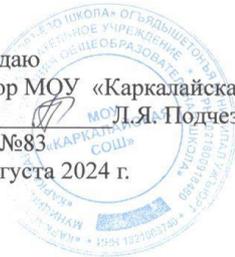


Муниципальное общеобразовательное учреждение  
"Каркалайская средняя общеобразовательная школа"

Рассмотрена на заседании школьного  
методического объединения классных  
руководителей  
Протокол № 1  
От 23 августа 2024 г.  
Руководитель ШМО   
Погудина Л.Н.

Принята на заседании  
Педагогического совета  
Протокол № 1  
От 26 августа 2024 г.

Утверждаю  
Директор МОУ «Каркалайская  
СОШ»  Л.Я. Подчерцева  
Приказ №83  
от 26 августа 2024 г.



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА  
«ПРОГРАММА-ПЛЮС»

Возраст учащихся: 11-16 лет  
Срок реализации программы: 1 год  
Составитель: Туйматов Н.И.,  
учитель МОУ «Каркалайская СОШ»

с.Каркалай, 2024 год

## 1. Пояснительная записка

Настоящая программа разработана в соответствии с нормативными документами и не противоречит законодательству Российской Федерации.

По форме организации образовательного процесса программа является модульной: 1-ый модуль - «Программирование в Scratch», 2-ой модуль - «Программирование на языке Python».

Программа основывается на использовании среды программирования Scratch и Python, которая понятна детям и соответствует современным направлениям в программировании. Обучающиеся набираются в группу по результатам анкетирования, направленного на выявление круга интересов обучающихся и их родителей (законных представителей) (Приложение №1).

Используются разнообразные педагогические технологии: ИКТ (информационно-компьютерные технологии), здоровьесберегающие технологии, ТРКМ (технологии развития критического мышления). Реализация программы предусматривает использование системно-деятельностного подхода.

Программа будет интересна детям от 11 до 16 лет, увлекающимся научно-технической направленностью, развивать свои творческие способности и эстетический вкус. Освоившие программу легко смогут на языке программирования Scratch создать интерактивный продукт: анимацию, презентацию и т.п., а на языке программирования Python - программы и применять их на практике.

Актуальность программы обусловлена тем, что мы живем в век информатизации общества, информационные технологии проникают в нашу жизнь с разных сторон. Одно из самых удивительных и увлекательных занятий настоящего времени - программирование. Обучение основам программирования школьников должно осуществляться на специальном языке программирования, который будет понятен детям, будет легок для освоения и соответствовать современным направлениям в программировании. В данном курсе предполагается вести изучение программирования в игровой, увлекательной форме. Одной из главных концепций языка Scratch, является развитие собственных задумок с первой идеи до конечного программного продукта.

Scratch – это объектно-ориентированная среда, в которой блоки программ собираются из разноцветных кирпичиков. Scratch – это мультимедийная система. Большая часть операторов языка направлена на работу с графикой и звуком, создание анимационных и видеоэффектов. Манипуляции с медиаинформацией – главная цель создания Scratch.

Мультимедийная среда Scratch позволяет сформировать у детей стойкий интерес к программированию, отвечает всем современным требованиям объектно-ориентированного программирования.

Изучение языка значительно облегчает последующий переход к изучению других языков программирования. Преимуществом Scratch, среди подобных сред программирования, также является наличие версий для различных операционных систем: для Windows, Mac OS, GNU/Linux. В настоящее время имеет смысл рассматривать программы с открытым кодом, что позволяет сформировать у учащихся более широкое представление о возможностях работы с цифровой техникой.

Аспект новизны заключается в том, что Scratch не просто язык программирования, а еще и интерактивная среда, где результаты действий визуализированы, что делает работу с программой понятной, интересной и увлекательной.

Особенность среды Scratch, позволяющая создавать в программе мультфильмы, анимацию и даже простейшие игры, делает образовательную программу «Программирование в Scratch» практически значимой для современного школьника, т.к. дает возможность увидеть практическое назначение алгоритмов и программ, что будет способствовать развитию интереса к профессиям, связанным с программированием.

Выбор современного языка программирования Python обусловлен достоинствами языка:

1. Кроссплатформенность и бесплатность.
2. Простой синтаксис и богатые возможности.
3. Расширенная стандартная библиотека, возможность, в будущем, разработки промышленных приложений.
4. Язык Python поддерживается на всех олимпиадах для школьников всероссийского уровня.

С 2015 года во всех вариантах КИМ ЕГЭ по информатике примеры программ записываются, в том числе и на языке Python. В зависимости от поставленных задач на занятиях используются различные методы обучения. Все задания расположены по возрастанию степени их сложности. Доступная детям практическая деятельность помогает избежать умственного переутомления.

Педагогическая целесообразность данной образовательной программы состоит в том, что изучая программирование в среде Scratch у обучающихся формируется не только логическое мышление, но и навыки работы с мультимедиа; создаются условия для активного, поискового учения, предоставляются широкие возможности для разнообразного моделирования, а при изучении программирования на языке Python формируются навыки работы по алгоритму, а также составления самого алгоритма работы программы.

Формы организации деятельности обучающихся: групповые, индивидуальные.

Методы обучения

- словесные методы (лекция, объяснение);
- демонстративно-наглядные (демонстрация работы в программе, схем, скриптов, таблиц);
- проектные методы (разработка проекта по спирали творчества, моделирование, планирование деятельности)
- работа с Интернет - сообществом (публикация проектов в Интернет - сообществе скретчеров).
- объяснительно – иллюстративные (методы обучения, при использовании которых, дети воспринимают и усваивают готовую информацию);
- частично-поисковые методы обучения (участие детей в коллективном поиске, решение поставленной задачи совместно с педагогом). исследовательские методы обучения (овладение детьми методами научного познания, самостоятельной творческой работы).

Группы формируются из разновозрастных детей, имеющих различные базовые знания. Большое внимание уделяется индивидуальной работе с каждым ребенком и используется личностно-ориентированный подход.

Объем программы составляет 102 часа: 68 часов на изучение первого модуля, 34 часа на изучение второго модуля (3 занятия в неделю по 45 минут).

Программа рассчитана на учащихся 11-16 лет.

## **2.Цели и задачи программы**

### **Первый модуль**

**Цель программы** — познакомить школьников с элементами информатики и на примере языка (и среды) программирования Scratch— с элементами программирования; сформировать базовые представления о языках программирования, алгоритме, исполнителе, способах записи алгоритма; организовать проектную научно-познавательную деятельность творческого характера; сформировать у школьника познавательный интерес к учёбе и исследовательские навыки.

### **Второй модуль**

**Целью** изучения модуля «Программирование на языке Python» является получение теоретических и практических знаний, умений и навыков в области современной информатики; формирование целостного мировоззрения, соответствующего современному уровню развития науки и общественной практики, учитывающего социальное, культурное, языковое, духовное многообразие современного мира.

**Основная задача курса** – способствовать формированию у школьников информационной и функциональной компетентности, развитию алгоритмического мышления.

### **Задачи курса:**

- дать обучающимся представление о современном подходе к изучению реального мира, о широком использовании алгоритмов и вычислительной техники в научных исследованиях;
- сформировать у обучающихся умения владеть компьютером как средством решения практических задач;

- подготовка учеников к активной полноценной жизни и работе в условиях технологически развитого общества;
- создание условий для внедрения новых информационных технологий в учебно-воспитательный процесс школы;
- реализовать в наиболее полной мере возрастающий интерес обучающихся к углубленному изучению программирования через совершенствование их алгоритмического и логического мышления;
- формирование знаний о роли информационных процессов в живой природе, технике, обществе;
- формирование знаний о значении информатики и вычислительной техники в развитии общества и в изменении характера труда человека;
- формирование знаний об основных принципах работы компьютера, способах передачи информации;
- формирование знаний об основных этапах информационной технологии решения задач в широком смысле;
- формирование умений моделирования и применения его в разных предметных областях;
- формирование умений и навыков самостоятельного использования компьютера в качестве средства для решения практических задач.

Реализация этих задач будет способствовать дальнейшему формированию взгляда школьников на мир, раскрытию роли информатики в формировании естественнонаучной картины мира, развитию мышления, в том числе формированию алгоритмического стиля мышления, подготовке учеников к жизни в информационном обществе.

### 3.Содержание программы

#### Учебно-тематический план

#### **Первый модуль**

№ занятия	Название раздела, темы	Количество часов			Формы организации занятий	Формы аттестации (контроля)
		Всего	Теория	Практика		
<b>Раздел 1. Введение. Знакомство со средой Scratch</b>						
1.	Знакомство, цели и задачи, ТБ и правила поведения.	1	1		Беседа	
2.	Основы работы с ПК	1	0,5	0,5	Беседа, практическое занятие	
3.	Знакомство со средой Scratch. Интерфейс	1	0,5	0,5	практическое занятие	
4.	Знакомство со средой Scratch. Интерфейс	1	0,5	0,5	практическое занятие	
5.	Основные инструменты встроенного графического редактора программной среды Scratch	1	0,5	0,5	практическое занятие	
6.	Что такое алгоритм	1	1		Беседа	
<b>Раздел 2. Управление спрайтом</b>						
7.	Спрайт. Управление спрайтом	1	0,5	0,5	практическое занятие	
8.	Спрайт. Управление спрайтом	1		1	практическое занятие	
9.	Управление несколькими спрайтами	1	0,5	0,5	Самостоятельная работа	
10.	Управление спрайтом в лабиринте	1	0,5	0,5	Самостоятельная работа	
11.	Слой. Работа со слоями	1	0,5	0,5	практическое занятие	

12.	Взаимодействие объектов	1	0,5	0,5	практическое занятие	
13.	Создание игры «Лабиринт»	1		1	Создание проекта	Промежуточная аттестация
<b>Раздел 3. Основы анимации в Scratch</b>						
14.	Изучение работы блоков категории «Движение»	1	0,5	0,5	практическое занятие	
15.	Изучение работы блоков категории «Внешность»	1	0,5	0,5	практическое занятие	
16.	Смена фона	1	0,5	0,5	практическое занятие	
17.	Звук	1	0,5	0,5	практическое занятие	
18.	Диалоги	1	0,5	0,5	практическое занятие	
19.	Создание анимационного проекта	1	0,5	0,5	создание проектов	
20.	Создание анимационного проекта	1		1	создание проектов, тест	Промежуточная аттестация
<b>Раздел 4. Алгоритмы в Scratch</b>						
21.	Линейные алгоритмы	1	0,5	0,5	практическое занятие	
22.	Разветвляющиеся алгоритмы	1	0,5	0,5	практическое занятие	
23.	Циклические алгоритмы	1	0,5	0,5	практическое занятие	
24.	Ящик «Управление»	1	0,5	0,5	практическое занятие	
25.	Создание проекта «Балерина»	1		1	создание проектов	Промежуточная аттестация
<b>Раздел 5. Циклы</b>						
26.	Цикл «повторять всегда»	1	0,5	0,5	практическое занятие	
27.	Цикл «повторять всегда»	1		1	создание проектов	
28.	Цикл «повторить n раз»	1	0,5	0,5	практическое занятие	
29.	Цикл «повторить n раз»	1		1	создание проектов	
30.	Цикл «повторять пока не ...»	1	0,5	0,5	практическое занятие	
31.	Цикл «повторять пока не ...»	1		1	создание проектов	
32.	Анимационный проект, с применением циклов	1		1	создание проектов	Промежуточная аттестация
<b>Раздел 6. Создание анимационных проектов</b>						
33.	Проект «Аквариум»	1		1	Создание проекта	
34.	Проект «Природа»	1		1	Создание проекта	
35.	Проект «Здоровый образ жизни»	1		1	Создание проекта	
36.	Проект «Игра»	1		1	Создание проекта	
37.	Проект «Правила	1		1	Создание	

	дорожного движения»				проекта	
38.	Анимированная открытка	1		1	Практическая работа	
39.	Проект «Случайные числа»	1		1	Создание проекта	
<b>Раздел 7. Переменные</b>						
40.	Переменные в Scratch	1	0,5	0,5	Самостоятельная работа	
41.	Команды ящика «Данные»	1	0,5	0,5	Практическая работа	
42.	Переменные и рычажки	1	0,5	0,5	Практическая работа	
43.	Создание игры с подсчетом очков	1		1	Создание проекта	Промежуточная аттестация
<b>Раздел 8. Списки</b>						
44.	Создание списков	1	0,5	0,5	Практическая работа	
45.	Добавление, удаление и редактирование элементов	1	0,5	0,5	Практическая работа	
46.	Добавление, удаление и редактирование элементов	1	0,5	0,5	Самостоятельная работа	
47.	Проект «Съедобное и несъедобное»	1		1	Создание проекта	Промежуточная аттестация
<b>Раздел 9. Музыка</b>						
48.	Ноты и паузы в Скретче	1	0,5	0,5	Практическая работа	
49.	Метроном Скретча	1	0,5	0,5	Самостоятельная работа	
50.	Мелодические инструменты. Барабаны	1	0,5	0,5	Практическая работа	
51.	Проект «Музыка»	1		1	Создание проекта	
<b>Раздел 10. Свободное проектирование</b>						
52.	Определение целей и задач проекта	1	0,5	0,5	беседа	
53.	Составление плана работы над проектом	1	0,5	0,5	беседа	
54.	Подготовка материалов для создания проекта	1	0,5	0,5	Самостоятельная работа	
55.	Подготовка материалов для создания проекта	1		1	Самостоятельная работа	
56.	Работа над проектом	1		1	Создание проекта	
57.	Работа над проектом	1		1	Создание проекта	
58.	Работа над проектом	1		1	Создание проекта	
59.	Работа над проектом	1		1	Создание проекта	
60.	Работа над проектом	1		1	Создание проекта	
61.	Работа над проектом	1		1	Создание проекта	
62.	Работа над проектом	1		1	Создание проекта	
63.	Работа над проектом	1		1	Создание проекта	

64.	Работа над проектом	1		1	Создание проекта	
65.	Работа над проектом	1		1	Создание проекта	
66.	Защита проектов	1		1	Создание проекта	Итоговая аттестация
67.	Защита проектов	1		1	Создание проекта	Итоговая аттестация
68.	Итоговое занятие	1		1	беседа	
Итого часов:		68	19	49		

**Раздел 1. Введение. Знакомство со средой Scratch (6 часов)**

*Тема 1. Знакомство, цели и задачи, ТБ и правила поведения (1ч.)*

Теория. Справка о содержании курса. Техника безопасности в кабинете. Правила работы с ПК.

*Тема 2. Основы работы с ПК (1ч.).*

Теория. Устройство компьютера. Органы управления. Приемы работы с устройствами ввода и вывода информации.

Практика. Работа с ПК.

*Тема 3. Знакомство со средой Scratch. Интерфейс (2ч.)*

Теория. Знакомство со средой программирования Scratch. Интерфейс Scratch. Главное меню Scratch. Коллекции спрайтов и фонов. Использование интернета для импорта объектов. Практика. Установка программы Scratch. Изучение интерфейса среды.

*Тема 4. Основные инструменты встроенного графического редактора программной среды Scratch (1ч.)*

Теория. Особенности графического редактора среды Scratch.

Практика. Поиск, импорт и редактирование спрайтов из интернета. Сохранение и открытие проектов.

*Тема 5. Что такое алгоритм (1ч.)*

Теория. Понятие алгоритма, способы записи алгоритма. Составление алгоритмов

**Раздел 2. Управление спрайтом (7 часов)**

*Тема 1. Спрайт. Управление спрайтом (2ч.)*

Теория. Понятие спрайта и объекта. Команды и блоки. Программные единицы: процедуры и скрипты. Навигация в среде Scratch. Система координат. Оси координат. Команды управления. Градусная мера угла.

Практика. Создание и редактирование спрайтов и фонов для сцены, создание новых спрайтов и сцен.

*Тема 2. Управление несколькими спрайтами (1ч.)*

Теория. Команды и алгоритмы управления спрайтами.

Практика. Создание и редактирование спрайтов и фонов для сцены, создание новых спрайтов и сцен.

*Тема 3. Управление спрайтом в лабиринте (1ч.)*

Теория. Процедуры и скрипты, как программные единицы управления. Управление спрайтом.

Практика. Проект «Лабиринт»

*Тема 4. Слои. Работа со слоями (1ч.)*

Теория. Средства рисования группы перо. Знакомство с правильными геометрическими фигурами.

Практика. Определение координат спрайта. Ориентация по координатам.

*Тема 5. Взаимодействие объектов (1ч)*

Теория. Способы взаимодействия спрайтов.

Практика. Работа с файлом «Кошка догоняет мышку»

*Тема 6. Создание игры «Лабиринт» (1ч.)*

Практика. Создание интерактивного продукта «Игра «Лабиринт»

**Раздел 3. Основы анимации в Scratch (7 часов).**

*Тема 1. Изучение работы блоков категории «Движение» (1ч.)*

Теория. Блок «Движение». Команды: шаг, поворот, переход, плыть, направление.

Практика. Применение возможностей блока «Движение» в управление спрайтом

*Тема 2. Изучение работы блоков категории «Внешность» (1ч.)*

Теория. Блок «Внешность». Коды команд: сказать, думать, показать, изменить, размер  
Практика. Применение возможностей блока «Внешность» в управление объектом анимации  
*Тема 3. Смена фона (1ч.)*

Теория. Фон. Смена фона. Алгоритм перехода объекта от одного фона к другому.  
Практика. Выбор фона. Создание фона. Работа с объектами

*Тема 4. Звук (1ч.)*

Теория. Знакомство с блоком «Звук». Запись звука

Практика. Работа со звуком

*Тема 5. Диалоги (1ч.)*

Теория. Организация диалога между спрайтами

Практика. Создание продукта, включающего в себя диалог между объектами сцены

*Тема 6. Создание анимационного проекта (2ч.)*

Практика. Создание проекта с применением мультимедийных возможностей Scratch.

**Раздел 4. Алгоритмы в Scratch (5 часов)**

*Тема 1. Линейные алгоритмы (1ч.)*

Теория. Понятие линейного алгоритма. Работа с линейным алгоритмом. Сенсоры.

Практика. Применение линейного алгоритма в управлении спрайтом

*Тема 2. Разветвляющиеся алгоритмы (1ч.)*

Теория. Понятие разветвляющегося алгоритма. Выбор видов ветвления. Выполнение скриптов с ветвлением.

Практика. Применение ветвления в среде визуального программирования Scratch.

*Тема 3. Циклические алгоритмы (1ч.)*

Теория. Циклы. Способы использования.

Практика. Работа с безусловным циклом, циклами со счетчиком, с предусловием и постусловием.

*Тема 4. Ящик «Управление» (1ч.)*

Теория. Содержимое ящика «Управление».

Практика. Использование команд управления для анимации объектов

*Тема 5. Создание проекта «Балерина» (1ч.)*

Практика. Проект «Балерина»

**Раздел 5. Циклы (7 часов)**

*Тема 1. Цикл «повторять всегда» (2ч.)*

Теория. Команда повторить. Конструкция всегда.

Практика. Работа с циклом «повторять всегда».

*Тема 2. Цикл «повторить n раз» (2ч.)*

Теория. Цикл «повторить n раз». Способы задания цикла.

Практика. Рисование узоров и орнаментов

*Тема 3. Цикл «повторять пока не ...» (2ч.)*

Теория. Цикл «повторять пока не ...». Истинное условие.

Практика. Работа с циклом «повторять пока не ...».

*Тема 4. Анимационный проект, с применением циклов (1ч.)*

Практика. Создание проекта «Генерирование чисел»

**Раздел 6. Создание анимационных проектов (7 часов)**

*Темы 1-7. Проект «Аквариум». Проект «Природа». Проект «Здоровый образ жизни». Проект «Игра». Проект «Правила дорожного движения». Анимированная открытка. Проект «Случайные числа»*

Практика. Разработка и создание сюжетов, сцен по заданным темам в среде программирования Скретч.

**Раздел 7. Переменные (4 часа)**

*Тема 1. Переменные в Scratch (1ч.)*

Теория. Что такое переменные в Скретч. Применение переменных.

Практика. Работа с блоком переменных

*Тема 2. Команды ящика «Данные» (1ч.)*

Теория. Ящик «Данные». Команды. Таймер.

Практика. Применение в сюжете команд ящика «Данные»

*Тема 3. Переменные и рычажки (1ч.)*

Теория. Счет очков. Ввод переменных с помощью рычажка.

Практика. Применение ввода переменных с помощью рычажка.

Тема 4. Создание игры с подсчетом очков (1ч.)

Практика. Создание в среде Scratch игры подсчитывающей набранные очки.

**Раздел 8.** Списки (4 часа)

Тема 1. Создание списков (1ч.)

Теория. Создание списков. Блок «Данные».

Практика. Создание списка в среде Скретч

Тема 2. Добавление, удаление и редактирование элементов (2ч.)

Теория. Заполнение и редактирование списков. Ручное заполнение. Чтение из файла. Подсчет элементов списка.

Практика. Работа со списками при создании продукта

Тема 3. Проект «Съедобное и несъедобное» (1ч.)

Практика. Создание проекта «Съедобное и несъедобное»

**Раздел 9.** Музыка (4 часа)

Тема 1. Ноты и паузы в Скретче (1ч.)

Теория. Музыкальная система Scratch.

Практика. Работа с нотами на языке программирования

Тема 2. Метроном Скретча (1ч.)

Теория. Длина звучания ноты. Темп. Блоки управления темпом.

Практика. Применение возможностей Скретча в работе с нотами

Тема 3. Мелодические инструменты. Барабаны (1ч.)

Теория. Группы мелодических инструментов. Барабаны.

Практика. Изучение мелодических инструментов и барабанов. Применение в проектах.

Тема 4. Проект «Музыка» (1ч.)

Практика. Создание собственного музыкального продукта.

**Раздел 10.** Свободное проектирование (17 часов)

Тема 1. Определение целей и задач проекта (1ч.)

Теория. Как определить цель. Какие поставить задачи.

Практика. Определение цели и постановка задач творческого проекта

Тема 2. Составление плана работы над проектом (1ч.)

Теория. Как составить план работы. Корректировка плана.

Практика. Составление плана работы над проектом

Тема 3. Подготовка материалов для создания проекта (2ч.)

Теория. Для чего нужна подготовка к проекту. Как собрать материал.

Практика. Подготовка материалов для создания проекта

Тема 4. Работа над проектом (10ч.)

Практика. Создание, проверка, тестирование проекта в среде Scratch

Тема 5. Защита проектов (2ч.)

Практика. Защита проектов

Тема 6. Итоговое занятие (1ч.)

### **Второй модуль**

№ занятия	Название раздела, темы	Количество часов			Формы организации занятий	Формы аттестации (контроля)
		Всего	Теория	Практика		
<b>Раздел 1. Введение в Python</b>						
1	Язык Python. Среда программирования.	1	1		Беседа	
2	Типы данных, операции. Оператор присваивания	1	0,5	0,5	Беседа, практическое занятие	
3	Типы данных, операции. Оператор присваивания	1		1	Практическое занятие	
4	Числа. Стандартные операции	1	1		Беседа	
5	Числа. Стандартные операции	1		1	Практическое занятие	

<b>Раздел 2. Алгоритмические конструкции</b>						
6	Условный оператор.	1	1		Беседа	
7	Условный оператор.	1		1	Практическое занятие	
8	Цикл while. Теория чисел	1	1		Беседа	
9	Цикл while. Теория чисел	1	0,5	0,5	Практическое занятие	
10	Цикл for	1	1		Беседа	
11	Цикл for	1		1	Практическое занятие	
<b>Раздел 3. Строки</b>						
12	Строки. Литералы строк	1	0,5	0,5	Беседа	
13	Строки. Литералы строк	1		1	Практическое занятие	
14	Строки. Срезы строк.	1	0,5	0,5	Беседа, практическое занятие	
15	Строки. Методы строк	1	0,5	0,5	Беседа, практическое занятие	Промежуточная аттестация
<b>Раздел 4. Функции</b>						
16	Функции. Парадигма и преимущества структурного программирования.	1	0,5	0,5	Беседа, практическое занятие	
17	Граф и стек вызовов функций. Области видимости переменных	1	0,5	0,5	Беседа, практическое занятие	
18	Прямая рекурсия	1	0,5	0,5	Беседа, практическое занятие	
19	Косвенная рекурсия	1	0,5	0,5	Беседа, практическое занятие	
<b>Раздел 5. Списки и кортежи</b>						
20	Списки и кортежи в Python. Сходства и различия	1	0,5	0,5	Беседа, практическое занятие	
21	Операции со списками	1	0,5	0,5	Беседа, практическое занятие	
22	Операции со списками	1		1	практическое занятие	
23	Срезы списков	1	0,5	0,5	Беседа, практическое занятие	
24	Матрицы. Операции над матрицами	1	0,5	0,5	Беседа, практическое занятие	
25	Матрицы. Операции над матрицами	1		1	практическое занятие	
<b>Раздел 6. Словари и множества</b>						
26	Словари	1	0,5	0,5	Беседа, практическое занятие	
27	Множества	1	0,5	0,5	Беседа, практическое	

					занятие	
28	Множества	1	0,5	0,5	Беседа, практическое занятие	
<b>Раздел 7. Объектно-ориентированное программирование</b>						
29	Классы в Python	1	0,5	0,5	Беседа, практическое занятие	
30	Классы в Python	1	0,5	0,5	Беседа, практическое занятие	
31	Классы в Python	1		1	Практическая работа	
<b>Раздел 8. Проектная деятельность</b>						
32	Подготовка проекта	1	0,5	0,5	Создание проекта	
33	Создание проекта	1		1	Создание проекта	
34	Защита проекта	1		1	Создание проекта	Итоговая аттестация
Итого часов:		34	14,5	19,5		

### **Раздел 1. Введение в Python (5 часов)**

#### *Тема 1. Язык Python. Среда программирования. (1ч.)*

Теория. Справка о содержании курса. Техника безопасности в кабинете. Правила работы с ПК. Язык программирования Python. Достоинства и недостатки.

#### *Тема 2. Типы данных, операции. Оператор присваивания (2ч.).*

Теория. Устройство компьютера. Органы управления. Ввод и вывод числовой информации.

Практика. Создание программы.

#### *Тема 3. Числа. Стандартные операции (2ч.)*

Теория. Стандартные операции с целыми и действительными числами.

Практика. Создание программы.

### **Раздел 2. Алгоритмические конструкции (6 часов)**

#### *Тема 1. Условный оператор (2ч.)*

Теория. Условная и циклическая инструкции. Каскадность и вложенность алгоритмических инструкций.

Практика. Создание программы с условным оператором

#### *Тема 2. Цикл while. Теория чисел (2ч.)*

Теория. Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума.

Практика. Создание программы с циклом

#### *Тема 3. Цикл for (2ч.)*

Теория. Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума.

Практика. Создание программы с циклом

### **Раздел 3. Строки (4 часа).**

#### *Тема 1. Строки. Литералы строк (2ч.)*

Теория. Ввод-вывод строки. Решение задач на ввод строки, поиск подстроки.

Практика. Создание и редактирование программ.

#### *Тема 2. Строки. Срезы строк. (1ч.)*

Теория. Преобразование строки. Решение упражнений

Практика. Написание программы

#### *Тема 3. Строки. Методы строк (1ч.)*

Теория. Применение методов строки.

Практика. Решение задач

### **Раздел 4. Функции (4 часов)**

*Тема 1. Функции. Парадигма и преимущества структурного программирования. (1ч.)*  
Теория. Обзор парадигм программирования. Особенности применения языков программирования.

Практика. Решение задач

*Тема 2. Граф и стек вызовов функций. Области видимости переменных (1ч.)*

Теория. Применение стек и граф вызовов функций.

Практика. Применение стек и граф вызовов функций в написании программ.

*Тема 3. Прямая рекурсия (1ч.)*

Теория. Понятие прямой рекурсии и ее применение

Практика. Создание программы с прямой рекурсией

*Тема 4. Косвенная рекурсия (1ч.)*

Теория. Понятие косвенной рекурсии и ее применение

Практика. Написание программы с косвенной рекурсией

**Раздел 5. Списки и кортежи (6 часов)**

*Тема 1. Списки и кортежи в Python. Сходства и различия (1ч.)*

Теория. Представление списка и кортежа в памяти компьютера, сходства и различия.

Практика. Работа со списками и кортежами

*Тема 2. Операции со списками (2ч.)*

Теория. Способы заполнения списка (с клавиатуры, из файла, случайным образом, по формуле)

Практика. Заполнение списка различными способами.

*Тема 3. Срезы списков (1ч.)*

Теория. Методы работы со списком и кортежем. Методы сортировки списка.

Практика. Работа со списком и кортежем, сортировка списка.

*Тема 4. Матрицы. Операции над матрицами (2ч.)*

Теория. Вычисление сложности алгоритма. Многомерные списки.

Практика. Использование матриц в написании программ

**Раздел 6. Словари и множества (3 часов)**

*Тема 1. Словари (1ч.)*

Теория. Понятие словаря. Способы создания. Словарь, преимущества и недостатки, методы работы со словарем. Словари со смешанными значениями. Кодирование и декодирование текста.

Практика. Создание словарей, работа со словарями, кодирование и декодирование текста.

*Тема 2. Множества (2ч.)*

Теория. Понятие множества. Создание множеств. Множество, преимущества и недостатки, методы работы с множеством.

Практика. Создание множеств

**Раздел 7. Объектно-ориентированное программирование (3 часа)**

*Тема 1. Классы в Python (3ч.)*

Теория. Понятия «класс», «метод INIT», «экземпляр», «наследование», «полиморфизм», «исключения», «виджет», «интерфейс», «интерфейс», «событие».

Практика. Работа с классами в Python.

**Раздел 8. Проектная деятельность (3 часа)**

*Тема 1. Подготовка проекта (1ч.)*

Теория. Как подготовить проект? С чего начать проект?

Практика. Подготовка к проекту

*Тема 2. Создание проекта (1ч.)*

Практика. Написание программы на выбранную тему (проекта)

*Тема 3. Защита проекта (1ч.)*

Практика. Защита проекта

#### **4. Планируемые результаты**

В результате обучения **по первому модулю** ученики будут уметь:

- самостоятельно планировать пути достижения целей, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;
- соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата;

- оценивать правильность выполнения учебной задачи, собственные возможности ее решения;
- создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;
- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
- умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий.
- формирование ответственного отношения к учению;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, творческой и других видов деятельности.
- на формирование *умений* формализации и структурирования информации, умения выбирать способ представления данных в соответствии с поставленной задачей с использованием соответствующих программных средств обработки данных;
- формирование навыков и умений безопасного и целесообразного поведения при работе с компьютерными программами и в Интернете, умения соблюдать нормы информационной этики и права.
- умение следовать требованиям техники безопасности, гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий;
- умение пользоваться персональным компьютером и его периферийным оборудованием;
- умение искать информацию с применением правил поиска (построения запросов), в компьютерных сетях, некомпьютерных источниках информации (справочниках и словарях, каталогах, библиотеках) при выполнении заданий и проектов по различным учебным дисциплинам;
- умение составлять сценарии проектов среды Scratch;
- умение составлять алгоритмы, определять последовательность выполнения команд; использовать обширную библиотеку готовых сцен и исполнителей;
- умение создавать линейные алгоритмы для исполнителя; умение создавать циклические и ветвящиеся алгоритмы;
- умение управлять одновременной работой нескольких исполнителей; умение передавать сообщения между исполнителями;
- умение тестировать и оптимизировать алгоритмы исполнителей;
- умение создавать и редактировать документы в текстовом процессоре;
- умение работать с блоками текста: выделять, копировать, удалять; использовать необходимые шрифты; форматировать документ;
- умение размещать документы в облачном хранилище; организовывать коллективную работу с документами; настраивать права доступа к документам;
- умение выбирать способ представления своего проекта с использованием соответствующих программных средств.

Для оценки обученности школьников при реализации программы «Программирование в Scratch» применяются следующие уровни:

- 1 уровень – репродуктивный,
- 2 уровень – самостоятельная работа с помощью педагога,
- 3 уровень – самостоятельная работа без помощи педагога,

- 4 уровень – творческий (самостоятельное моделирование и изготовление)

Конечным результатом выполнения программы предполагается выход обучающихся на 3 и 4 уровни обученности, участие в выставках и конкурсах различных уровней.

Контроль теоретических знаний осуществляется в форме опроса, уровень практических умений определяется через выполнение практических заданий, участие в конкурсах.

В результате обучения по второму модулю ученики будут уметь:

- самостоятельно планировать пути достижения целей, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;
- соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата;
- оценивать правильность выполнения учебной задачи, собственные возможности ее решения;
- создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;
- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
- умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий.
- формирование ответственного отношения к учению;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, творческой и других видов деятельности.
- на формирование *умений* формализации и структурирования информации, умения выбирать способ представления данных в соответствии с поставленной задачей с использованием соответствующих программных средств обработки данных;
- формирование навыков и умений безопасного и целесообразного поведения при работе с компьютерными программами и в Интернете, умения соблюдать нормы информационной этики и права.
- умение следовать требованиям техники безопасности, гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий;
- умение пользоваться персональным компьютером и его периферийным оборудованием;
- умение искать информацию с применением правил поиска (построения запросов), в компьютерных сетях, некомпьютерных источниках информации (справочниках и словарях, каталогах, библиотеках) при выполнении заданий и проектов по различным учебным дисциплинам;
- умение составлять сценарии проектов на языке программирования Python;
- умение составлять алгоритмы, определять последовательность выполнения команд;
- умение создавать линейные алгоритмы для исполнителя; умение создавать циклические и ветвящиеся алгоритмы;
- умение тестировать и оптимизировать алгоритмы исполнителей;
- умение выбирать способ представления своего проекта с использованием соответствующих программных средств.

Для оценки обученности школьников при реализации программы «Программирование на языке Python» применяются следующие уровни:

- 1 уровень – репродуктивный,
- 2 уровень – самостоятельная работа с помощью педагога,
- 3 уровень – самостоятельная работа без помощи педагога,
- 4 уровень – творческий (самостоятельное моделирование и изготовление)

Конечным результатом выполнения программы предполагается выход обучающихся на 3 и 4 уровни обученности, участие в выставках и конкурсах различных уровней.

Контроль теоретических знаний осуществляется в форме опроса, уровень практических умений определяется через выполнение практических заданий, участие в конкурсах.

### 5.Календарный учебный график

Учебный год: 02.09.2024 г - 26.05.2025 г.

#### Первый модуль

Месяцы	№ недели	Даты	Кол-во часов/из них на контроль	Аттестация уч-ся
<b>1-ый модуль</b>				
Сентябрь	1	02-06	3	
	2	09-13	3	
	3	16-20	3	
	4	23-27	3	
Октябрь	5	30.09-04.10	3	
	6	07-11	3	
	7	14-18	3	
	8	21-25	3	
Ноябрь	9	05-08	3	
	10	11-15	3	
	11	18-22	3	
	12	25-29	3	
Декабрь	13	02-06	3	
	14	09-13	3	
	15	16-20	3	
	16	23-27	3	
Январь	17	30.12-10.01	3	
	18	13-17	3	
	19	20-24	3	
	20	27-31	3	
Февраль	21	03-07	3	
	22	10-14	3	
	23	17-21	2 (1-ый модуль)/1(2-ой модуль)	
<b>2-ой модуль</b>				
	24	24-28	3/1	Промежуточная аттестация
Март	25	03-07	3	
	26	10-14	3	
	27	17-21	3	
Апрель	28	31.03-04.04	3	
	29	07-11	3	
	30	14-18	3	
	31	21-25	3	
	32	28-30	1	
Май	33	05-08	2	
	34	12-16	3/1	Итоговый контроль
	35	19-23	3/1	Итоговый контроль
Всего учебных недель	35			

Всего часов по программе	102		102/3	
--------------------------	-----	--	-------	--

### 6. Условия реализации программы

Для успешной реализации программы необходим учебный кабинет с хорошим освещением, компьютерной техникой.

Техническое оснащение:

1. Персональный компьютер
2. Проектор
3. Бесплатно распространяемые программы «Scratch 3.0», «Python»

### 7. Формы аттестации/контроля

Контроль уровня достижения планируемых результатов осуществляется через промежуточную и итоговую аттестацию. Промежуточная и итоговая аттестации проходят в форме презентаций проектов, выставок, конкурсов.

### 8. Оценочные материалы

Критерии оценивания проектных работ

0 баллов – несоответствие критерию

1 балл – неполное соответствие критерию

2 балла – полное соответствие критерию

№	Критерий	Балл
1	Соответствие заявленной теме	
2	Оригинальность	
3	Творческий подход	
4	Эстетический вид, оформление работы	
5	Общее впечатление от работы	
6	Сложность выполнения	

### Первый модуль

#### Тест по разделам «Управление спрайтом. Основы анимации в Scratch»

1. Как называется подвижный графический объект, который действует на сцене проекта и выполняет разнообразные алгоритмы (сценарии). Исполнитель алгоритмов, которому доступны все команды языка Scratch.

- А) Скрипт    Б) Спрайт    В) Сцена    Г) Котенок

2. Блоки команд в программе Scratch разделены на разноцветные категории. Сколько таких категорий?

- А) 20    Б) 15    В) 10    Г) 7

3. Как называется алгоритм (или сценарий), составленный из блоков языка Scratch для какого-нибудь объекта?

- А) Скрипт    Б) Спрайт    В) Сцена    Г) Код

4. Чему равна ширина сцены?

- А) 320 точек    Б) 480 точек    В) 260 точек    Г) Может меняться

5. Сколько костюмов может иметь спрайт?

- А) 1    Б) 2    В) Любое количество    Г) Можно не более 7

6. Чему равна высота сцены?

- А) 320 точек    Б) 480 точек    В) 360 точек    Г) Может меняться

7. Как называется место, где спрайты двигаются, рисуют и взаимодействуют?

- А) Скрипт    Б) Спрайт    В) Сцена    Г) Котенок

8. Можно ли сделать проект, в котором нет сцены?

- А) Да    Б) Нет    В) Иногда можно

9. Какое расширение имеют файлы, созданные в среде Scratch?

- А) .sb2    Б) .exe    В) .psd    Г) .bmp

10. Набор команд, которые может выполнять объект, называют ...

- А) СКИ    Б) Алгоритм    В) Скрипт    Г) Программа

Ответы на тест:

1. Б    2. В    3. А    4. Б    5. В    6. В    7. В    8. Б    9. А    10. А

#### Тест по разделам «Алгоритмы в Scratch. Циклы»

1. Назовите три основных типа алгоритмов.

Ответ: \_\_\_\_\_.

2. Форма блоков, которые служат для создания циклов и ветвлений.  
 А) стек Б) заголовок В) заглушка Г) логический Д) С-блок
3. Понятная компьютеру система символов для точной записи алгоритмов и их выполнения компьютером.  
 А) интерфейс Б) язык программирования В) скрипт Г) блок-схема Д) сценарий
4. Алгоритм, выраженный на языке программирования.  
 А) интерфейс Б) язык программирования В) скрипт Г) блок-схема Д) сценарий
5. Нарисуйте блок-схему линейного алгоритма.

Ответы на тест:

1. Линейный, разветвляющийся, циклический
2. Д
3. В
4. В

### Итоговый тест

1. Какого ящика команд не существует  
 А) картинка Б) внешний вид В) движение
2. Описание последовательности действий приводящее к конечному результату  
 А) команда Б) исполнение В) алгоритм
3. Поле в котором происходит действие программы  
 А) рабочий стол Б) сцена В) холст
4. Что такое скрипт?  
 А) звук в программе Б) звук двери В) отдельное действие спрайта
5. Можно ли нарисовать спрайт/сцену самому?  
 А) да Б) нет
6. Как называется подвижный графический объект, который действует на сцене проекта и выполняет разнообразные сценарии. Исполнитель алгоритмов, которому доступны все команды языка Scratch?  
 А) скрипт Б) спрайт В) кот
7. Блоки команд в программе Scratch разделены на разноцветные категории. Сколько таких категорий?  
 А) 15 Б) 10 В) 8
8. Можно ли создавать игры с помощью данной программы?  
 А) да Б) нет

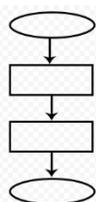
Ответы на тест:

1. А 2. В 3. Б 4. В 5. А 6. Б 7. Б 8. А

### Второй модуль

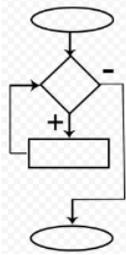
#### Промежуточная аттестация

1. Язык программирования Python подходит для разработки:
  - a. Компьютерных и мобильных приложений
  - b. Аналитика и машинное обучение
  - c. Игр
  - d. Ничего из этого.
2. Назовите тип алгоритма:



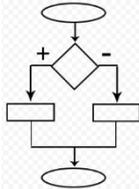
- a. Разветвляющийся
- b. Линейный
- c. Циклический
- d. Смешанный

3. Назовите тип алгоритма:



- a. Линейный
- b. Разветвляющийся
- c. Циклический
- d. Смешанный

4. Назовите тип алгоритма:



- a. Разветвляющийся
- b. Линейный
- c. Смешанный
- d. Циклический

5. Что хранит в себе переменная?

- a. Имя
- b. Значение
- c. Тип
- d. Длину своего значения

6. Что обозначает тип данных int?

- a. Целочисленное
- b. Вещественное
- c. Строковое
- d. Булево

7. Выберите правильную запись оператора присваивания:

- a.  $10 = x$
- b.  $y = 7,8$
- c.  $a = 5$
- d.  $a == b + x$

8. Укажите оператор ввода:

- a. input()
- b. print()
- c. int()
- d. random()

9. Сколько возможных значений у переменной типа bool?

- a. 2
- b. 4
- c. 10
- d. Сколько угодно

10. Какой оператор здесь используется?

If  $n < 100$ :

$b = n + a$

- a. Условный оператор
- b. Оператор присваивания
- c. Оператор сложения
- d. Оператор умножения

11. Что лучше использовать для множественного ветвления?

- a. if – elif – else
- b. Много if
- c. if – else – elif

- d. while
12. Оператор цикла в языке Python:
- while
  - for
  - if
  - print

13. Сколько раз произойдет итерация цикла? (Итерация – единичное выполнение тела цикла)

```
print("Осталось", total)
```

```
total = 100
i = 0
while i < 0:
    n = int(input())
    total = total - n
    i = i + 1
```

14. Для чего нужен оператор break?
- Для завершения программы
  - Для выхода из цикла
  - Для поломки компьютера
  - Для удаления программы

Ключи:

- |          |           |         |
|----------|-----------|---------|
| 1. a,b,c | 6. a      | 11. a   |
| 2. b     | 7. c      | 12. a,b |
| 3. c     | 8. a      | 13. b   |
| 4. a     | 9. a      | 14. b   |
| 5. b     | 10. a,b,c |         |

### Итоговая аттестация

- Где находятся параметры, а где аргументы функции?
  - Параметры пишутся при объявлении функции, аргументы при вызове
  - Аргументы пишутся при объявлении функции, параметры при вызове
  - Это одно и то же!
  - У функции есть только параметры
- Что делает функция len()?
  - Возвращает длину строки
  - Возвращает случайное число
  - Возвращает номер символа
  - Возвращает модуль числа
- Как добавить модуль в программу?
  - import math
  - import math()
  - import (math)
  - import.math
- На каких операционных системах может работать Python?
  - Windows
  - Linux
  - macOS
  - Ничего из этого
- От чего язык программирования называется «Питон»?
  - В честь змеи
  - В честь ТВ-шоу
  - В честь игры

- d. В честь блюда
6. Сколько уже лет языку программирования Python?
- a. 30
  - b. 19
  - c. 7
  - d. 15

7. Создатель языка программирования Python
- a. Гвидо Ван Россум
  - b. Дэвид Паттерсон
  - c. Эрвин Дональд Кнут
  - d. Джеймс Артур Гослинг

8. `a = 345`. Что выведет команда `print(//100)`
- a. 3
  - b. 5
  - c. 4
  - d. 34

9. Выберите циклический алгоритм

a. `k = 0`  
`while k < 10:`  
`print("Привет")`  
`k += 1`

b. `a = int(input())`  
`b = int(input())`  
`c = int(input())`  
`s = a+b+c`

`print(c)`

c. `a = int(input())`  
`if a > 0:`  
`print(a)`

`else:`  
`print(a)`

10. В какой строке правильно записан ввод числа с клавиатуры?

- 1. `a = int(input())`
- 2. `b=input(int())`
- 3. `c=int(input())`
- 4. `s=a+b+c`
- 5. `print(s)`

- a. 1
- b. 2
- c. 3
- d. 4
- e. 5

11. В какой строке допущена ошибка?

```

1. a = int(input())
2. if a>0:
3.     print(a):
4. else
5.     print(A)

```

- a. 1
- b. 2
- c. 5
- d. 4

12. Что такое «else»?

- a. Так как
- b. Иначе
- c. Если
- d. Потому что

13. Сколько раз программа напишет слово «Пока»?

```

k=0
while k<10:
    print(«Привет»)
    k += 1

```

- a. 9
- b. 0
- c. 10
- d. Бесконечно

Ключи:

- |          |         |
|----------|---------|
| 1. a     | 7. a    |
| 2. a     | 8. a    |
| 3. a     | 9. a    |
| 4. a,b,c | 10. c   |
| 5. b     | 11. c,d |
| 6. a     | 12. b   |
|          | 13. b   |

### 9. Методическое обеспечение

1. Дидактические материалы
2. Разработки конспектов занятий.
3. Видеоуроки, видеолекции

### 10. Календарно-тематическое планирование

#### Первый модуль

Название раздела	№ занятия	Тема занятия	Дата проведения
Введение. Знакомство со средой Scratch	1	Знакомство, цели и задачи, ТБ и правила поведения.	
	2	Основы работы с ПК	
	3	Знакомство со средой Scratch. Интерфейс	
	4	Знакомство со средой Scratch. Интерфейс	

	5	Основные инструменты встроенного графического редактора программной среды Scratch	
	6	Что такое алгоритм	
Управление спрайтом	7	Спрайт. Управление спрайтом	
	8	Спрайт. Управление спрайтом	
	9	Управление несколькими спрайтами	
	10	Управление спрайтом в лабиринте	
	11	Слой. Работа со слоями	
	12	Взаимодействие объектов	
	13	Создание игры «Лабиринт»	
Основы анимации в Scratch	14	Изучение работы блоков категории «Движение»	
	15	Изучение работы блоков категории «Внешность»	
	16	Смена фона	
	17	Звук	
	18	Диалоги	
	19	Создание анимационного проекта	
	20	Создание анимационного проекта	
Алгоритмы в Scratch	21	Линейные алгоритмы	
	22	Разветвляющиеся алгоритмы	
	23	Циклические алгоритмы	
	24	Ящик «Управление»	
	25	Создание проекта «Балерина»	
Циклы	26	Цикл «повторять всегда»	
	27	Цикл «повторять всегда»	
	28	Цикл «повторить n раз»	
	29	Цикл «повторить n раз»	
	30	Цикл «повторять пока не ...»	
	31	Цикл «повторять пока не ...»	
	32	Анимационный проект, с применением циклов	
Создание	33	Проект «Аквариум»	

анимационных проектов	34	Проект «Природа»	
	35	Проект «Здоровый образ жизни»	
	36	Проект «Игра»	
	37	Проект «Правила дорожного движения»	
	38	Анимированная открытка	
	39	Проект «Случайные числа»	
Переменные	40	Переменные в Scratch	
	41	Команды ящика «Данные»	
	42	Переменные и рычажки	
	43	Создание игры с подсчетом очков	
Списки	44	Создание списков	
	45	Добавление, удаление и редактирование элементов	
	46	Добавление, удаление и редактирование элементов	
	47	Проект «Съедобное и несъедобное»	
Музыка	48	Ноты и паузы в Скретче	
	49	Метроном Скретча	
	50	Мелодические инструменты. Барабаны	
	51	Проект «Музыка»	
	52	Определение целей и задач проекта	
	53	Составление плана работы над проектом	
	54	Подготовка материалов для создания проекта	
	55	Подготовка материалов для создания проекта	
	56	Работа над проектом	
	57	Работа над проектом	
	58	Работа над проектом	
	59	Работа над проектом	
	60	Работа над проектом	
	61	Работа над проектом	

	62	Работа над проектом	
	63	Работа над проектом	
	64	Работа над проектом	
	65	Работа над проектом	
	66	Защита проектов	
	67	Защита проектов	
	68	Итоговое занятие	

### **Второй модуль**

<b>Название раздела</b>	<b>№ занятия</b>	<b>Тема занятия</b>	<b>Дата проведения</b>
Введение в Python	1	Язык Python. Среда программирования.	
	2	Типы данных, операции. Оператор присваивания	
	3	Типы данных, операции. Оператор присваивания	
	4	Числа. Стандартные операции	
	5	Числа. Стандартные операции	
Алгоритмические конструкции	6	Условный оператор.	
	7	Условный оператор.	
	8	Цикл while. Теория чисел	
	9	Цикл while. Теория чисел	
	10	Цикл for	
	11	Цикл for	
Строки	12	Строки. Литералы строк	
	13	Строки. Литералы строк	
	14	Строки. Срезы строк.	
	15	Строки. Методы строк	
Функции	16	Функции. Парадигма и преимущества структурного программирования.	
	17	Граф и стек вызовов функций. Области видимости переменных	
	18	Прямая рекурсия	
	19	Косвенная рекурсия	
Списки и кортежи	20	Списки и кортежи в Python. Сходства и различия	
	21	Операции со списками	
	22	Операции со списками	
	23	Срезы списков	
	24	Матрицы. Операции над матрицами	
	25	Матрицы. Операции над матрицами	
Словари и множества	26	Словари	
	27	Множества	
	28	Множества	
Объектно-ориентированное программирование	29	Классы в Python	
	30	Классы в Python	
	31	Классы в Python	
Проектная деятельность	32	Подготовка проекта	
	33	Создание проекта	
	34	Защита проекта	

## 11. Рабочая программа воспитания

Современный национальный воспитательный идеал — это высоконравственный, творческий, компетентный гражданин России, принимающий судьбу Отечества как свою личную, осознающий ответственность за настоящее и будущее своей страны, укоренённый в духовных и культурных традициях многонационального народа Российской Федерации. Исходя из этого воспитательного идеала, а также основываясь на базовых для нашего общества ценностях (таких как семья, труд, отечество, природа, мир, знания, культура, здоровье, человек) общая цель воспитания в общеобразовательной организации – личностное развитие школьников.

Программа «Программа-плюс» позволяет создать благоприятные условия для развития социально значимых отношений школьников, и, прежде всего, ценностных отношений:

- к труду как основному способу достижения жизненного благополучия человека, залогом его успешного профессионального самоопределения и ощущения уверенности в завтрашнем дне;
- к миру как главному принципу человеческого общежития, условию крепкой дружбы, налаживания отношений с коллегами по работе в будущем и создания благоприятного микроклимата в своей собственной семье;
- к знаниям как интеллектуальному ресурсу, обеспечивающему будущее человека, как результату кропотливого, но увлекательного учебного труда;
- к самим себе как хозяевам своей судьбы, самоопределяющимся и самореализующимся личностям, отвечающим за свое собственное будущее.

## 12. Календарный план

№ п\п	Дела, события, мероприятия	Ориентировочные сроки
1	Урок цифры. «Искусственный интеллект в старпах»	26.09-16.10
2	Урок цифры. «Видеотехнологии»	21.11-11.12
3	Урок цифры. «Искусственный интеллект и метеорология»	16.01-05.02
4	Урок цифры. «Анализ в бизнесе и программной разработке»	13.02-12.03
5	Урок цифры. «Что прячется в смартфоне: исследуем мобильные угрозы»	13.03-02.04
6	Урок цифры. «Квантовые технологии»	05.04-30.04

## 13. Список методической литературы, материально-техническое обеспечение

### Первый модуль

1. Сборник «Информатика. Математика. Программы внеурочной деятельности для начальной и основной школы: 3 – 6 классы» / М.С. Цветкова, О.Б. Богомолова. – М.: БИНОМ. Лаборатория знаний, 2013. – 128 с.: ил.
2. Ю.В. Пашковская «Творческие задания в среде Scratch» . – М.: БИНОМ. Лаборатория знаний, 2014. – 200 с.: ил.
3. Вордерман Кэрл, Вудкок Джон, Макаманус Шон. Переводчик: Ломакин Станислав Программирование для детей: Манн, Иванов и Фербер, 2015 г.
4. Авторская программа «Творческие задания в среде программирования Скретч», которая входит в сборник «Информатика. Математика. Программы внеурочной деятельности для начальной и основной школы: 3 – 6 классы» / М.С. Цветкова, О. Б. Богомолова. – М.: БИНОМ. Лаборатория знаний, 2013. – 128 с.
5. <http://scratch.mit.edu> – официальный сайт Scratch
6. <http://letopisi.ru/index.php/Скретч> - Скретч в Летописи.ру
7. <http://setilab.ru/scratch/category/commun> - Учитесь со Scratch

### Второй модуль

1. К. Вордерман и др. Программирование на Python: Иллюстрированное руководство для детей. Издательство: Манн, Иванов и Фербер, 2018 г.
2. Программирование для детей на языке Python. Издательство: АСТ, 2017 г
3. Д. Бриггс. Python для детей: Самоучитель по программированию. Издательство: Манн, Иванов и Фербер, 2018 г.
4. Б. Пэйн. Python для детей и родителей. Издательство: Эксмо, 2017 г.

5. П. Томашевский. Привет, Python! Моя первая книга по программированию. Издательство: Наука и Техника, 2018 г.
6. <https://pythontutor.ru/>
7. Введение в криптографию. Под редакцией В.В.Яценко Издание четвертое, дополненное, Москва, МЦНМО, 2012.
8. <https://habr.com/ru/company/yandex/blog/324866/>
9. <https://tproger.ru/translations/understanding-cryptography/>
10. <https://younglinux.info/pygame/pygame>
11. <https://habr.com/ru/post/347138/>