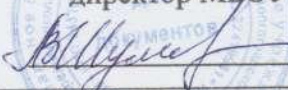


**Администрация Урюпинского муниципального района Волгоградской области
Муниципальное бюджетное общеобразовательное учреждение «Салтынская средняя
школа Урюпинского муниципального района Волгоградской области»**

Рассмотрена на педсовете Протокол № 14 от 26.08.2024	Утверждаю, директор МБОУ Салтынской СШ  Л.В. Шумилина
---	--



**Дополнительная общеобразовательная
общеразвивающая программа
технической направленности
«Программирование в среде Scratch»**

Возраст детей: 8-13 лет
Срок реализации: 1 год

Автор-составитель: Бурдина Наталия Алексеевна,
педагог дополнительного образования

х. Салтынский, 2024 год

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Программа разработана на основе:

- Федерального закона от 29 декабря 2012 года № 273 - ФЗ «Об образовании в Российской Федерации» (редакция от 03.08.2018г.);
- приказа Министерства просвещения Российской Федерации «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» от 09 ноября 2018 года № 196;
- постановления Главного государственного санитарного врача РФ от 4 июля 2014 г. N 41 "Об утверждении СанПиН 2.4.4.3172-14 "Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей";
- с учетом:
- распоряжения правительства Российской Федерации от 04.09 2014 г. № 1726-р «Об утверждении Концепции развития дополнительного образования детей»;
- письма Министерства образования и науки РФ от 18.11. 2015 г. № 09-3242 «О направлении информации»;
- письма Министерства образования и науки РФ от 29.03. 2016 г. № ВК641/09 «О направлении методических рекомендаций»;

Направленность.

Дополнительная общеобразовательная общеразвивающая программа «Программирование в Scratch» имеет техническую направленность.

Программа имеет техническую направленность. Программа ориентирована на развитие технических и творческих способностей и умений учащихся, организацию профессионального самоопределения учащихся.

Актуальность.

Актуальность программы состоит в том, что мультимедийная среда Scratch позволяет сформировать у учащихся интерес к программированию, отвечает всем современным требованиям объектно-ориентированного программирования. Среда Scratch позволяет сформировать навыки программирования, раскрыть технологию программирования. Изучение языка значительно облегчает последующий переход к изучению других языков программирования. Преимуществом Scratch, среди подобных сред программирования, является наличие версий для различных операционных систем, к тому же программа является свободно распространяемой, что немало важно для образовательных учреждений.

Новизна 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 «Увлекательное программирование» практически значимой для современного подростка, т.к. дает возможность увидеть практическое назначение алгоритмов и программ, что будет способствовать развитию интереса к профессиям, связанным с программированием.

Педагогическая целесообразность

Педагогическая целесообразность связана с реализацией следующих возможностей для развития ребенка:

- создание максимального количества ситуаций успеха;
- возможность длительного влияния на формирование личности обучающегося,
- выявление и стимулирование проявлений положительных личностных качеств ребенка,
- практическая значимость (расширение кругозора, использование приобретаемых – качеств, знаний в повседневной жизни),
- предоставление обучающемуся широких возможностей для самовыражения средствами программирования.

Цель программы:

Создание условий для обучения программированию через создание творческих проектов в среде Scratch.

Задачи программы:

Обучающие:

- Овладение базовыми понятиями объектно-ориентированного программирования и применение их при создании проектов в визуальной среде программирования Scratch
- сформировать представление о профессии «программист»;
- сформировать навыки разработки, тестирования и отладки несложных программ;
- сформировать навыки разработки проектов: интерактивных историй, интерактивных игр, мультфильмов, интерактивных презентаций.
- совершенствование навыков работы на компьютере и повышение интереса к программированию.

Развивающие:

- способствовать развитию логического мышления, памяти и умению анализировать;
- создание условия для повышения самооценки обучающегося, реализации его как личности;
- формирование потребности в саморазвитии;
- способствовать развитию познавательной самостоятельности.

Воспитательные:

- формирование культуры и навыки сетевого взаимодействия;
- способствовать развитию творческих способностей и эстетического вкуса подростков;
- способствовать развитию коммуникативных умений и навыков обучающихся.

Отличительные особенности программы

Курс построен таким образом, чтобы помочь учащимся заинтересоваться программированием. Программа «Программирование в среде Scratch» позволяет

создавать собственные программы для решения конкретной задачи. Это является отличительной особенностью данной программы.

Программа предназначена для обучающихся младшего звена без предъявления требований к уровню подготовки. В программе предусматривается определенная последовательность прохождения тем. Занятия состоят из теоретической и практической частей. Для успешной реализации программы используются различные методические разработки и наглядные пособия.

Адрес программы

Возраст детей, участвующих в реализации дополнительной общеобразовательной общеразвивающей программы «Программирование в среде Scratch» 8-13 лет, наполняемость групп 12- 14 человек.

Сроки реализации программы

Срок реализации программы – 1 год.

Форма и режим занятий

Форма обучения: очная.

Формы проведения занятий: аудиторные.

В данной программе используется индивидуальная, групповая и фронтальная формы работы.

Занятия проводятся 1 раз в неделю по 1 академическому часу. Продолжительность занятия 40 минут. Срок освоения программы – 1 год. Количество часов – 34.

Вид программы – **ознакомительный**.

Формы организации образовательного процесса

- групповая,
- фронтальная ,
- индивидуальная,
- парная.

Виды занятий:

- теоретические
- практические
- мастер-классы
- конкурсы.

Цель – обучение программированию через создание творческих проектов в среде Scratch.

Задачи:

Личностные:

- формировать широкие познавательные интересы, инициативу и любознательность, мотивы познания и творчества; готовность и способность учащихся к саморазвитию и реализации творческого потенциала в духовной и предметно-продуктивной деятельности за счет развития их образного, алгоритмического и логического мышления;
- формировать готовность к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики и ИКТ.

Предметные:

- формировать интерес к информатике и ИКТ, стремление использовать полученные знания в процессе обучения другим предметам и в жизни;
- объяснить структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных;

- сформировать умение составлять линейные, разветвляющиеся и циклические алгоритмы управления исполнителями на языке программирования Скретч;
- сформировать умение создавать и выполнять программы для решения несложных алгоритмических задач в программе Скретч.

Метапредметные:

- способствовать формированию основных универсальных умений информационного характера: постановки и формулированию проблемы;
- воспитывать умение самостоятельно создавать алгоритмы деятельности при решении проблем творческого и поискового характера;
- воспитывать умение владения основами продуктивного взаимодействия и сотрудничества со сверстниками и взрослыми: умение правильно, четко и однозначно сформулировать мысль в понятной собеседнику форме;
- способствовать использованию коммуникационных технологий в учебной деятельности и повседневной жизни.

Учебный план

п/п	Название раздела, темы	Количество часов			Форма аттестации/ контроля
		Всего	Теория	Практика	
I	Интерфейс программы Scratch	1			
1.1	Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.	1	1		Анкетирование
II	Начало работы в среде Scratch	2			
2.1	Сцена. Редактирование фона. Добавление фона из файла.	1	1	1	Наблюдение, ответы на вопросы
2.2	Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.	1	1	1	Наблюдение, ответы на вопросы
III	Основные скрипты программы Scratch	18			
3.1	Синий ящик – команды движения.	1	1	1	Наблюдение, ответы на вопросы
3.2	Темно-зеленый ящик – команды рисования.	1	1	1	Наблюдение, ответы на вопросы
3.3.	Фиолетовый ящик – внешний вид объекта.	1	1	1	Наблюдение, ответы на вопросы
3.4	Оживление объекта с помощью добавления костюмов.	1	1	1	Наблюдение, ответы на вопросы
3.5	Желтый ящик – контроль.	1	1	1	Наблюдение, ответы на вопросы
3.6	Лиловый ящик – добавление звуков.	1	1	1	Наблюдение, ответы на вопросы
3.7	Использование в программах условных операторов.	1	1	1	Наблюдение, ответы на вопросы
3.8	Использование в программах условных операторов.	1	1	1	Наблюдение, ответы на вопросы
3.9	Функциональность работы циклов.	1	1	1	Наблюдение, ответы на вопросы
3.10	Цикличность выполнения действий	1	1	1	Наблюдение,

	в зависимости от поставленных условий.				ответы на вопросы
3.11	Зеленый ящик – операторы.	1	1	1	Наблюдение, ответы на вопросы
3.12	Использование арифметических и логических блоков вместе с блоками управления.	1	1	1	Наблюдение, ответы на вопросы
3.13	События. Оранжевый ящик – переменные.	1	1	1	Наблюдение, ответы на вопросы
3.14	Списки.	1	1	1	Наблюдение, ответы на вопросы
3.15	Голубой ящик – сенсоры. Ввод-вывод данных.	1	1	1	Наблюдение, ответы на вопросы
3.16	Голубой ящик – сенсоры. Ввод-вывод данных.	1	1	1	Наблюдение, ответы на вопросы
3.17	Последовательность и параллельность выполнения скриптов.	1	1	1	Наблюдение, ответы на вопросы
3.18	Последовательность и параллельность выполнения скриптов.	1	1	1	Наблюдение, ответы на вопросы
IV	Работа с несколькими объектами. Синхронизация их работы	4			
4.1	Взаимодействие между спрайтами.	1	1	1	Наблюдение, ответы на вопросы
4.2	Взаимодействие между спрайтами. Управление через обмен сообщениями.	1	1	1	Наблюдение, ответы на вопросы
4.3	Виды компьютерных игр. Алгоритмическая разработка листинга программы.	1	1	1	Наблюдение, ответы на вопросы
4.4	Виды компьютерных игр. Алгоритмическая разработка листинга программы.	1	1	1	Наблюдение, ответы на вопросы
V	Использование программы Scratch для создания мини-игр	7			
5.1	Разработка базовых спрайтов для игры. Формирование базовых скриптов.	1	1	1	Наблюдение, ответы на вопросы
5.2	Разработка базовых спрайтов для игры. Формирование базовых скриптов.	1	1	1	Наблюдение, ответы на вопросы
5.3	Синхронизация работы скриптов для разных спрайтов	1	1	1	Наблюдение, ответы на вопросы
5.4	Синхронизация работы скриптов для разных спрайтов.	1	1	1	Наблюдение, ответы на вопросы
5.5.	Переход из одной сцены в другую. Создание интерфейса игры.	1		1	Наблюдение, ответы на вопросы
5.6	Переход из одной сцены в другую. Создание интерфейса игры.	1	1	1	Наблюдение, ответы на вопросы
5.7	Сообщество Scratch в Интернете.	1	1	1	Наблюдение,

	Просмотр и публикация проектов.				ответы на вопросы
VI	Разработка творческого проекта	2			
6.2	Разработка и защита творческого проекта	1	1	1	Наблюдение, ответы на вопросы
6.2	Резерв	1	1	1	Наблюдение, ответы на вопросы
	Всего	34			

Содержание учебной программы

Раздел 1. Интерфейс программы Scratch

Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.

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. Стил поворота. Закладки. Панель инструментов, Новый спрайт. Координаты мышки. Режим представления. Окно скриптов. Окно блоков. Блоки стека. Блоки заголовков. Блоки ссылок. Самодостаточные и открытые скрипты.

Раздел 2. Начало работы в среде Scratch

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.

Практика. Создание фона сцены на выбранную учащимся тему.

Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.

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 спрайтов из файлов форматов JPG, BMP, PNG, GIF. Выбор случайного спрайта. Удаление спрайтов.

Практика. Создание фона сцены и прорисовка основных спрайтов для Scratch-истории.

Раздел 3. Основные скрипты программы Scratch.

- Синий ящик – команды движения. Темно-зеленый ящик – команды рисования.

Теория. Команды – идти; повернуться направо (налево); повернуть в направлении; повернуться к; изменить x (y) на; установить x (y) в; если край, оттолкнуться. Принципиальное различие действия команд идти в и плыть в. Назначение сенсоров положение x, положение y и направления. Команды – 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.

Практика. Создание программ для передвижения спрайтов по сцене. Создание программ для рисования различных фигур.

- Фиолетовый ящик – внешний вид объекта. Оживление объекта с помощью добавления костюмов.

Теория. Костюмы спрайта. Копирование и редактирование костюма спрайта с помощью редактора рисования. Переупорядочивание костюмов. Команды – 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. Назначение сенсоров костюм и размер. Понятие раскадровки движения. Изменение костюма спрайта для имитации движения.

Практика. Создание программы для управления внешним видом объекта. Создание Scratch-историй с имитацией хождения и движения объектов.

- Желтый ящик – контроль. Лиловый ящик – добавление звуков.

Теория. Кнопка с зеленым флажком и ее назначение. Управление последовательностью выполнения скриптов. Понятие управляющих сообщений. Команды – передать, передать и ждать, когда я получу. Скрипты для создания условных конструкций программы – если, если...или. Скрипты для управления циклами – всегда, повторить, всегда, если, повторять до.. Команды – когда клавиша...нажата, когда щелкнут по, ждать...секунд, ждать до, остановить скрипт, остановить все. Загрузка звуков из стандартной коллекции и из файлов жесткого диска. Запись звука через микрофон. Принципиальная разница работы 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 темп на, установить темп. Назначение сенсоров громкость и темп.

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

- 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.

- 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.

- 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, длина 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.

- 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 переменную. Удаление переменных. Создание счетчиков с помощью переменных.

Практика. Разработка сценария Scratch-историй с несколькими событиями. Создание проектов с использованием глобальных и локальных переменных.

- Списки.

Теория. Создание списков и необходимость их использования в проектах Scratch. Добавление в список данных. Удаление данных из списка. Удаление списка. Команды работы со списками – добавить...к, удалить...из, поставить...в...из, заменить элемент...в...на, элемент...из, длина списка.

Практика. Создание программ-тестов по принципу сравнения данных из нескольких списков.

- Голубой ящик – сенсоры. Ввод-вывод данных.

Теория. Понятие сенсора. Правила применения и область действия команд касается, касается цвета и цвет.касается. Функционал команды спросить...и ждать. Сенсоры мышка по х, мышка по у, мышка нажата?, клавиша...нажата?, расстояние до, перезапустить таймер. Сенсоры, значение которых можно выводить на экран – ответ, таймер, громкость, громко? ...значение сенсора и сенсор.... Необходимость ввода данных для их обработки в программе. Ввод данных с помощью команды спросить. Вывод конечного результата обработки с помощью команд говорить и сказать.

Практика. Создание проектов с использованием значений сенсоров и команды спросить.

Создание программ для обработки данных пользователя с выводом на экран конечного результата.

Раздел 4. Работа с несколькими объектами. Синхронизация их работы.

- Последовательность и параллельность выполнения скриптов.

Теория. Последовательные и параллельные потоки в программах Scratch. Одновременная и попеременная работа нескольких исполнителей.

Практика. Создание Scratch-историй с одновременной и попеременной работой нескольких исполнителей.

- Взаимодействие между спрайтами. Управление через обмен сообщениями.

Теория. Решение проблемы появления новых исполнителей только после того, как старые исполнители выполнили свои действия. Взаимодействие спрайтов с неподвижными объектами с помощью команд касается и касается цвета. Взаимодействие спрайтов с помощью команд передать и когда я получу. Использование сообщений для создания событий.

Практика. Создание Scratch-историй с взаимодействием нескольких исполнителей и неподвижных объектов. Создание Scratch-историй с взаимодействием нескольких исполнителей.

Раздел 5. Использование программы Scratch для создания мини-игр.

- 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.

- 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.

- Синхронизация работы скриптов для разных спрайтов.

Практика. Доработка основного листинга программы с целью установления связей между спрайтами. Тестирование и отладка программы.

- 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.

- 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. <https://scratch.mit.edu/>

Раздел 6. Разработка творческого проекта

Разработка и защита творческого проекта. Разработка и создание программы с использованием подготовленных материалов. Тестирование и отладка проекта. Защита проекта.

Ожидаемые результаты обучения по программе

Личностные, предметные и метапредметные результаты освоения дополнительной общеобразовательной (общеразвивающей) программы

Личностные:

- широкие познавательные интересы, инициатива и любознательность, мотивы познания и творчества; готовность и способность учащихся к саморазвитию и реализации творческого потенциала в духовной и предметно-продуктивной деятельности за счет развития их образного, алгоритмического и логического мышления;
- готовность к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики и ИКТ.

Предметные:

- интерес к информатике и ИКТ, стремление использовать полученные знания в процессе обучения другим предметам и в жизни;
- способность связать учебное содержание с собственным жизненным опытом и личными смыслами, понять значимость подготовки в области информатики и ИКТ в условиях развития информационного общества;
- 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;
- прогнозирование – предвосхищение результата;
- контроль – интерпретация полученного результата, его соотнесение с имеющимися данными с целью установления соответствия или несоответствия (обнаружения ошибки);
- коррекция – внесение необходимых дополнений и корректив в план действий в случае обнаружения ошибки;
- оценка – осознание учащимся того, насколько качественно им решена учебно-познавательная задача;
- поиск и выделение необходимой информации, применение методов информационного поиска;
- структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных;
- умение составлять линейные, разветвляющиеся и циклические алгоритмы управления исполнителями на языке программирования Скретч;
- умение создавать и выполнять программы для решения несложных алгоритмических задач в программе Скретч;
- способность к избирательному отношению к получаемой информации за счет умений ее анализа и критичного оценивания; ответственное отношение к информации с учетом правовых и этических аспектов ее распространения;

1.				Беседа, практика	1	Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.	Кабинет Точки Роста	Входной контроль
2.				Беседа, практика	1	Сцена. Редактирование фона. Добавление фона из файла.	Кабинет Точки Роста	Беседа, практическая работа
3.				Беседа, практика	1	Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.	Кабинет Точки Роста	Беседа, практическая работа
4.				Беседа, практика	1	Синий ящик – команды движения.	Кабинет Точки Роста	Беседа, практическая работа
5.				Беседа, практика	1	Темно-зеленый ящик – команды рисования.	Кабинет Точки Роста	Беседа, практическая работа
6.				Беседа, практика	1	Фиолетовый ящик – внешний вид объекта.	Кабинет Точки Роста	Беседа, практическая работа
7.				Беседа, практика	1	Оживление объекта с помощью добавления	Кабинет Точки Роста	Беседа, практическая работа

						КОСТЮМОВ.		
8.				Беседа, практика	1	Желтый ящик – контроль.	Кабинет Точки Роста	Беседа, практическая работа
9.				Беседа, практика	1	Лиловый ящик – добавление звуков.	Кабинет Точки Роста	Беседа, практическая работа
10.				Беседа, практика	1	Использование в программах условных операторов.	Кабинет Точки Роста	Беседа, практическая работа
11.				Беседа, практика	1	Использование в программах условных операторов.	Кабинет Точки Роста	Беседа, практическая работа
12.				Беседа, практика	1	Функциональность работы циклов.	Кабинет Точки Роста	Беседа, практическая работа
13.				Беседа, практика	1	Цикличность выполнения действий в зависимости от поставленных условий.	Кабинет Точки Роста	Беседа, практическая работа
14.				Беседа, практика	1	Зеленый ящик – операторы.	Кабинет Точки Роста	Беседа, практическая работа
15.				Беседа, практика	1	Использование арифметических и логических блоков вместе с блоками управления.	Кабинет Точки Роста	Беседа, практическая работа
16.				Практик	1	События. Оранжевый	Кабинет Точки	Беседа, практическая

				а		ящик – переменные.	Роста	еская работа
17.				Беседа, практика		Списки.	Кабинет Точки Роста	Беседа, практическая работа
18.				Беседа, практика		Голубой ящик – сенсоры. Ввод-вывод данных.	Кабинет Точки Роста	Беседа, практическая работа
19.				Беседа, практика		Голубой ящик – сенсоры. Ввод-вывод данных.	Кабинет Точки Роста	Беседа, практическая работа
20.				Беседа, практика		Последовательность и параллельность выполнения скриптов.	Кабинет Точки Роста	Беседа, практическая работа
21.				Беседа, практика		Последовательность и параллельность выполнения скриптов.	Кабинет Точки Роста	Беседа, практическая работа
22.				Беседа, практика		Взаимодействие между спрайтами.	Кабинет Точки Роста	Беседа, практическая работа
23.				Беседа, практика		Взаимодействие между спрайтами. Управление через обмен сообщениями.	Кабинет Точки Роста	Беседа, практическая работа
24.				Беседа, практика		Виды компьютерных игр. Алгоритмическая разработка листинга	Кабинет Точки Роста	Беседа, практическая работа

						программы.		
25.				Беседа, практика		Виды компьютерных игр. Алгоритмическая разработка листинга программы.	Кабинет Точки Роста	Беседа, практическая работа
26.				Беседа, практика		Разработка базовых спрайтов для игры. Формирование базовых скриптов.	Кабинет Точки Роста	Беседа, практическая работа
27.				Беседа, практика		Разработка базовых спрайтов для игры. Формирование базовых скриптов.	Кабинет Точки Роста	Беседа, практическая работа
28.				Беседа, практика		Синхронизация работы скриптов для разных спрайтов.	Кабинет Точки Роста	Беседа, практическая работа
29.				Беседа, практика		Синхронизация работы скриптов для разных спрайтов.	Кабинет Точки Роста	Беседа, практическая работа
30.				Беседа, практика		Переход из одной сцены в другую. Создание интерфейса игры.	Кабинет Точки Роста	Беседа, практическая работа
31.				Беседа, практика		Переход из одной сцены в другую. Создание интерфейса	Кабинет Точки Роста	Беседа, практическая работа

						игры.		
32.				Беседа, практик а		Сообщество Scratch в Интернете. Просмотр и публикация проектов.	Кабинет Точки Роста	Беседа, практич еская работа
33.				Беседа, практик а		Разработка и защита творческого проекта	Кабинет Точки Роста	Беседа, практич еская работа
34.				Беседа, практик а		Резерв	Кабинет Точки Роста	Беседа

Форма подведения итогов реализации программы:

Представляют портфолио творческих работ, участие в конкурсах. Защита творческих работ. Участие в конкурсах и соревнованиях по программированию в среде Scratch. Представление проектов через размещение на Всемирном Scratch портале.

Условия реализации программы:

1. Кабинет
2. Компьютер (для педагога)
3. 12 Raybook (для детей)
4. Интерактивная доска
5. Программа Scratch

Контрольно измерительные материалы (диагностические материалы)

Критерии оценки знаний, умений и навыков, полученных в результате освоения программы

Высокий уровень – учащийся глубоко изучил учебный материал, последовательно и исчерпывающе отвечает на поставленные вопросы, задание выполняет правильно, уверенно и быстро; владеет логическими операциями, выделять существенные признаки И выделяет самостоятельно закономерности; хорошо ориентируется в изученном материале, может самостоятельно найти нужный источник информации, умеет самостоятельно наблюдать и делать простые выводы; проявляет активный интерес к деятельности, стремится к самостоятельной творческой активности, самостоятельно занимается дома, помогает другим, активно участвует в конкурсах, проявляет доброжелательность.

Средний уровень – учащийся знает лишь основной материал, на заданные вопросы отвечает недостаточно четко и полно, при выполнении практической работы испытывает затруднения, устраняет отдельные неточности с помощью дополнительных вопросов педагога, может допускать ошибки, не влияющие на результат; владеет логическими операциями частично, группирует по несущественным признакам; не всегда может определить круг своего незнания и найти нужную информацию в дополнительных источниках; понимает различные позиции других людей, но не всегда проявляет доброжелательность, дает обратную связь, когда уверен в своих знаниях, проявляет интерес к деятельности, настойчив в достижении цели, проявляет активность только при изучении определенных тем или на определенных этапах работы.

Низкий уровень – учащийся не может достаточно полно и правильно ответить на оставленные вопросы, имеет отдельные представления об изученном материале, при выполнении практической работы задание или не сделано, или допущены ошибки, влияющие на результат; логические операции не сформированы; самостоятельно не может определять круг своего незнания, не может делать самостоятельные выводы; редко понимает и принимает позицию других людей, считая свое мнение единственно верным, присутствует на занятиях, но не активен, выполняет задания только по четким инструкциям и указаниям педагога.

Кадровое обеспечение

Для реализации программы необходимы преподаватели:

- знающие предметную область “Информатика”;

- владеющие методикой преподавания “Информатики и ИКТ” в начальной и средней школе;
- имеющие навыки программирования на любом языке программирования высокого уровня;
- имеющие навыки программирования на Scratch;
- освоившие материал, представляемой программы.

Методическое обеспечение программы

Список литературы

Залогова Л.А. Компьютерная графика: учебное пособие / Л.А. Залогова. - 3-е изд. - Москва: Бином. Лаб. знаний, 2020 - 213 с.

Торгашева Ю. Первая книга юного программиста. Учимся писать программы на Scratch. – СПб.: Питер. 2021. – 128 с.: ил. – (Серия «Вы и ваш ребенок»)

Программирование для детей на языке Scratch/ пер. А. Банкрашкова. – Москва: Издательство АСТ. 2020. – 94, [2] с.: ил.

Голиков Д.Н. Scratch для юных программистов. — СПб.: БХВ-Петербург, 2019. — 192 с.

Электронные ресурсы

Учебник Л.А. Залоговой «Компьютерная графика» <http://www.alleng.ru/d/comp/comp46.htm>

Официальный сайт проекта Scratch – <http://scratch.mit.edu>

Учитесь со Scratch – <https://sites.google.com/a/uvk6.info/scratch/home>

Уроки

по

Скретч

<https://www.youtube.com/playlist?list=PLMIhDclNR1GsZ9CJBZESbm7k3Xpr7awy>