

**Муниципальное казенное общеобразовательное учреждение
«Корневская средняя общеобразовательная школа №1 им. В. Крохина»
Корневского района Курской области**

Рассмотрено и принято
педагогическим советом
Протокол № 1 от 30.08.2023 г.
Председатель
педагогического Совета
_____ З. А. Гудкова

Утверждено
приказом от 01.09.2023 г.
№ 2-78
Директор школы
_____ Т. Н. Подлесных

**Рабочая программа внеурочной деятельности
«Алгоритмика»
направление: общеинтеллектуальное**

4 классы

Учитель Сидаш Виталий Геннадьевич

**Программа реализуется с использованием
оборудования центра «Точка роста»**

на 2023-2024 учебный год

Содержание тем учебного курса

1. Введение. Основные алгоритмические конструкции.

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 проекта, 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 (1 час).

II. Начало работы в среде Scratch.

2. Сцена. Редактирование фона. Добавление фона из файла.

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. Создание фона сцены на выбранную учащимся тему (1 час).

3. Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.

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.

Создание фона сцены и прорисовка основных спрайтов для Scratch-истории (1 час).

III. Основные скрипты программы Scratch.

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

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 *направлении*. 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*. Создание программ для передвижения спрайтов по сцене. Создание программ для рисования различных фигур (2 часа).

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

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 слов*. Назначение сенсоров *костюм* и *размер*. Понятие раскадровки движения. Изменение костюма спрайта для имитации движения. Создание программы для управления внешним видом объекта. Создание Scratch-историй с имитацией хождения и движения объектов (2 часа).

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

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*. Создание программ с элементами управления объектом. Озвучивание Scratch-историй (2 часа).

7. Использование в программах условных операторов.

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. Создание программ с изменением последовательного выполнения скриптов при наличии условий (2 часа).

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

Циклы с фиксированным числом повторений. Заголовок цикла. Тело цикла. Циклы с условным оператором. Заголовок цикла. Тело цикла. Предусловие и постусловие. Заикливание. Создание программ с использованием циклов с фиксированным числом повторений. Создание программ с использованием циклов с предусловием и постусловием (2 часа).

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

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. Создание программ с использованием операций сравнения данных. Создание программ с использованием арифметических данных и логических операций (2 часа).

10. События. Оранжевый ящик – переменные.

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. Разработка сценария Scratch-историй с несколькими событиями. Создание проектов с использованием глобальных и локальных переменных (2 часа).

11. Списки.

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*. Создание программ-тестов по принципу сравнения данных из нескольких списков (2 часа).

12. Голубой ящик – сенсоры. Ввод-вывод данных.

Понятие сенсора. Правила применения и область действия команд *касается, касается цвета и цвет. касается*. Функционал команды *спросить...и ждать*. Сенсоры *мышка по x, мышка по y, 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*.

Создание проектов с использованием значений сенсоров и команды *спросить*. Создание программ для обработки данных пользователя с выводом на экран конечного результата (2 часа).

IV. Работа с несколькими объектами. Синхронизация их работы.

13. Последовательность и параллельность выполнения скриптов.

Последовательные и параллельные потоки в программах Scratch. Одновременная и попеременная работа нескольких исполнителей. Создание Scratch-историй с одновременной и попеременной работой нескольких исполнителей (2 часа).

14. Взаимодействие между спрайтами. Управление через обмен сообщениями.

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. Создание Scratch-историй с взаимодействием нескольких исполнителей и неподвижных объектов. Создание Scratch-историй с взаимодействием нескольких исполнителей (2 часа).

V. Использование программы Scratch для создания мини-игр.

15. Виды компьютерных игр. Алгоритмическая разработка листинга программы.

Компьютерные игры – вред или польза. Виды компьютерных игр. Этапы разработки игр программистами. Алгоритмическая разработка проекта, запись на естественном языке событий и точек взаимодействия героев будущей игры (2 часа).

16. Разработка базовых спрайтов для игры. Формирование базовых скриптов.

Логика создания персонажей для игры. Перевод алгоритма, написанного на естественном языке, в коды Scratch. Разработка и создание основных спрайтов и их костюмов для будущей игры. Разработка скриптов для спрайтов и объектов (2 часа).

17. Синхронизация работы скриптов для разных спрайтов.

Доработка основного листинга программы с целью установления связей между спрайтами. Тестирование и отладка программы (1 час).

18. Переход из одной сцены в другую. Создание интерфейса игры.

Односторонний (без возможности вернуться назад) переход из одного пространства в другое. Понятие интерфейса. Элементы интерфейса. Основные принципы дизайна интерфейсов. Обратная связь. Необходимые элементы меню. Создать программу для перемещения объекта по игровой карте и разработать интерфейс для Scratch-проекта (2 часа).

19. Сообщество Scratch в Интернете. Просмотр и публикация проектов.

Правила работы в сети. Интернет-сообщества. Сообщество Scratch. Регистрация на сайте. Использование заимствованных кодов и объектов. Авторские права. Публикация проектов Scratch. Регистрация на сайте сообщества Scratch. Просмотр проектов сообщества и публикация собственных проектов (1 часа).

VI. Разработка творческого проекта.

20. Разработка и защита творческого проекта. Разработка и создание программы с использованием подготовленных материалов. Тестирование и отладка проекта. Защита проекта (2 часа).

Взаимосвязь с федеральной рабочей программой воспитания

Программа курса разработана с учетом рекомендаций федеральной рабочей программы воспитания, предполагает объединение учебной и воспитательной деятельности педагогов, нацелена на достижение все основных групп образовательных результатов – личностных, метапредметных, предметных.

Программа направлена на:

- установление доверительных отношений между учителем и его учениками, способствующих позитивному восприятию учащимися требований и просьб учителя,

привлечению их внимания к обсуждаемой на уроке информации, активизации их познавательной деятельности;

- побуждение школьников соблюдать на уроке общепринятые нормы поведения, правила общения со старшими (учителями) и сверстниками (школьниками), принципы учебной дисциплины и самоорганизации;
- привлечение внимания школьников к ценностному аспекту изучаемых на уроках явлений, организация их работы с получаемой на уроке социально значимой информацией – инициирование ее обсуждения, высказывания учащимися своего мнения по ее поводу, выработки своего к ней отношения;
- использование воспитательных возможностей содержания учебного предмета через демонстрацию детям примеров ответственного, гражданского поведения, проявления человеколюбия и добросердечности, через подбор соответствующих текстов для чтения, задач для решения, проблемных ситуаций для обсуждения в классе;
- специально разработанные занятия-уроки, занятия-экскурсии, которые расширяют образовательное пространство предмета, воспитывают любовь к прекрасному, к природе, к родному поселку, району;
- организацию предметных образовательных событий (проведение предметных недель) для обучающихся с целью развития познавательной и творческой деятельности в различных сферах предметной деятельности, раскрытия творческих способностей обучающихся с разными образовательными потребностями и индивидуальными возможностями;
- проведение учебных (олимпиады, занимательные уроки и пятиминутки, урок-деловая игра, урок-путешествие, урок-мастер-класс, урок-исследование и др.) и учебно-развлекательных мероприятий (викторины, литературные композиции, экскурсии и др.);
- использование на уроке ИКТ и дистанционных образовательных технологий обучения, обеспечивающих современные активности обучающихся (программы-тренажеры, тесты, зачеты в электронных приложениях, мультимедийные презентации, научно-популярные передачи, фильмы, обучающие сайты, онлайн-уроки, видеолекции, онлайн-конференции и др.);
- применение на уроке интерактивных форм работы учащихся: интеллектуальных игр, стимулирующих познавательную мотивацию школьников; дискуссий, которые дают учащимся возможность приобрести опыт ведения конструктивного диалога; групповой работы или работы в парах, которые учат школьников командной работе и взаимодействию с другими детьми;
- включение в урок игровых процедур, которые помогают поддержать мотивацию детей к получению знаний, налаживанию позитивных межличностных отношений в классе, помогают установлению доброжелательной атмосферы во время урока;
- использование визуальных образов (предметно-эстетической среды, наглядная агитация школьных стендов, предметной направленности, совместно производимые видеоролики по темам урока);
- организацию шефства мотивированных и эрудированных учащихся над их неуспевающими одноклассниками, дающего школьникам социально значимый опыт сотрудничества и взаимной помощи;
- инициирование и поддержка исследовательской деятельности школьников в рамках реализации ими индивидуальных и групповых исследовательских проектов, что даст школьникам возможность приобрести навык самостоятельного решения теоретической проблемы, навык генерирования и оформления собственных идей, навык уважительного отношения к чужим идеям, оформленным в работах других исследователей, навык публичного выступления перед аудиторией, аргументирования и отстаивания своей точки зрения;

- использование технологии «Портфолио» с целью развития самостоятельности, рефлексии и самооценки, планирования деятельности, видения правильного вектора для дальнейшего развития способностей.

Личностные, метапредметные, предметные результаты освоения курса

Личностные:

- формирование ответственного отношения к учению, способности к саморазвитию;
- развитие осознанного и ответственного отношения к собственным поступкам;
- формирование коммуникативной компетентности в процессе образовательной, учебно-исследовательской, творческой и других видов деятельности.

Метапредметные:

- умение самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в познавательной деятельности;
- владение основами самоконтроля, принятия решений;
- умение устанавливать причинно-следственные связи, строить логическое рассуждение;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;
- умение осознанно использовать речевые средства в соответствии с задачей коммуникации;
- владение устной и письменной речью; формирование и развитие далее ИКТ-компетенции.

Предметные:

- умение использовать термины «сообщение», «данные», «алгоритм», «программа»;
- умение составлять неветвящиеся (линейные) алгоритмы управления исполнителями и записывать их на языке программирования; умение использовать логические значения, операции и выражения с ними;
- умение формально выполнять алгоритмы, описанные с использованием конструкций ветвления (условные операторы) и повторения (циклы);
- умение создавать и выполнять программы для решения несложных алгоритмических задач. Таким образом, первое знакомство со средой программирования можно организовать через Scratch, что для детей означает – через игру. Через Scratch можно раскрыть многие вопросы школьной информатики для младших школьников.

Учащиеся познакомятся не только с языком программирования, но и с текстовым, графическим редакторами, элементами пользовательского интерфейса, логикой, новыми математическими понятиями, элементами проектной деятельности.

Благодаря использованию технологии Scratch, дети получают возможность:

- постепенно учиться программированию;
- реализовать свои творческие порывы;
- участвовать в интерактивном процессе создания игр и составления программ на языке, разнообразных историях как индивидуально, так и вместе со своими сверстниками из разных стран;
- получать живой отклик от единомышленников; оценить свои творческие способности.

Деятельность по изучению Скретч происходит в рамках кружка, работа учащихся не оценивается. Имеет место неформальная оценка, обычно производимая самими учениками, когда достигаемые умения используются в последующем в учебной деятельности. Определённо можно сказать, что дети, изучавшие Скретч, отличаются от других своими подходами к решению некоторых учебных задач.