



Государственное бюджетное общеобразовательное учреждение  
«Средняя общеобразовательная школа №7 с.п. Пседах»

«РАССМОТРЕНО»  
на заседании  
методического объединения  
протокол № 1 от  
«31» августа 2022г.

«СОГЛАСОВАНО»  
зам.директора по УВР  
«СОШ№7 с.п.Пседах»  
/А.Ю.Гогиева/  
«31» августа 2022г.



**РАБОЧАЯ ПРОГРАММА КУРСА**  
**дополнительного образования «Знакомство со средой**  
**программирования Scratch**  
**на 2022-2023 учебный год**  
**в рамках федерального проекта**  
**«Почка роста»**

### 3-4 классы

#### Кафедра начальной школы

#### Срок реализации: 2021-2022 учебный год

Программа внеурочной деятельности «Scratch программирование» разработана в соответствии с Федеральным государственным образовательным стандартом начального общего образования (приказ Минобрнауки России от 17.12.2010 № 1897) и на основе требований к результатам освоения основной образовательной программы начального общего образования ГБОУ «СОШ № 30 с. п. Сагопши». Данная рабочая программа к курсу «Scratch программирование» предназначена для обучающихся 3 и 4 классов, первый год обучения.

### 1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Актуальность программы состоит в том, что мультимедийная среда Scratch позволяет сформировать у детей стойкий интерес к программированию, отвечает всем современным требованиям объектно-ориентированного программирования. Среда Scratch позволяет сформировать навыки программирования, раскрыть технологию программирования.

Внеурочная деятельность «**Программирование в среде SCRATCH**» (направление: информатика) реализуется в 3 и 4 классах по 1 часу в неделю. Всего 34 ч. «Среда программирования Scratch» является отличной средой для проектной деятельности. В ней есть все необходимое:

- графический редактор для создания и модификации визуальных объектов;
- библиотека готовых графических объектов (некоторые из них содержат наборы скриптов);
- библиотека звуков и музыкальных фрагментов;
- большое количество примеров.

Scratch является отличным инструментом для организации научно-познавательной деятельности школьника благодаря нескольким факторам:

- эта программная среда легка в освоении и понятна даже младшим школьникам, но при этом - она позволяет составлять сложные программы;
- эта программа позволяет заниматься и программированием, и созданием творческих проектов;
- вокруг Scratch сложилось активное, творческое международное сообщество, что позволяет участвовать школьникам в международной конференции по программированию.

Язык Scratch особенно интересен для начального уровня изучения программирования. Обучение основам программирования в этой среде наиболее эффективно при выполнении небольших (поначалу) проектов. При этом естественным образом ученик овладевает интерфейсом новой для него среды, постепенно углубляясь как в возможности Scratch, так и в идеи собственно программирования. Базовый проект един для всех учеников и выполняется совместно с учителем. Затем предлагаются возможные направления развития базового проекта, которые у разных учеников могут быть различными.

При создании сложных проектов ученик не просто освоит азы программирования, но и познакомится с полным циклом разработки программы, начиная с этапа описания идеи и заканчивая тестированием и отладкой.

Scratch не просто среда для программирования, через нее можно выйти на многие другие темы школьной информатики. Важно то, что ребенок имеет возможность поделиться результатами своего творчества с друзьями или другими пользователями.

#### **Цели и задачи программы**

**Цель:** воспитание творческой личности, обогащенной общетехническими знаниями и умениями, развитие индивидуальных творческих способностей, интереса к науке и технике.

Данная программа решает следующие основные задачи:

#### **Образовательные:**

1. овладеть навыками составления алгоритмов;
2. изучить функциональность работы основных алгоритмических конструкций;



3. сформировать представление о профессии «программист»;
4. сформировать навыки разработки программ;
5. познакомить с понятием проекта и алгоритмом его разработки;
6. сформировать навыки разработки проектов: интерактивных историй, квестов, интерактивных игр, обучающих программ, мультфильмов, моделей и интерактивных презентаций.

**Развивающие:**

1. способствовать развитию критического, системного, алгоритмического и творческого мышления;
2. развивать внимание, память, наблюдательность, познавательный интерес;
3. развивать умение работать с компьютерными программами и дополнительными источниками информации;
4. развивать навыки планирования проекта, умение работать в группе.

**Воспитательные:**

1. формировать положительное отношение к информатике и ИКТ;
2. развивать самостоятельность и формировать умение работать в паре или в малой группе;
3. формировать умение, демонстрировать результаты своей работы.

**Количество часов в неделю: 1 (в каждом классе)**

**Количество часов в год: 34**

## 2. РЕЗУЛЬТАТЫ ИЗУЧЕНИЯ ПРЕДМЕТА

УУД	ХАРАКТЕРИСТИКА
Личностные	<ul style="list-style-type: none"> <li>• владение умениями организации собственной учебной деятельности, включающими: целеполагание как постановку учебной задачи на основе соотнесения того, что уже известно, и того, что требуется установить;</li> <li>• планирование – определение последовательности промежуточных целей с учетом конечного результата, разбиение задачи на подзадачи, разработка последовательности и структуры действий, необходимых для достижения цели при помощи фиксированного набора средств;</li> <li>• прогнозирование – предвосхищение результата;</li> <li>• контроль – интерпретация полученного результата, его соотнесение с имеющимися данными с целью установления соответствия или несоответствия (обнаружения ошибки);</li> <li>• коррекция – внесение необходимых дополнений и корректив в план действий в случае обнаружения ошибки;</li> <li>• оценка – осознание учащимся того, насколько качественно им решена учебно-познавательная задача;</li> <li>• владение основными универсальными умениями информационного характера: постановка и формулирование проблемы;</li> <li>• поиск и выделение необходимой информации, применение методов информационного поиска;</li> <li>• структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных условий;</li> <li>• самостоятельное создание алгоритмов деятельности при решении проблем творческого и поискового характера;</li> <li>• владение основами продуктивного взаимодействия и сотрудничества со сверстниками и взрослыми: умение правильно, четко и однозначно сформулировать мысль в понятной собеседнику форме;</li> <li>• умение осуществлять совместную информационную деятельность, в частности при выполнении проекта;</li> </ul>

<p><b>Предметные</b></p>	<ul style="list-style-type: none"> <li>использование коммуникационных технологий в учебной деятельности и повседневной жизни.</li> </ul> <p>1 год обучения: Учащиеся овладевают следующими знаниями, умениями и способами деятельности: знают принципы и структуру Scratch проектов, формы представления и управления информацией в проектах; умеют спроектировать, изготовить и разместить в сети или подготовить для иной формы представления Scratch проекты; владеют способами работы с изученными программами; знают и умеют применять при создании Scratch проектов основные принципы композиции и колористики; способны осуществлять рефлексивную деятельность, оценивать свои результаты, корректировать дальнейшую деятельность по разработке Scratch проектов.</p> <p>2 года обучения: Учащиеся овладевают следующими знаниями, умениями и способами деятельности: владеют специальными знаниями и практическими навыками в области программирования в среде Scratch; знают принципы и структуру проектов, формы представления и управления информацией в проектах в среде Scratch; владеют способами работы с изученными программами и оборудованием в среде Scratch; владеют приемами организации и самоорганизации работы по созданию проектов в среде Scratch; имеют положительный опыт коллективного сотрудничества при разработке проектов в среде Scratch; имеют опыт коллективной разработки и публичной защиты проектов в среде Scratch; способны осуществлять рефлексивную деятельность, оценивать свои результаты, корректировать дальнейшую деятельность по разработке проектов в среде Scratch.</p>
<p><b>Метапредметные</b></p>	<p>изучения курса «Программирование в среде Scratch» являются формирование следующих универсальных учебных действий:</p> <p><b><u>Регулятивные УУД:</u></b></p> <ul style="list-style-type: none"> <li>планирование последовательности шагов алгоритма для достижения цели;</li> <li>поиск ошибок в плане действий и внесение в него изменений.</li> </ul> <p><b><u>Познавательные УУД:</u></b></p> <ul style="list-style-type: none"> <li>моделирование – преобразование объекта из чувственной формы в модель, где выделены существенные характеристики объекта (пространственно-графическая или знаково-символическая);</li> <li>анализ объектов с целью выделения признаков (существенных, несущественных);</li> <li>синтез – составление целого из частей, в том числе самостоятельное достраивание с восполнением недостающих компонентов;</li> <li>выбор оснований и критериев для сравнения, сериации, классификации объектов;</li> <li>подведение под понятие;</li> </ul>



- установление причинно-следственных связей;
  - построение логической цепи рассуждений.
- Коммуникативные УУД:**
- аргументирование своей точки зрения на выбор оснований и критериев при выделении признаков, сравнении и классификации объектов;
  - выслушивание собеседника и ведение диалога;
  - признание возможности существования различных точек зрения и права каждого иметь свою.

### 3. СОДЕРЖАНИЕ УЧЕБНОГО ПРЕДМЕТА

№ п/п	Название раздела, кол-во часов	Содержание раздела
1	<b>Интерфейс программы Scratch (1 ч)</b>	<p>Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.</p> <p>Теория. История создания среды Scratch. Основные базовые алгоритмические конструкции (линейные алгоритмы, с условным оператором, циклического типа с предусловием и постусловием) и их исполнение в среде Scratch. Понятие исполнителя, алгоритма и программы, их назначение, виды и использование. Виды управления исполнителем. Способы записи алгоритма. Основные характеристики исполнителя. Система команд исполнителя. Понятие проект, его структура и реализация в среде Scratch. Основные компоненты проекта Scratch: спрайты и скрипты. Принцип создания анимации и движения объектов. Листинг программы. Сцена. Текущие данные о спрайте. Стилль поворота. Закладки. Панель инструментов, Новый спрайт. Координаты мышки. Режим представления. Окно скриптов. Окно блоков. Блоки стека. Блоки заголовков. Блоки ссылок. Самодостаточные и открытые скрипты.</p>
2	<b>Начало работы в среде Scratch (2 ч)</b>	<p>Сцена. Редактирование фона. Добавление фона из файла. Теория. Сцена. Ширина и высота сцены. Текущие координаты объекта. Редактирование текущего фона. Вставка нового фона из файла. Вставка стандартного фона из библиотечного модуля среды. Рисование фона в графическом редакторе. Создание нескольких фонов в одной сцене.</p> <p>Практика. Создание фона сцены на выбранную учащимся тему.</p> <p><b>3.</b> Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.</p> <p>Теория. Стандартный объект. Спрайты. Список спрайтов. Редактор рисования для создания новых спрайтов. Инструменты рисования (кисточка, линия, текст, эллипс) и редактирования объекта (ластик, заливка, поворот, выбор, печать, пипетка). Центрирование костюма. Масштабирование спрайта. Загрузка на сцену спрайтов из стандартной коллекции среды Scratch. Вставка</p>

		<p>спрайтов из файлов форматов JPG, BMP, PNG, GIF. Выбор случайного спрайта. Удаление спрайтов. Практика. Создание фона сцены и прорисовка основных спрайтов для Scratch-истории.</p>
3	<p><b>Основные скрипты программы Scratch (18 ч).</b></p>	<p>Синий ящик – команды движения. Темно-зеленый ящик – команды рисования.</p> <p>Теория. Команды – <i>идти; повернуться направо (налево); повернуть в направлении; повернуться к; изменить x (y) на; установить x (y) в; если край, оттолкнуться</i>. Принципиальное различие действия команд <i>идти в</i> и <i>плыть в</i>. Назначение сенсоров <i>положение x, положение y и направлении</i>. Команды – <i>очистить, опустить перо, поднять перо, установить цвет пера, изменить цвет пера на, установить цвет пера, изменить тень пера, установить тень пера, изменить размер пера на, установить размер пера, печать</i>.</p> <p>Практика. Создание программ для передвижения спрайтов по сцене. Создание программ для рисования различных фигур.</p> <p>5. Фиолетовый ящик – внешний вид объекта. Оживление объекта с помощью добавления костюмов.</p> <p>Теория. Костюмы спрайта. Копирование и редактирование костюма спрайта с помощью редактора рисования. Переупорядочивание костюмов. Команды – <i>перейти к костюму, следующий костюм, говорить...в течении...секунд, сказать, думать, думать...секунд, изменить... эффект на, установить эффект...в значение, убрать графические эффекты, изменить размер на, установить размер, показаться, спрятаться, перейти в верхний слой, перейти назад на...1 слоев</i>. Назначение сенсоров <i>костюм и размер</i>. Понятие раскадровки движения. Изменение костюма спрайта для имитации движения.</p> <p>Практика. Создание программы для управления внешним видом объекта. Создание Scratch-историй с имитацией хождения и движения объектов.</p> <p>6. Желтый ящик – контроль. Лиловый ящик – добавление звуков.</p> <p>Теория. Кнопка с зеленым флажком и ее назначение. Управление последовательностью выполнения скриптов. Понятие управляющих сообщений. Команды – <i>передать, передать и ждать, когда я получу</i>. Скрипты для создания условных конструкций программы – <i>если, если...или</i>. Скрипты для управления циклами – <i>всегда, повторить, всегда, если, повторять до..</i> Команды – <i>когда клавиша...нажата, когда щелкнут по, ждать...секунд, ждать до, остановить скрипт, остановить все</i>. Загрузка звуков из стандартной коллекции и из файлов жесткого диска. Запись звука через микрофон. Принципиальная разница работы</p>



команд *играть* *звук и играть* *звук* *до*  
завершения. Команды – *остановить все звуки, барабану*  
*играть...тактов,* *оставшиеся...тактов,*  
*ноту...играть...тактов,* *выбрать инструмент,*  
*изменить громкость, установить громкость, изменить*  
*темпа, установить темп.* Назначение  
сенсоров *громкость и темп.*

Практика. Создание программ с элементами управления объектом. Озвучивание Scratch-историй.

7. Использование в программах условных операторов.

Теория. Базовая конструкция ветвление, назначение, виды (полная и неполная форма). Понятие условия. Изменение порядка выполнения скриптов в зависимости от условия. Разветвление листинга программы. Скрипты условных операторов. Использование неполной формы ветвления в системе Scratch.

Практика. Создание программ с изменением последовательного выполнения скриптов при наличии условий.

8. Функциональность работы циклов. Цикличность выполнения действий в зависимости от поставленных условий.

Теория. Циклы с фиксированным числом повторений. Заголовок цикла. Тело цикла. Циклы с условным оператором. Заголовок цикла. Тело цикла. Предусловие и постусловие. Зацикливание.

Практика. Создание программ с использованием циклов с фиксированным числом повторений. Создание программ с использованием циклов с предусловием и постусловием.

9. Зеленый ящик – операторы. Использование арифметических и логических блоков вместе с блоками управления.

Теория. Числа. Строинги. Логические величины. Логические выражения. Арифметические операции. Логические операции. Операции сравнения. Команды для работы со строингами – *слить, буква...в, длинна строки.* Команда *выдать* *случайное от...до.* Использование арифметических и логических блоков в листинге программы. Просмотр полученного результата.

Практика. Создание программ с использованием операций сравнения данных. Создание программ с использованием арифметических данных и логических операций.

10. События. Оранжевый ящик – переменные.

Теория. События в проектах Scratch. Понятие переменных и необходимость их использования в листинге программы. Глобальные и локальные переменные. Имя переменной и правила его формирования. Команды для переменных – *поставить...в, изменить...на, показать переменную, спрятать*

		<p><i>переменную. Удаление переменных. Создание счетчиков с помощью переменных.</i></p> <p>Практика. Разработка сценария Scratch-историй с несколькими событиями. Создание проектов с использованием глобальных и локальных переменных.</p> <p><b>11. Списки.</b></p> <p>Теория. Создание списков и необходимость их использования в проектах Scratch. Добавление в список данных. Удаление данных из списка. Удаление списка. Команды работы со списками – <i>добавить...к, удалить...из, поставить...в...из, заменить элемент...в...на, элемент...из, длина списка.</i></p> <p>Практика. Создание программ-тестов по принципу сравнения данных из нескольких списков.</p> <p><b>12. Голубой ящик – сенсоры. Ввод-вывод данных.</b></p> <p>Теория. Понятие сенсора. Правила применения и область действия команд <i>касается, касается цвета и цвет.касается.</i> Функционал команды <i>спросить...и ждать.</i> Сенсоры <i>мышка по x, мышка по y, мышка нажата?, клавиша...нажата?, расстояние до, перезапустить таймер.</i> Сенсоры, значение которых можно выводить на экран – <i>ответ, таймер, громкость, громко? ...значение сенсора и сенсор....</i> Необходимость ввода данных для их обработки в программе. Ввод данных с помощью команды <i>спросить.</i> Вывод конечного результата обработки с помощью команд <i>говорить</i> и <i>сказать.</i></p> <p>Практика. Создание проектов с использованием значений сенсоров и команды <i>спросить.</i> Создание программ для обработки данных пользователя с выводом на экран конечного результата.</p>
4	<p><b>Работа с несколькими объектами. Синхронизация их работы (4 ч).</b></p>	<p><b>13. Последовательность и параллельность выполнения скриптов.</b></p> <p>Теория. Последовательные и параллельные потоки в программах Scratch. Одновременная и попеременная работа нескольких исполнителей.</p> <p>Практика. Создание Scratch-историй с одновременной и попеременной работой нескольких исполнителей.</p> <p><b>14. Взаимодействие между спрайтами. Управление через обмен сообщениями.</b></p> <p>Теория. Решение проблемы появления новых исполнителей только после того, как старые исполнители выполнили свои действия. Взаимодействие спрайтов с неподвижными объектами с помощью команд <i>касается</i> и <i>касается цвета.</i> Взаимодействие спрайтов с помощью команд <i>передать</i> и <i>когда я получу.</i> Использование сообщений для создания событий.</p> <p>Практика. Создание Scratch-историй с взаимодействием нескольких исполнителей и неподвижных объектов. Создание Scratch-историй с взаимодействием нескольких исполнителей.</p>



5	<p><b>Использование программы Scratch для создания мини-игр (7 ч).</b></p>	<p>15. Виды компьютерных игр. Алгоритмическая разработка листинга программы. Теория. Компьютерные игры – вред или польза. Виды компьютерных игр. Этапы разработки игр программистами. Практика. Алгоритмическая разработка проекта, запись на естественном языке событий и точек взаимодействия героев будущей игры.</p> <p>16. Разработка базовых спрайтов для игры. Формирование базовых скриптов. Теория. Логика создания персонажей для игры. Перевод алгоритма, написанного на естественном языке, в коды Scratch. Практика. Разработка и создание основных спрайтов и их костюмов для будущей игры. Разработка скриптов для спрайтов и объектов.</p> <p>17. Синхронизация работы скриптов для разных спрайтов. Практика. Доработка основного листинга программы с целью установления связей между спрайтами. Тестирование и отладка программы.</p> <p>18. Переход из одной сцены в другую. Создание интерфейса игры. Теория. Односторонний (без возможности вернуться назад) переход из одного пространства в другое. Понятие интерфейса. Элементы интерфейса. Основные принципы дизайна интерфейсов. Обратная связь. Необходимые элементы меню. Практика. Создать программу для перемещения объекта по игровой карте и разработать интерфейс для Scratch-проекта.</p> <p>19. Сообщество Scratch в Интернете. Просмотр и публикация проектов. Теория. Правила работы в сети. Интернет-сообщества. Сообщество Scratch. Регистрация на сайте. Использование заимствованных кодов и объектов. Авторские права. Публикация проектов Scratch. Практика. Регистрация на сайте сообщества Scratch. Просмотр проектов сообщества и публикация собственных проектов.</p>
5	<p><b>Разработка творческого проекта (2 ч)</b></p>	<p>20. Разработка и защита творческого проекта. Разработка и создание программы с использованием подготовленных материалов. Тестирование и отладка проекта. Защита проекта.</p>

#### 4. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ.

##### *Книгопечатная продукция*

- Голиков Д.И. «Scratch для юных программистов», «БХВ-Петербург», Санкт-Петербург, 2017.
- Краля Н. А. Метод учебных проектов как средство активизации учебной деятельности учащихся: Учебно-методическое пособие / Под ред. Ю. П. Дубенского. Омск: Изд-во ОмГУ, 2005. 59 с.

- Матвеева Н. В. Информатика и ИКТ. 3 класс: методическое пособие / Н. В. Матвеева, Е. Н. Челак, Н. К. Конопатова, Л. П. Панкратова. М.: БИНОМ. Лаборатория знаний, 2009. 420 с.

- Патаракин Е. Д. Учимся готовить в среде Скретч (Учебно-методическое пособие). М: Интуит.ру, 2008. 61 с.

- Скретч [Электронный ресурс] // Материал с Wiki-ресурса Letopisi.Ru — «Время вернуться домой». URL: <http://letopisi.ru/index.php/Скретч>

**Средства обучения**

- дидактические материалы (опорные конспекты, проекты примеры, раздаточный материал для практических работ).
- методические разработки (презентации, видеоуроки, flash-ролики).
- сетевые ресурсы Scratch.
- видеохостинг Youtube (видеоуроки «работа в среде Scratch»).
- учебно-тематический план.

**Технические средства обучения**

- Персональный компьютер с принтером, проектор, экран.

**Экранно-звуковые пособия**

- Видеофильмы (в том числе в цифровой форме) по природоведению, истории, обществоведению, этнографии народов России и мира.
- Аудиозаписи в соответствии с содержанием обучения (в том числе в цифровой форме)

**Оборудование класса**

- Ученические столы двухместные с комплектом стульев.
- Стол учительский с тумбой.
- Шкафы для хранения учебников, дидактических материалов, пособий и пр.
- Настенные доски для вывешивания иллюстративного материала.
- Подставки для книг, держатели для карт и т. п.

**Материальное обеспечение программы**

- Ноутбук;
- Доступ к сети Интернет.

**Программное обеспечение**

- Scratch

5. КАЛЕНДАРНО - ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

Класс: 3

Количество часов: всего 34 часа; в неделю 1 час.

№ п/п	СОДЕРЖАНИЕ (РАЗДЕЛЫ, ТЕМЫ)	Кол-во часов
1.	Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.	1
2.	Сцена. Редактирование фона. Добавление фона из файла.	1
3.	Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.	1
4.	Синий ящик – команды движения.	1
5.	Темно-зеленый ящик – команды рисования.	1
6.	Фиолетовый ящик – внешний вид объекта.	1
7.	Оживление объекта с помощью добавления костюмов.	1
8.	Желтый ящик – контроль.	1
9.	Лиловый ящик – добавление звуков.	1



10.	Использование в программах условных операторов.	1
11.	Использование в программах условных операторов.	1
12.	Функциональность работы циклов.	1
13.	Цикличность выполнения действий в зависимости от поставленных условий.	1
14.	Зеленый ящик – операторы.	1
15.	Использование арифметических и логических блоков вместе с блоками управления.	1
16.	События. Оранжевый ящик – переменные.	1
17.	Списки.	1
18.	Голубой ящик – сенсоры. Ввод-вывод данных.	1
19.	Голубой ящик – сенсоры. Ввод-вывод данных.	1
20.	Последовательность и параллельность выполнения скриптов.	1
21.	Последовательность и параллельность выполнения скриптов.	1
22.	Взаимодействие между спрайтами.	1
23.	Взаимодействие между спрайтами. Управление через обмен сообщениями.	1
24.	Виды компьютерных игр. Алгоритмическая разработка листинга программы.	1
25.	Виды компьютерных игр. Алгоритмическая разработка листинга программы.	1
26.	Разработка базовых спрайтов для игры. Формирование базовых скриптов.	1
27.	Разработка базовых спрайтов для игры. Формирование базовых скриптов.	1
28.	Синхронизация работы скриптов для разных спрайтов.	1
29.	Синхронизация работы скриптов для разных спрайтов.	1
30.	Переход из одной сцены в другую. Создание интерфейса игры.	1
31.	Переход из одной сцены в другую. Создание интерфейса игры.	1
32.	Сообщество Scratch в Интернете. Просмотр и публикация проектов.	1
33.	Разработка и защита творческого проекта	1
34.	РЕЗЕРВ	

