



**Муниципальное автономное общеобразовательное учреждение
«Средняя общеобразовательная школа №3» г. Горнозаводска**

Рассмотрено и принято
на заседании
Педагогического совета
Протокол № 1 от 30.08.2024

УТВЕРЖАЮ

Директор

Дёмина Н.И.



**Дополнительная общеобразовательная программа внеурочной
деятельности технической направленности
«Среда программирования Scratch»
для обучающихся 5-6 классов
на 2024-2025 учебный год
педагог Мелехина Г.В..**

Пояснительная записка

Федеральные государственные образовательные стандарты предусматривают формирование новой образовательной среды школы, включающей не только материальные ресурсы, но и организацию учебной и внеурочной деятельности обучающихся.

Для организации учебной и внеурочной деятельности внимание сосредоточено на развитие у учащихся универсальных учебных действий, путем выбора содержания, методов и средств обучения, которые охватывают познавательные процессы (самостоятельность мышления, способы нахождения и решения проблемы), развитие качеств личности, умение прогнозировать и оценивать результаты своей деятельности, умение взаимодействовать со сверстниками и взрослыми.

Научно-познавательная деятельность ученика, организованная в форме выполнения проектов, является наиболее приемлемым методом для формирования универсальных учебных действий. Включение подростка в проектную деятельность творческого характера позволяет сформировать у него познавательный интерес и исследовательские навыки. А это в свою очередь требует использования определенных средств (инструмента).

В качестве такого инструмента по информатике для учащихся основной школы хорошо подойдет среда программирования Scratch, которая является средством для творчества, моделирования, проектирования и, конечно, программирования.

Популярность сферы программной инженерии неизменно возрастает с каждым годом, и многие родители хотят привить любовь детей к этой профессии с детства. Конечно, заинтересовать ребёнка 10-12 лет текстовыми языками программирования, вроде Python, Java, C++ и так далее довольно непросто, в первую очередь потому, что сложно придумать интересное задание, которое смотивировало бы его.

Поэтому стоит начать с языков “попроще”, то есть с визуальных языков программирования, например, со Scratch о котором речь пойдет ниже. Он понятен детям, легок для освоения и соответствует современным направлениям в программировании. В данном курсе предполагается вести изучение программирования в игровой, увлекательной форме.

Актуальность данной образовательной программы состоит в том, что среда Scratch позволяет формировать навыки программирования, раскрывает технологию программирования. Изучение языка значительно облегчает последующий переход к изучению других сложных языков.

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

Предлагаемый курс по информатике в 5-6 классах «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.

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.

Цели программы:

Развитие логического мышления, творческого и познавательного потенциала подростка.

Задачи:

Образовательные:

- овладение базовыми понятиями объектно-ориентированного программирования и применение их при создании проектов в визуальной среде программирования Scratch;
- приобщение обучающихся к новым технологиям, способным помочь им в реализации собственного творческого потенциала;
- развитие познавательной деятельности учащихся в области новых информационных технологий;
- совершенствование навыков работы на компьютере и повышение интереса к программированию.

Воспитательные:

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

Развивающие:

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

Освоив основы Scratch в 5-6 классе, можно будет использовать ее на уроках других предметов (от математики, физики до литературы, рисования, музыки) в качестве среды для создания моделей явлений, ситуаций и т.д.

Требования к результатам обучения

Личностные результаты:

- готовность и способность учащихся к саморазвитию и реализации творческого потенциала в духовной и предметно-продуктивной деятельности за счет развития образного, алгоритмического и логического мышления;
- готовность к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики и ИКТ;
- интерес к информатике и ИКТ, стремление использовать полученные знания в процессе обучения другим предметам и в жизни;
- готовность к самостоятельным поступкам и действиям, принятию ответственности за их результаты;
- готовность к осуществлению индивидуальной и коллективной информационной деятельности;
- способность к избирательному отношению к получаемой информации за счет умений ее анализа и критичного оценивания;
- способность и готовность к принятию ценностей здорового образа жизни за счет знания основных гигиенических, эргономических и технических условий безопасной эксплуатации средств ИКТ.

Метапредметные результаты:

- владение умениями организации собственной учебной деятельности, включающими: целеполагание как постановку учебной задачи на основе соотнесения того, что уже известно, и того, что требуется установить;
- 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 результата;
- контроль – интерпретация полученного результата, его соотнесение с имеющимися данными с целью установления соответствия или несоответствия;
- коррекция – внесение необходимых дополнений и корректив в план действий в случае обнаружения ошибки;
- оценка – осознание учащимся того, насколько качественно им решена учебно-познавательная задача;
- владение основными универсальными умениями информационного характера: постановка и формулирование проблемы;
- поиск и выделение необходимой информации, применение методов информационного поиска;
- структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных условий;

- самостоятельное создание алгоритмов деятельности при решении проблем творческого и поискового характера;
- владение основами продуктивного взаимодействия и сотрудничества со сверстниками и взрослыми: умение правильно, четко и однозначно сформулировать мысль в понятной собеседнику форме;
- умение осуществлять в коллективе совместную информационную деятельность, в частности при выполнении проекта;
- умение выступать перед аудиторией, представляя ей результаты своей работы с помощью средств ИКТ;
- использование коммуникационных технологий в учебной деятельности и повседневной жизни.

Предметные результаты:

- умение использовать термины «информация», «сообщение», «данные», «алгоритм», «программа»; понимание различий между употреблением этих терминов в обыденной речи и в информатике;
- умение составлять линейные, разветвляющиеся и циклические алгоритмы управления исполнителями на языке программирования Scratch;
- умение использовать логические значения, операции и выражения с ними;
- овладение понятиями класс, объект, обработка событий;
- умение формально выполнять алгоритмы, описанные с использованием конструкций ветвления (условные операторы) и повторения (циклы), вспомогательных алгоритмов;
- умение создавать и выполнять программы для решения несложных алгоритмических задач;
- умение использовать готовые прикладные компьютерные программы и сервисы;
- навыки выбора способа представления данных в зависимости от поставленной задачи.

В результате учебной деятельности, для решения разнообразных учебно-познавательных и учебно-практических задач, у обучающихся будут формироваться и развиваться необходимые универсальные учебные действия и специальные учебные умения, что заложит основу успешной учебной деятельности в средней и старшей школе.

Содержание программы

I. Интерфейс программы Scratch (1 ч)

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

Теория. Основные базовые алгоритмические конструкции и их исполнение в среде Scratch. Понятие исполнителя алгоритма. Виды управления исполнителем. Способы записи алгоритма. Основные характеристики исполнителя. Система команд исполнителя. Понятие проекта, его структуры 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. Панель инструментов.

II. Начало работы в среде Scratch (2 ч)

2. 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.

3. Понятие спрайтов. Добавление новых спрайтов. Рисование новых и редактирование старых объектов.

Теория. Стандартный спрайт. Редактор рисования для создания новых спрайтов и изменения старых. Инструменты рисования в растровом и векторном редакторах. Масштабирование спрайта. Загрузка на сцену спрайтов из стандартной коллекции среды Scratch. Вставка спрайтов из файлов форматов JPG, PNG, GIF. Удаление спрайтов.

Практика. Создание фона сцены и выбор и редактирование основных спрайтов для Scratch-историй.

III. Основные скрипты программы Scratch (15 ч)

4. Синий ящик – команды движения. Темно-зеленый ящик – команды рисования (1,5 ч).

Теория. Команды – *идти; повернуться направо (налево); повернуть в направлении; повернуться к; изменить x (y) на; установить x (y) в; если край, оттолкнуться.* 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.*

Практика. Создание программ для передвижения спрайтов по сцене. Рисование сцены и спрайта с использованием векторного и растрового редактора.

5. Фиолетовый ящик – внешний вид объекта. Оживление объекта с помощью добавления костюмов (1,5 ч).

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 слов.* Назначение сенсоров *костюм* и *размер*. Изменение костюма спрайта для имитации движения.

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

6. Желтый ящик – контроль. Лиловый ящик – добавление звуков (1,5 ч).

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. Назначение сенсоров *громкость* и *темп*.

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

7. Использование в программах условных операторов (1,5 ч).

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

8. Функциональность работы циклов. Цикличность выполнения действий в зависимости от поставленных условий (2 ч).

Теория. Циклы с фиксированным числом повторений. Тело цикла. Циклы с условным оператором. 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 (2 ч).

Теория. Числа. Строки. Логические величины. Логические выражения. Арифметические операции. Логические операции. Операции сравнения. Команды для работы со строками – *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 (1,5 ч).

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-истории с несколькими событиями. Создание проектов с использованием глобальных и локальных переменных.

11. Списки (1,5 ч).

Теория. Создание списков и необходимость их использования в проектах Scratch. Добавление в список данных. Удаление данных из списка. Удаление списка. Команды работы со списками.

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

12. Голубой ящик – сенсоры. Ввод-вывод данных (2 ч).

Теория. 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*.

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

IV. Работа с несколькими объектами. Синхронизация их работы (3 ч)

13. Последовательность и параллельность выполнения скриптов (1,5 ч).

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.

14. Взаимодействие между спрайтами. Управление через обмен сообщениями (1,5 ч).

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.

V. Использование программы Scratch для создания мини-игр (6 ч)

15. Виды компьютерных игр. Алгоритмическая разработка листинга программы (2 ч).

Теория. Виды компьютерных игр. Этапы разработки игр.

Практика. Алгоритмическая разработка проекта, запись на естественном языке событий и точек взаимодействия героев будущей игры.

16. Разработка базовых спрайтов для игры. Формирование базовых скриптов (2 ч).

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.

17. Синхронизация работы скриптов для разных спрайтов (1 ч).

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

18. Переход из одной сцены в другую. Создание интерфейса игры (1 ч).

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.

VI. Подпрограммы и функции (1 ч)

19. Другие блоки в Scratch (1 ч).

Теория. Оптимизация программного кода. Использование подпрограмм в больших программах для выполнения одних и тех же действий.

Практика. Разработка проекта с подпрограммами.

VII. Сообщество Scratch в Интернете (1 ч)

20. 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.

VIII. Командная разработка игр в Scratch (2 ч)

21. Теория. Импорт и экспорт в Scratch. Перенос из одного проекта в другой скриптов, спрайтов, костюмов, фонов с помощью Рюкзака Scratch в режиме онлайн. Создание студий для эффективной организации командной работы.

Практика. Использование сайта сообщества Scratch для работы в команде. Умение создавать проект из частей. Умение пользоваться Рюкзаком.

IX. Разработка творческого проекта (3 ч)

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

Планируемые результаты обучения

По окончании курса учащиеся должны научиться составлять линейные, разветвляющиеся и циклические алгоритмы управления исполнителями на языке программирования Scratch, публиковать свои проекты в глобальной сети.

У учащихся должен быть сформирован познавательный интерес к предмету информатика. Полученные знания и умения учащихся, будут способствовать развитию алгоритмического мышления и формированию информационной культуры школьников.

Учащиеся должны научиться аргументировать свою точку зрения, отстаивать свою позицию, уметь организовывать и планировать учебное сотрудничество с учителем и сверстниками, определять цели и функции участников коллективного проекта, способы взаимодействия.

Учебно-методическое обеспечение

1. Патаракин Е.П. «Учимся готовить в среде Скретч - Версия 2.0»

2. Голиков Д. В. «Программирование на Scratch 2»
3. Пашковская Ю.В. «Творческие задания в среде Scratch: рабочая тетрадь 5-6 класс»
4. Рындак В. Г., Дженжер В. О., Денисова Л. В. Проектная деятельность школьника в среде программирования Scratch
5. <http://scratch.mit.edu/> - официальный сайт проекта Scratch
6. <http://qps.ru/97p6r/> -изучение команд среды
7. <http://scratch.ucoz.net>-изучение команд среды