксис.

Теория: О языке Python. Установка Python и сред разработки. Синтаксис языка Python. Переменные в Python. Типы данных и приоритеты операций в языке Python. Преобразование типов в Python. Ввод-вывод в Python. Где применяется Python.

Практика: Решение задач.

2.2. Арифметические операции.

Теория: Целочисленная арифметика. Целочисленный тип (int) в Python.

Практика: Решение задач.

2.3. Логика. Логические операторы.

Теория: Условная инструкция в Python. Ветвление в программе. Логический тип (bool).

Практика: Решение задач.

2.4. Условный оператор. Сложные условия.

Теория: Условный оператор: задачи на шахматной доске. Построение сложных условий. Разные задачи на условный оператор.

Практика: Решение задач.

2.5. Цикл for.

Теория: Цикл for в Python. Понятие об итераторе. Функция range(). Решение задач с циклом for.

Практика: Решение задач.

2.6. Цикл while.

Теория: Цикл while в Python. Инструкция управления циклом в Python. Отличия от цикла for. Применимость в каждом конкретном случае.

Практика: Решение задач.

2.7. Строковый тип данных.

Теория: Строки в Python. Срезы строк в Python. Методы строк в Python. Задачи на обработку строк.

Практика: Решение задач.

2.8. Списки и индексация в них.

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.

Практика: Решение задач.

2.9. Функции, лямбды.

Теория: Функции в Python. Локальные и глобальные переменные. Парадигма структурного программирования. Математические функции в Python. Лямбда-функции в Python.

Практика: Решение задач.

3. Алгоритмы (4 часов)

3.1. Арифметические алгоритмы

Теория: Что такое алгоритм. Различные формы записи алгоритмов (нумерованный список, таблица, блок-схема). Примеры линейных алгоритмов, алгоритмов с ветвлениями и повторениями (в повседневной жизни, в литературных произведениях, на уроках математики. Решето Эратосфена. Факторизация. Проверка на простоту.

Практика: Решение задач.

3.2. Сортировки. Понятие асимптотической сложности.

Теория: Асимптотическая сложность алгоритма. Сортировка подсчётом. Поразрядная сортировка. Быстрая сортировка Хоара. Сортировка слиянием.

Практика: Решение задач.

4. Структуры данных (6 часов)

4.1. Многомерные массивы

Теория: Множества в Python. Двумерные и многомерные списки в Python.

Практика: Решение задач.

4.2. Кортежи, множества, словари

Теория: Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения. Множества в языке Python. Множества. Множественный тип данных. Описание множеств. Операции, допустимые над множествами: объединение, пересечение, разность, включение. Оператор определения принадлежности элемента множеству. Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.

Практика: Решение задач.

4.3. Работа со словарями

Теория: Генераторы списков.

Практика: Решение задач.

4.4. Стек, очередь, дек.

Теория: Стек, очередь, дек.

Практика: Решение задач.

5. Разработка и защита проекта (кейса). (5 часов)

5.1. Алгоритм создания проекта.

Теория: жизненный цикл проекта.

Практика: разработка плана проекта.

5.2. Подбор материалов для проекта.

Теория: Поиск информации в Интернете.

Практика: Поиск информации по теме проекта.

5.3. Создание программы с использованием подготовленных материалов.

Практика: Создание программного кода. Разработка собственного проекта, его программирование, дизайн, оформление.

5.4. Тестирование и отладка проекта.

Практика: Тестирование и отладка проекта.

5.5. Подготовка к защите проекта.

Практика: подготовка публичного выступления

5.6. Защита проекта.

Практика: Публичное выступление. Защита проекта.

4. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДООП

При освоении программы отслеживаются три вида результатов: предметный, метапредметный и личностный, что позволяет определить динамическую картину творческого развития обучающихся.

Личностные

- сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
- потребность в самореализации в творческой деятельности, желании учиться;
- умение осуществлять совместную информационную деятельность, в частности, при выполнении учебных проектов;

- осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов; отношение к профессиональной деятельности как возможности участия в решении личных, общественных, государственных, общенациональных проблем.

Метапредметные

- умение самостоятельно выделять и формулировать познавательные цели;
- умение применять методы моделирования;
- умение прогнозировать результат деятельности и его характеристики, вносить необходимые коррективы в план по ходу его выполнения;
- приобретение навыков самостоятельного создания способов решения проблем творческого и поискового характера;
- умение использовать средства ИКТ в решении когнитивных, коммуникативных и организационных задач с соблюдением требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности;
- готовность и способность к самостоятельной информационно-познавательной деятельности, включая умение ориентироваться в различных источниках информации, критически оценивать и интерпретировать информацию, получаемую из различных источников;
- умение продуктивно общаться и взаимодействовать в процессе совместной деятельности, учитывать позиции других участников деятельности, эффективно разрешать конфликты.

Предметные

- умение интерпретировать сообщения с позиции их смысла, синтаксиса, ценности;
- умение планировать действия, необходимые для достижения заданной цели (связанной с преобразованием информации) с помощью фиксированного набора средств, способность реализовать эти действия с использованием языка программирования, оценивать результаты работы;
- приобретение опыта планирования учебного сотрудничества с учителем и сверстниками: определение цели, функции участников, способов взаимодействия;
- овладение понятием сложности алгоритма, знание основных алгоритмов обработки числовой и текстовой информации, алгоритмов поиска и сортировки;

- овладение навыками алгоритмического мышления и пониманием необходимости формального описания алгоритмов;
- умение реализовывать алгоритмы в виде программ и программных систем; владение элементарными навыками документирования программ.
- умение сопоставлять математические модели задач и их компьютерные аналоги, анализировать полученные результаты с точки зрения соответствия объекту и целям моделирования;
- умение анализировать разные способы записи алгоритмов с позиции того, что они являются информационными моделями;
- владение универсальным языком программирования высокого уровня, представлениями о базовых типах данных и структурах данных; умением использовать основные управляющие конструкции языка; умением производить отладку программ с помощью встроенного отладчика.

II. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

1. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК /с 01.09.2023г. по 25.05.2024г./

Месяц	Сроки изучения учебного материала/нед	№ группы/год обучения/			Аттестация обучающихся	Каникулярный период
		Количество часов				
		№ 1/1		-		
Сентябрь	1 неделя	1				
	2 неделя	1				
	3 неделя	1				
	4 неделя	1				
	5 неделя	1				
Октябрь	1 неделя	1				
	2 неделя	1				
	3 неделя	1				
	4 неделя	Самостоятельная работа				
Ноябрь	1 неделя	1				

	2 неделя	1				
	3 неделя	1				
	4 неделя	1				
Декабрь	1 неделя	1				
	2 неделя	1				
	3 неделя	1				
	4 неделя	1				
	5 неделя	1				
Январь	1 неделя	Самостоятельная работа				
	2 неделя	1				
	3 неделя	1				
	4 неделя	1				
Февраль	1 неделя	1				
	2 неделя	1				
	3 неделя	1				
	4 неделя	1				
Март	1 неделя	1				
	2 неделя	1				
	3 неделя	Самостоятельная работа				
	4 неделя	1				
Апрель	1 неделя	1				
	2 неделя	1				
	3 неделя	1				
	4 неделя	1				
Май	1 неделя	1				
	2 неделя	1				
	3 неделя	1			Защита проектов	
	4 неделя					

Всего учебных недель	34					
Всего часов по программе		34	34			

2. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

2.1. Материально-техническое обеспечение

Для эффективности образовательного процесса необходимы:

Техническое оборудование:

Компьютер

Периферийные устройства: Сканер, принтер, проектор.

Локальная сеть, выход в Интернет.

Операционная система: Windows

Среда программирования PUNTON.

2.2. Кадровое обеспечение

Программа реализуется педагогом дополнительного образования, имеющим профессиональное образование по направленности программы.

3. ФОРМЫ АТТЕСТАЦИИ/КОНТРОЛЯ

Система мониторинга достижения детьми планируемых результатов освоения дополнительной общеобразовательной общеразвивающей программы обеспечивает комплексный подход к оценке текущих и промежуточных результатов программы и позволяет осуществить оценку динамики достижения обучающихся, а также включает описание объекта, форм, периодичность и содержание мониторинга. Отслеживание результатов осуществляется через входной, текущий, промежуточный и итоговый контроль.

Время проведения	Цель проведения	Формы контроля
Входной контроль		
В начале учебного года	Определение уровня развития детей, их знания ИКТ.	беседа
Текущий контроль		
В течение учебного года	Определение степени усвоения обучающимися учебного материала. Определение готовности детей к восприятию нового материала. Выявление детей, отстающих и опережающих обучение. Подбор наиболее эффективных методов и средств обучения.	Тестирование Решение практических задач Участие в конкурсах, олимпиадах
Промежуточный контроль		
В конце месяца, полугодия	Определение степени усвоения обучающимися учебного материала. Определение результатов обучения.	Практические и творческие задания
Итоговый контроль		
В конце учебного года /или курса обучения/	Определение изменения уровня развития детей, их творческих способностей. Определение результатов	Защита проекта

	обучения.	
--	-----------	--

4. КАЛЕНДАРНЫЙ ПЛАН ВОСПИТАТЕЛЬНЫХ МЕРОПРИЯТИЙ

Название мероприятия (тема)	Срок (месяц)
Проведение классных часов и родительских собраний	Сентябрь 2023
Участие в конкурсах, олимпиадах	Февраль, март 2024
Защита проектов	Май 2024

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Система оценки результатов освоения программы состоит из текущего контроля успеваемости и промежуточной и итоговой аттестации учащихся. Текущий контроль учащихся проводится с целью установления фактического уровня теоретических знаний и практических умений и навыков по темам (разделам) дополнительной общеразвивающей программы. Текущий контроль успеваемости учащихся осуществляется педагогом по каждой изученной теме. Достигнутые учащимися умения и навыки заносятся в диагностическую карту. Текущий контроль может проводиться в следующих формах: практическая (лабораторная) работа, самостоятельные работы репродуктивного характера; вопросники, тестирование, проект.

Промежуточная аттестация учащихся проводится с целью повышения ответственности педагогов и учащихся за результаты образовательного процесса, за объективную оценку усвоения учащимися дополнительных общеразвивающих программ каждого года обучения; за степень усвоения учащимися дополнительной общеразвивающей программы в рамках учебного года. Промежуточная аттестация проводится как оценка результатов обучения за определённый промежуток учебного времени – полугодие, год. Промежуточная аттестация учащихся осуществляется администрацией Учреждения. Промежуточная аттестация учащихся включает в себя проверку теоретических знаний и практических умений и навыков. Промежуточная аттестация учащихся может проводиться в следующих формах: самостоятельные работы репродуктивного характера; срезовые работы; вопросники, тестирование, практическая (лабораторная) работа.

Итоговая аттестация учащихся проводится с целью выявления уровня развития способностей и личностных качеств ребенка и их соответствия прогнозируемым результатам освоения дополнительной общеразвивающей

программы. Итоговая аттестация учащихся проводится по окончании обучения по дополнительной общеразвивающей программе. Итоговая аттестация учащихся включает в себя проверку теоретических знаний и практических умений и навыков. Итоговая аттестация учащихся проводится следующих формах: проектная работа, практическое занятие.

Критерии оценки уровня теоретической подготовки:

- высокий уровень – учащийся освоил практически весь объём знаний 100-80%, предусмотренных программой за конкретный период; специальные термины употребляет осознанно и в полном соответствии с их содержанием;
- средний уровень – у учащегося объём усвоенных знаний составляет 70-50%; сочетает специальную терминологию с бытовой;
- низкий уровень – учащийся овладел менее чем 50% объёма знаний, предусмотренных программой; ребёнок, как правило, избегает употреблять специальные термины;
- программу не освоил - учащийся овладел менее чем 20% объёма знаний, предусмотренных программой.

Критерии оценки уровня практической подготовки:

- высокий уровень – учащийся овладел на 100-80% умениями и навыками, предусмотренными программой за конкретный период; работает с оборудованием самостоятельно, не испытывает особых трудностей; выполняет практические задания с элементами творчества;
- средний уровень – у учащегося объём усвоенных умений и навыков составляет 70-50%; работает с оборудованием с помощью педагога; в основном, выполняет задания на основе образца;
- низкий уровень - ребёнок овладел менее чем 50% предусмотренных умений и навыков, испытывает серьёзные затруднения при работе с оборудованием; в состоянии выполнять лишь простейшие практические задания педагога;
- программу не освоил - учащийся овладел менее чем 20% предусмотренных программой объёма умений и навыков.

6. МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

В обучении школьников наиболее приемлемы комбинированные занятия, предусматривающие смену методов обучения и деятельности обучаемых, позволяющие свести работу за компьютером к регламентированной норме. С учетом данных о распределении усвоения информации и кризисах внимания учащихся на уроке, рекомендуется проводить объяснения в первой части занятия, а на конец – планировать деятельность, которая наиболее интересна для учащихся и имеет для них большее личностное значение. В комбинированном занятии можно выделить следующие основные этапы:

1. организационный момент;
2. активизация мышления и актуализация ранее изученного (разминка, короткие задания на развитие внимания, сообразительности, памяти, фронтальный опрос по ранее изученному материалу);
3. объяснение нового материала, сопровождаемое, как правило, компьютерной презентацией; на этом этапе

учитель четко и доступно объясняет материал, по возможности используя традиционные и электронные наглядные пособия; учитель в процессе беседы вводит новые понятия, организует совместный поиск и анализ примеров, при необходимости переходящий в игру или в дискуссию;

4. работа за компьютером (выполнение работ компьютерного практикума, решение задач);
 5. подведение итогов занятия.

На занятиях очень важно, чтобы каждый ученик имел доступ к компьютеру и пытался выполнять практические работы, пытался выполнять практические задания по описанию, обращаясь за помощью к учителю или товарищам.

Чтобы выполнение заданий компьютерного практикума шло успешно, школьников всякий раз следует подготовить к новому для них виду деятельности, подробно объяснив, что каждое задание выполняется в заданной последовательности и в строгом соответствии с описанием, поэтому нужно очень внимательно читать каждое указание (каждый пункт), выполнять его, и только после этого переходить к следующему указанию (пункту). Нужно чтобы ученик очень четко осознавал, что он делает и какая именно операция у него не получается. Очень важно, чтобы учитель не подсказывал готовые решения, а, выявив истинную причину возникшего у ученика затруднения, направлял его к правильному решению.

Задания творческого характера представляют собой информационные мини-задачи. Выполнение творческого задания требует от ученика значительной самостоятельности при уточнении его условий, по представлению необходимой информации, по выбору технологических средств и приемов его выполнения

Критерии результативности реализации программы

Уровень освоения программы	Критерии	Показатели
Высокий уровень	Учащиеся должны знать правила техники безопасности при работе, грамотно излагать программный материал, знать	Выполнение 100% - 70% заданий

	основные блоки команд, уметь самостоятельно применять изученные способы, аргументировать свою позицию, оценивать ситуацию и полученный результат.	
Средний уровень	Учащиеся должны знать основные блоки команд, уметь самостоятельно или во взаимодействии с педагогом выполнять задания, уметь высказывать мнение, обобщать, классифицировать, обсуждать, грамотно и по существу излагать программный материал, не допуская существенных неточностей в ответе.	Выполнение от 50% до 70% заданий
Низкий уровень	учащиеся не знают значительной части материала, допускают существенные ошибки, с большими затруднениями выполняют практические задания.	Выполнение менее 50% заданий

6. СПИСОК ЛИТЕРАТУРЫ

1. Поляков К.Ю. Программирование. Python. C++. Учебное пособие – М.: БИНОМ. Лаборатория знаний, 2019
- 2 Самоучитель Python. Выпуск 2.0. Дмитрий Мусин. – Май, 2017
- 3 Программирование для детей на языке Python — Москва: Издательство АСТ — 2017
4. Учим Python, делая крутые игры/ Эл Свейгарт [пер. с англ. М.А.Райтмана]. – Москва: Эксмо, 2018

□

ЭЛЕКТРОННЫЕ ОБРАЗОВАТЕЛЬНЫЕ РЕСУРСЫ

1. <https://foxford.ru/> – Фоксфорд — онлайн-школа для учеников 1–11 классов, учителей и родителей.
2. <https://pythonworld.ru/> - Python для начинающих.
3. <https://younglinux.info/python.php> - Python. Введение в программирование.
- 4.