

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Министерство образования и науки Пермского края
Районное управление образования Пермского муниципального округа
МАОУ «Усть-Качкинская средняя школа»

«Точка роста»

РАССМОТРЕНО:

Руководитель школьного
методического
объединения

 Мартынова Е.П.

СОГЛАСОВАНО:

Руководитель Центра
«Точка роста»

 Луначева В.Ю.

УТВЕРЖДЕНО:

Приказ 366/1
от « 29 » августа 2025 г.

Директор МАОУ «Усть-
Качкинская средняя
школа»

 Байдина Т.Г.



Рабочая программа дополнительного образования

«Первые шаги в Scratch»

Составила учитель начальных
классов

Маслаускене Оксана Юрьевна
Целевая аудитория: 3 - 5 классы
Срок реализации: 32 часа

Усть-Качка, 2025

Пояснительная записка

Направленность программы: настоящая программа «Первые шаги в Scratch» является дополнительной общеразвивающей программой технической направленности, относящейся к стартовому уровню реализации.

Актуальность программы состоит в том, что мультимедийная среда Scratch позволяет сформировать у детей стойкий интерес к программированию, отвечает всем современным требованиям объектно-ориентированного программирования. Особенность среды Scratch, позволяющая создавать в программе мультфильмы, анимацию и даже простейшие игры, делает образовательную программу по программированию практически значимой для современного учащегося, т.к. дает возможность увидеть практическое назначение алгоритмов и программ, что будет способствовать развитию интереса к профессиям, связанным с программированием.

Разработка дополнительной общеразвивающей программы «Первые шаги в Scratch» велась на основе изучения программ «Творческие задания в среде программирования Скретч», изданной в сборнике «Информатика. Математика. Программы внеурочной деятельности для начальной и основной школы: 3-6 класс» М.С.Цветкова, О.Б.Богомолова. – М.: БИНОМ. Лаборатория знаний, 2015.

В результате определились следующие отличительные особенности программы «Первые шаги в Scratch»:

- возрастной адресат программы, ее содержание адаптировано для детей 9–11 летнего возраста;
- программа построена таким образом, чтобы помочь учащимся заинтересоваться программированием вообще и найти ответы на вопросы, с которыми им приходится сталкиваться в повседневной жизни при работе с большим объемом информации при решении практических и жизненных задач;
- программа основана на использовании среды Scratch при обучении детей, что позволяет создавать собственные программы для решения конкретной задачи.

Педагогическая целесообразность данной программы состоит в том, что изучая программирование в среде Scratch, у учащихся формируется не только логическое мышление, но и навыки работы с мультимедиа; создаются условия для активного, поискового учения, предоставляются широкие возможности для разнообразного программирования.

Scratch является отличным трамплином для плавного перехода в мир настоящих программистов.

Программа «Первые шаги в Scratch» адресована обучающимся начальных классов.

Возраст детей, участвующих в реализации данной программы 9-11 лет. Дети в этом возрасте активны, любят приключения, игры. Развиваются интеллектуально: им нравится исследовать все, что незнакомо. Понимают законы последовательности, хорошо мыслят. Активно идет процесс социального созревания - начинают быть самостоятельными.

Набор в группы осуществляется на добровольной основе, то есть принимаются все желающие заниматься.

Форма занятий

Занятия по программе «Первые шаги в Scratch» проводятся 1 раз в неделю, часовая нагрузка 32 часа в год. Программа рассчитана на 1 год обучения. Исходя из санитарно-гигиенических норм, продолжительность часа занятий для учащихся 9-11 лет - 45 минут. Занятия проводятся в очной форме. При реализации программы используются индивидуальная, групповая и фронтальная формы работы.

В качестве основных **методов обучения** применяются объяснительно-иллюстративный, метод проблемного изложения, метод мозгового штурма, частично-поисковый (эвристический), проектный метод. Обсуждение рациональных алгоритмов формирует у учащихся навыки оптимизации алгоритмов и способствует формированию структурного типа мышления.

Демонстрация нетипичных (нетрадиционных) способов решения задач стимулирует творческий потенциал учащихся. Обязательным для каждого обучающегося является создание программных продуктов на языке Scratch как результат реализации собственных проектов. Проектная технология позволяет закрепить умения учащихся по работе в графических редакторах, самостоятельно выполнять творческий проект; ориентироваться в информационном

пространстве, кроме этого добиваться метапредметной результативности освоения программы и развивать творческое мышление.

Цели и задачи

Цель: Развитие творческих способностей и формирование раннего профессионального самоопределения обучающихся в процессе программирования.

Задачи обучающие:

- формирование представлений об основных предметных понятиях («информация», «алгоритм», «исполнитель») и их свойствах;
- изучение основных принципов алгоритмизации, знакомство со структурами (линейной, условной и циклической);
- изучение функциональности работы основных алгоритмических конструкций;
- формирование умения составить и записать алгоритм для конкретного исполнителя;
- обучение созданию простых программ, игр.

Развивающие:

- развитие логических способностей и алгоритмического и мышления;
- развитие пространственных представлений, навыков геометрических построений и моделирования таких процессов, развитие изобразительных умений с помощью средств ИКТ;
- формирование информационной и алгоритмической культуры, развитие основных навыков использования компьютерных устройств и программ.

Воспитательные:

- воспитание интереса к программированию как к ключевой технологии XXI века, стремления использовать полученные знания, умения и навыки в учебной деятельности и в повседневной жизни;
- ответственное отношение к учению, способность довести до конца начатое дело на примере завершённых творческих учебных проектов;
- способность к саморазвитию и самообразованию.

Ожидаемые результаты освоения образовательной программы

Обучающиеся будут **знать** основные понятия в программировании и алгоритмизации процессов.

Обучающиеся будут **уметь** создавать простые программы, игры;

У обучающихся будет **развиваться** творческий подход к решению задач по программированию.

Воспитываться информационная культура, а также сознательное отношение к выбору других образовательных программ по программированию. Содержание практических занятий ориентировано не только на овладение учащимися навыками программирования, но и на подготовку их как грамотных пользователей ПК.

Кроме того, обучающиеся получают опыт участия в соревнованиях и проектах, у них будет формироваться коммуникативная компетентность в общении и сотрудничестве со сверстниками в процессе образовательной и проектной деятельности, участия в конкурсах и соревнованиях различного уровня.

Обучающиеся приобретают умения самостоятельно ставить и формулировать для себя новые задачи, планировать пути решения поставленной проблемы для получения эффективного результата, критически оценивать правильность решения задачи, корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями.

В ходе обучения формируются и развиваются компетентности в области использования информационно коммуникационных технологий; умение организовывать учебное сотрудничество и совместную деятельность с педагогом.

Контроль и оценка результатов обучения

Система отслеживания результатов: определение начального уровня знаний, умений и навыков, промежуточный и итоговый контроль индивидуальных работ.

Формы подведения итогов

Итоговый контроль осуществляется в форме индивидуальных проектов.

Тематическое планирование

№	Тема	Количество часов
1	Интерфейс программы Sretch	1
2	Начало работы в среде Sretch	2
3	Основные скрипты программы Sretch	15
4	Работа с несколькими объектами. Синхронизация их работы	4
5	Использование программы Sretch для создания мини-игр	8
6	Разработка собственного проекта	2
Итого:		32

Учебно-тематический план дополнительной общеразвивающей программы

«Первые шаги в Scratch»

№ п/п	Наименование темы	Количество часов		
		теория	практика	всего

1	Введение. Техника безопасности при работе с компьютером. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.	1	0	1
2	Сцена. Редактирование фона. Добавление фона из файла.	0,5	0,5	1
3	Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.	0,5	0,5	1
4	Команды движения. Команды рисования.	0,5	1,5	2
5	Внешний вид объекта. Оживление объекта с помощью добавления костюмов.	0,5	1,5	2
6	Контроль. Добавление звуков.	0,5	1,5	2
7	Функциональность работы циклов. Цикличность выполнения действий в зависимости от поставленных условий.	0,5	1,5	2
8	Операторы. Использование арифметических и логических блоков вместе с блоками управления.	0,5	1,5	2
9	События. Переменные.	0,5	1,5	2
10	Списки.	0,5	0,5	1
11	Сенсоры. Ввод-вывод данных.	0,5	1,5	2
12	Последовательность и параллельность выполнения скриптов.	0,5	1,5	2
13	Взаимодействие между спрайтами.	0, 5	1,5	2
14	Виды компьютерных игр. Алгоритмическая разработка листинга программы.	0,5	1, 5	2
15	Разработка базовых спрайтов для игры. Формирование базовых скриптов.	0,5	1,5	2

16	Синхронизация работы скриптов для разных спрайтов.	0, 5	1, 5	2
17	Переход из одной сцены в другую. Создание интерфейса игры.	0,5	1,5	2
18	Разработка и защита творческого проекта	0	2	2
Итого:		9	23	32

Содержание программы учебного курса

I. Интерфейс программы Scratch (1 ч).

1. Введение. Правила техники безопасности при работе с компьютером. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.
Теория. История создания среды Scratch. Основные базовые алгоритмические конструкции.

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.
Практика. Создание фона сцены на выбранную детьми тему.
3. 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).
Практика. Создание фона сцены и прорисовка основных спрайтов для Scratch-истории.

III. Основные скрипты программы Scratch (15 ч).

4. Команды движения. Команды рисования.

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

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

5. Внешний вид объекта. Оживление объекта с помощью добавления костюмов.

Теория. Костюмы спрайта. Копирование и редактирование костюма спрайта с помощью редактора рисования. 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.

6. Контроль. Добавление звуков.

Теория. Кнопка с зеленым флажком и ее назначение. Управление последовательностью выполнения скриптов. Понятие управляющих сообщений. 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. Функциональность работы циклов. Цикличность выполнения действий в зависимости от поставленных условий.

Теория. Циклы с фиксированным числом повторений. Заголовок цикла. Тело цикла. Зацикливание.

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

8. Операторы. Использование арифметических и логических блоков вместе с блоками управления.

Теория. Числа. Использование арифметических и логических блоков в листинге программы.

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

9. События. Переменные.

Теория. События в проектах Scratch. Понятие переменных и необходимость их использования в листинге программы. Создание счетчиков с помощью переменных.

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

10. Списки.

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

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

11. Сенсоры. Ввод-вывод данных.

Теория. Понятие сенсора.

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 (4 ч).

12. Последовательность и параллельность выполнения скриптов.

Теория. Последовательные и параллельные потоки в программах Scratch.

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

13. 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.

V. Использование программы Scratch для создания мини-игр (8 ч).

14. 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.

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

Теория. Логика создания персонажей для игры. Перевод алгоритма, написанного на естественном языке, коды Scratch.

Практика. Разработка и создание основных спрайтов и их костюмов для будущей игры. Разработка скриптов для спрайтов и объектов

16. Синхронизация работы скриптов для разных спрайтов.

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

17. 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

VI. Разработка творческого проекта (2 ч)

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

Методическое обеспечение программы

Для реализации программы используются следующие **методы обучения**:

- **по источнику полученных знаний**: словесные, наглядные, практические.
- **по способу организации познавательной деятельности**:
- развивающее обучение (проблемный, проектный, творческий, частично-поисковый, исследовательский, программированный);
- дифференцированное обучение (уровневые, индивидуальные задания).
- игровые методы (конкурсы, игры-конструкторы, турниры с использованием мультимедиа).

Средства обучения:

- дидактические материалы (опорные конспекты, проекты примеры, раздаточный материал для практических работ).
- методические разработки (презентации, видеоуроки).
- сетевые ресурсы Scratch.

Материальное обеспечение программы

1. компьютер;
2. доступ к сети Интернет.

Программное обеспечение: Scratch

Список используемой литературы

Для учителя:

1. «Раннее обучение программированию в среде Scratch», В.Г. Рындак, В.О. Джинжер, Л.В. Денисова, 2019.
2. Голиков Д.И. «Scratch для юных программистов», «БХВ-Петербург», Санкт-Петербург, 2017.
3. 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.
4. Школа Scratch [Электронный ресурс] // Материал с Wiki-ресурса Letopisi.Ru — «Время вернуться домой».
URL: http://letopisi.ru/index.php/Школа_Scratch

Для ученика:

1. Голиков Д.В. Scratch для юных программистов. – СПб: БХВ-Петербург, 2018.
2. Свейгарт, Эл. Программирование для детей: делай игры и учи язык Scratch! – М.: Эксмо, 2020.
3. Маржи, Мажед. Scratch для детей. Самоучитель по программированию. – М.: Манн, Иванов и Фербер, 2018.