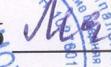


Муниципальное общеобразовательное учреждение
средняя школа №6 Тутаевского муниципального района

Утверждена

Директор МОУ СШ №6  Е.В. Манокина

Приказ по школе № 240/01-07

от 30.08.2023



**Дополнительная общеобразовательная общеразвивающая программа
«Программируем на Scratch»**

Для детей – 7-13 лет

Срок реализации – 1 год

Количество часов за учебный год – 34 часа

Составитель: учитель
Информатики
Кононова С.Р.

Тутаев 2023

Пояснительная записка

Существуют множество языков программирования, которые предназначены для выполнения конкретных задач. Scratch (Скретч) гораздо легче, чем все традиционные языки программирования. В Скретче можно сочинять истории, рисовать и оживлять на экране придуманные персонажи, создавать презентации, игры, в том числе интерактивные.

Скретч легко перекидывает мостик между программированием и другими школьными науками. Так возникают межпредметные проекты. Готовыми проектами, созданными по этой технологии, можно обмениваться внутри сообщества, объединяющего весь мир, которое активно формируется сейчас в Интернете. Scratch полностью бесплатен, его свободно можно загрузить с сайта разработчиков, как версию для Windows, так и для Linux и др.

Поскольку любой персонаж в среде Скретч может выполнять параллельно несколько действий – двигаться, поворачиваться, изменять цвет, форму и т.д., юные скретчисты учатся мыслить любое сложное действие как совокупность простых. В результате они не только осваивают базовые концепции программирования (циклы, ветвления, логические операторы, случайные числа, переменные, массивы), которые пригодятся им при изучении более сложных языков, но и знакомятся с полным циклом решения задач, начиная с этапа описания идеи и заканчивая тестированием и отладкой программы.

Одной из важнейших особенностей Scratch как языка программирования является его событийно-ориентированный характер. Это означает, что все объекты взаимодействуют при помощи обмена сообщениями. Такая схема обмена информацией делает Scratch близким к современным объектно-ориентированным языкам.

Рабочая программа курса «Программируем на Scratch» разработана на основе программы курса «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, 2013 г., а также Т.Е. Сорокина, поурочные разработки «Пропедевтика программирования со Scratch» для 5-го класса, 2015 г.

Цель программы: создать условия для формирования личностных, предметных компетенций, универсальных учебных действий в ходе проектной научно-познавательной деятельности учеников посредством изучения среды программирования Scratch.

Задачи программы:

- ✓ формирование у обучающихся базовых представлений о языках программирования, алгоритме, исполнителе, способах записи алгоритма;
- ✓ изучение основных базовых алгоритмических конструкций;
- ✓ знакомство с понятием переменной и команды присваивания;
- ✓ овладение навыками алгоритмизации задачи;
- ✓ овладение понятиями класс, объект, обработка событий;
- ✓ освоение основных этапов решения задачи;
- ✓ формирование навыков разработки, тестирования и отладки несложных программ;
- ✓ знакомство с понятием проекта, его структуры, дизайна и разработки;
- ✓ освоение навыков планирования проекта, умение работать в группе;
- ✓ выработка навыков работы в сети для обмена материалами работы;
- ✓ предоставление возможности самовыражения в компьютерном творчестве.

Так как базовый курс информатики начинается с 7 класса, то программа курса дополняет потребность школьников в развитии алгоритмической, логической и познавательной деятельности и ориентирована на получение нового образовательного результата в рамках ФГОС.

Программа рассчитана на 34 часа в год, 1 час в неделю.

Планируемые результаты освоения программы

Личностные результаты

Личностные результаты имеют направленность на решение задач воспитания, развития и социализации обучающихся средствами предмета.

1. Гражданское воспитание:

- представление о социальных нормах и правилах межличностных отношений в коллективе, в том числе в социальных сообществах; готовность к разнообразной совместной деятельности при выполнении учебных, познавательных задач, создании учебных проектов; стремление к взаимопониманию и взаимопомощи в процессе этой учебной деятельности; готовность оценивать своё поведение и поступки своих товарищей с позиции нравственных и правовых норм с учётом осознания последствий поступков.

2. Патриотическое воспитание:

- ценностное отношение к отечественному культурному, историческому и научному наследию; понимание значения информатики как науки в жизни современного общества; владение достоверной информацией о передовых мировых и отечественных достижениях в области информатики и информационных технологий; заинтересованность в научных знаниях о цифровой трансформации современного общества.

3. Духовно–нравственное воспитание:

- ориентация на моральные ценности и нормы в ситуациях нравственного выбора; готовность оценивать своё поведение и поступки, а также поведение и поступки других людей с позиции нравственных и правовых норм с учётом осознания последствий поступков; активное неприятие асоциальных поступков, в том числе в сети Интернет.

4. Эстетическое воспитание:

- Эстетическое отношение к миру, готовность к эстетическому обустройству собственного быта.

5. Ценности научного познания:

- сформированность мировоззренческих представлений об информации, информационных процессах и информационных технологиях, соответствующих современному уровню развития науки и общественной практики и составляющих базовую основу для понимания сущности научной картины мира; интерес к обучению и познанию; любознательность; готовность и способность к самообразованию, исследовательской деятельности, осознанному выбору направленности и уровня обучения в дальнейшем; сформированность информационной культуры, в том числе навыков самостоятельной работы с учебными текстами, справочной литературой, разнообразными средствами информационных технологий, а также умения самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учёбе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности.

6. Формирование культуры здоровья:

- осознание ценности жизни; ответственное отношение к своему здоровью; установка на здоровый образ жизни, в том числе и за счёт освоения и соблюдения требований безопасной эксплуатации средств информационных и коммуникационных технологий (ИКТ).

7. Трудовое воспитание:

- интерес к практическому изучению профессий и труда в сферах профессиональной деятельности, связанных с информатикой, программированием и информационными технологиями, основанными на достижениях науки информатики и научно-технического прогресса; осознанный выбор и построение индивидуальной траектории образования и жизненных планов с учётом личных и общественных интересов и потребностей.

8. Экологическое воспитание:

- осознание глобального характера экологических проблем и путей их решения, в том числе с учётом возможностей ИКТ.

Предметные результаты

Основные предметные результаты, формируемые в процессе изучения «Программируем на Scratch» направлены на:

- осознание значения математики и информатики в повседневной жизни человека;
- формирование представлений об основных предметных понятиях — «информация», «алгоритм», «модель» и их свойствах;
- развитие логических способностей и алгоритмического мышления, умений составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами — линейной, условной и циклической;
- развитие представлений о числах, числовых системах;
- овладение символьным языком алгебры, умение составлять и использовать сложные алгебраические выражения для моделирования учебных проектов, моделировать реальные ситуации на языке алгебры;
- 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;
- формирование умения соблюдать нормы информационной этики и права.

Содержание курса

Раздел 1. Знакомство с программной средой Scratch – 2 ч

Свободное программное обеспечение. Авторы программной среды Scratch. Параметры для скачивания и установки программной среды на домашний компьютер.

Основные элементы пользовательского интерфейса программной среды Scratch. Внешний вид рабочего окна. Блочная структура систематизации информации. Функциональные блоки. Блоки команд, состояний, программ, запуска, действий и исполнителей. Установка русского языка для Scratch.

Создание и сохранение документа. Понятия спрайта, сцены, скрипта. Очистка экрана.

Основной персонаж как исполнитель программ. Система команд исполнителя (СКИ). Блочная структура программы. Непосредственное управление исполнителем.

Библиотека персонажей. Сцена и разнообразие сцен, исходя из библиотеки данных. Систематизация данных библиотек персонажей и сцен. Иерархия в организации хранения костюмов персонажа и фонов для сцен. Импорт костюма, импорт фона.

Аналитическая деятельность:

- выделять аппаратное и программное обеспечение компьютера;
- определять технические устройства для ввода и вывода информации;
- понимать иерархическую организацию библиотеки данных программной среды;
- выделять путь к элементам библиотеки;
- выделять фрагменты изображения для дальнейшей работы с ними;
- планировать работу по созданию сложных изображений путем копирования и масштабирования простых;
- выбирать наиболее подходящий инструмент графического редактора для создания фрагмента изображения;
- различать верхний и нижний цвета изображения;
- придумывать и создавать различные градиенты для заливки замкнутой области;
- планировать создание симметричных изображений.

Практическая деятельность:

- выбирать и запускать программную среду Scratch;
- работать с основными элементами пользовательского интерфейса программной среды;
- изменять размер и перемещать окно программы, выбирать необходимый режим окна;
- вводить имя файла с помощью клавиатуры;
- выбирать необходимый файл из нужной папки библиотеки программы;
- создавать, копировать, переименовывать, перемещать, копировать и удалять файлы;
- соблюдать требования техники безопасности при работе в компьютерном классе.

Раздел 2. Компьютерная графика – 5 ч

Компьютерная графика. Векторные и растровые графические редакторы. Встроенный растровый графический редактор. Основные инструменты графического редактора — кисточка, ластик, заливка (цветом или градиентом), рисование линий, прямоугольников, квадратов, эллипсов и окружностей, выбор фрагмента изображения и отражение его по горизонтали или вертикали, использование инструмента печать для копирования выделенной области изображения, работа с текстом. Масштаб фрагмента изображения. Палитра цветов, установка цвета переднего плана и фона, выбор цвета из изображения с помощью инструмента пипетка. Изменение центра костюма. Изменение размера костюма.

Основные возможности изменения внешнего вида исполнителя: 1) использование встроенной библиотеки данных путём импорта её элемента; 2) редактирование выбранного элемента с помощью инструментов встроенного растрового графического редактора; 3)

создание собственных изображений в других программах (например, LibreOfficeDraw) и импортирование их в программную среду Scratch.

Знакомство с основными графическими примитивами векторного редактора LibreOfficeDraw. Возможность создания геометрических фигур без внутренней заливки, но с текстовым блоком внутри. Стрелки, их направление.

Аналитическая деятельность:

- выделять фрагменты изображения для дальнейшей работы с ними;
- планировать работу по созданию сложных изображений путем копирования и масштабирования простых;
- выбирать наиболее подходящий инструмент графического редактора для создания фрагмента изображения;
- различать верхний и нижний цвета изображения;
- придумывать и создавать различные градиенты для заливки замкнутой области;
- планировать создание симметричных изображений.

Практическая деятельность:

- использовать простейшие растровые и векторные редакторы для создания и редактирования изображений;
- изменять центр изображения;
- вносить изменения в изображения из встроенной библиотеки;
- создавать сложные графические объекты путем копирования и модификации простых объектов и их фрагментов,
- использовать возможности работы с цветом.

Раздел 3. Алгоритмы и исполнители – 14 ч

Алгоритм. Понятие алгоритма как формального описания последовательности действий исполнителя, приводящих от исходных данных к конечному результату. Схематическая запись алгоритма. Использование геометрических фигур для схематической записи алгоритма. Создание блок-схем в свободном векторном редакторе LibreOfficeDraw.

Линейные алгоритмы

Основные признаки линейного алгоритма. Схематическое описание линейного алгоритма. Геометрические примитивы, используемые для описания линейного алгоритма.

Программное управление исполнителем. Создание программ для перемещения исполнителя по экранному полю. Понятие поворота исполнителя в определенное направление. Прямой угол. Поворот исполнителя на прямой угол по часовой стрелке и против часовой стрелки.

Создание программ для рисования линий. Изменение цвета и толщины рисуемой линии. Особенности пунктирной линии. Написание программы для исполнителя, чтобы он оставлял пунктирную линию при перемещении по экранному полю.

Прямоугольник, квадрат — основные черты. Написание программ для движения исполнителя вдоль сторон квадрата, прямоугольника. Внесение изменений в программу рисования квадрата, если необходимо получить другой размер стороны квадрата.

Прерывание программы.

Циклические алгоритмы

Многократное повторение команд как организация цикла. Особенности использования цикла в программе. Упрощение программы путём сокращения количества команд при переходе от линейных алгоритмов к циклическим.

Схематическая запись циклического алгоритма.

Типы циклических алгоритмов. Основные конструкции программной среды, используемые для написания программ исполнителем с применением циклов.

Конечный цикл. Сокращение программы для исполнителя, рисующего линии, квадраты, прямоугольники при использовании цикла. Программа исполнителя для рисования нескольких однотипных геометрических фигур, например, нескольких квадратов из одной вершины, но с различным значением стороны.

Конструкции программной среды спрятаться/показаться. Выполнение программы исполнителем, не показанным на поле выполнения программы.

Написание и отладка программ с применением конструкции цикл в цикле.

Бесконечный цикл. Повторяющаяся смена внешности исполнителя для имитации движения персонажа. Использование бесконечного цикла для создания анимации.

Получение различного эффекта воспроизведения программы при изменении костюма исполнителя Scratch.

Параллелизм в программной среде

Использование нескольких исполнителей. Копирование программы одного исполнителя другим. Выполнение одинаковых программ разными исполнителями с использованием различных начальных условий. Параллельное выполнение однотипных действий. Принцип суперкомпьютерных технологий. Таймер для вычисления времени выполнения программы. Уменьшение показаний таймера при использовании параллельных вычислений.

Интерактивность программ. Возможность организации диалога между исполнителями. Операторы для слияния текстовых выражений.

Взаимодействие исполнителей путём касания друг друга или цвета. Использование сенсоров при взаимодействии исполнителей. Задержка выполнения программы.

Работа исполнителей в разных слоях изображения.

Ветвление в алгоритмах

Использование ветвления при написании программ. Короткая форма. Полная форма условного оператора. Конструкции ветвления для моделирования ситуации.

Цикл пока. Повторение команд исполнителя при выполнении определенного условия.

Последовательное выполнение фрагментов программы разными исполнителями

Типы исполнителей программной среды Scratch. Системы команд исполнителей. Различные системы команд для разных типов исполнителей.

Управление событиями. Передача сообщений исполнителям для выполнения определенной последовательности команд.

Передача управления между различными типами исполнителей.

Аналитическая деятельность:

- придумывать задачи для исполнителей программной среды;
- выделять ситуации, для описания которых можно использовать линейный алгоритм, алгоритм с ветвлениями, повторениями;
- определять эффективный способ решения поставленной задачи;
- находить параллельности в выполняемых действиях и программировать их с помощью нескольких исполнителей;
- планировать последовательность событий для заданного проекта.

Практическая деятельность:

- составлять и отлаживать программный код;
- использовать конструкции программной среды для создания линейных, разветвленных и циклических алгоритмов;
- организовывать параллельные вычисления;
- организовывать последовательность событий программы, передачу управления от одних исполнителей другим.

Раздел 4. Проектная деятельность и моделирование процессов и систем – 12 ч

Мультимедийный проект. Описание сюжетных событий. Анимация. Создание эффекта анимации с помощью последовательной смены изображений. Имитационные модели. Интерактивные проекты. Игры.

Аналитическая деятельность:

- создавать план появления событий для отражения определенной темы;
- выбирать иллюстративный материал из встроенной библиотеки;
- выбирать метод анимации для конкретной задачи;
- планировать последовательность событий для создания эффекта анимации по выбранному сценарию.

Практическая деятельность:

- использовать возможности программной среды Scratch для создания мультимедийных проектов;
- создавать имитационные модели, интерактивные проекты и игры средствами программной среды.

Ниже представлено рекомендуемое поурочное планирование модуля «Пропедевтика программирования со Scratch».

В результате освоения модуля «Пропедевтика программирования со Scratch» школьники получают представление о:

- свободно распространяемых программах;
- функциональном устройстве программной среды Scratch и основных структурных элементах пользовательского интерфейса;
 - назначении и использовании основных блоков команд, состояний, программ;
 - правилах сохранения документа и необходимости присвоения правильного имени;
 - возможности и способах отладки написанной программы;
 - сущности понятий «спрайт», «сцена», «скрипт»;
 - исполнителях и системах их команд, возможности непосредственного управления исполнителем;
 - наличии заготовок для персонажей и сцен в соответствующих библиотеках, иерархическом устройстве библиотек и возможности импортирования их элементов;
 - возможности использования встроенного растрового редактора, наличии и назначении основных инструментов;
 - использовании других программ (например, LibreOfficeDraw) для создания собственных изображений;
 - алгоритме как формальном описании последовательности действий исполнителя, приводящих от исходных данных к конечному результату;
 - использовании схематического описания алгоритма;
 - программном управлении исполнителем и линейных алгоритмах;
 - написании программ для исполнителей, создающих геометрические фигуры на экране в процессе своего перемещения;
 - необходимости программного прерывания;
 - использовании циклических команд при необходимости повторений однотипных действий;
 - видах циклических алгоритмов и их применении;
 - достижении эффекта перемещения путем использования циклов;

- возможности распараллеливания однотипных действий за счёт использования нескольких исполнителей;

- организации интерактивности программ;

- возможности взаимодействия исполнителей между собой, в различных слоях изображения;

- видах и формах разветвленных алгоритмов, включая циклы с условием;

- управлении событиями.

- использовании метода проектов для моделирования объектов и систем;

- возможности описания реальных задач средствами программной среды;

- создании анимационных, игровых, обучающих проектов, а также систем тестирования в программной среде Scratch.

Школьники будут уметь:

- самостоятельно устанавливать программную среду на домашний компьютер;

- изменять некоторые стандартные установки пользовательского интерфейса (например, язык отображения информации);

- использовать различные способы отладки программ, включая пошаговую отладку;

- уверенно использовать инструменты встроенного графического редактора, включая работу с фрагментами изображения и создание градиентов;

- создавать собственные изображения в других программах (например, LibreOfficeDraw) и импортировать их в программную среду Scratch;

- использовать графические примитивы векторного редактора LibreOfficeDraw для создания объектов;

- создавать изображения из пунктирных и штрих-пунктирных линий с изменением цвета и толщины линии;

- упрощать программы за счёт использования циклических команд и применять их;

- составлять простые параллельные алгоритмы;

- создавать программы и игры с использованием интерактивных технологий;

- моделировать ситуации с использованием необходимых форм ветвления алгоритма, включая цикл по условию;

- передавать сообщения исполнителям для выполнения последовательности команд (включая разные типы исполнителей).

- планировать и создавать анимации по определенному сюжету;

- создавать игры, используя интерактивные возможности программной среды Scratch;

- планировать и создавать обучающие программы для иллюстрации пройденного материала других предметных областей;

- продумывать и описывать интерактивное взаимодействие для создания простейших тренажеров;

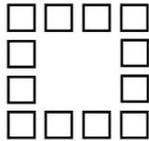
- подходить творчески к построению моделей различных объектов и систем.

Календарно-тематическое планирование

№ п/п	Тема	Элементы содержания	Виды деятельности	Формы контроля	Основные направления воспитания. Деятельности
Тема 1 «Знакомство с программной средой Scratch» (2ч.)					
1.	Знакомство со средой Scratch. Внешний вид среды, поля. Анимация.	<ul style="list-style-type: none"> • Техника программная среда Scratch, возможности установки; • основные элементы окна программной среды Scratch; • программа — создание и запуск. Начало выполнения программы при нажатии на зелёный флажок; • команды — использование и редактирование. Выполнение команд: блока управления (жёлтый блок) всегда, задержка выполнения команд. Команды движения (синий блок): перемещение на указанное количество шагов и столкновение с краем экрана; • персонажи — импорт и редактирование; 	<ul style="list-style-type: none"> • создание программы в среде Scratch; • рассмотрение примеров использования возможностей программной среды Scratch для создания анимированных моделей; • приобретение умения использовать встроенные библиотеки персонажей 	Кейс 1. Научи kota бегать и мяукать.	1, 3, 4, 5, 6, 7
2.	Исполнитель Scratch, цвет и размер пера.	<ul style="list-style-type: none"> • командный блок контроль (жёлтый) — команда начала выполнения программы при нажатой клавише пробел КОГДА КЛАВИША ПРОБЕЛ НАЖАТА и команда полного завершения программы ОСТАНОВИТЬ ВСЁ; • командный блок управления пером (зелёный) — команды: ОЧИСТИТЬ графические изображения, ОПУСТИТЬ ПЕРО для рисования по экрану, УСТАНОВИТЬ ЦВЕТ ПЕРА и УСТАНОВИТЬ РАЗМЕР ПЕРА,; 	<ul style="list-style-type: none"> • приобретение умения использовать цветовую палитру в среде Scratch; • применение умений изменять размер пера при написании программ для рисования в программной среде Scratch; • приобретение навыка грамотного написания программ, включая завершение программы; • приобретение навыка создания алгоритма для выполнения поставленной задачи. 	Кейс 2. Рисуем разноцветные лужи и облака для прогулки kota.	3,4,5
Тема 2 «Компьютерная графика» (5 ч.)					
3.	Основные инструменты встроенного растрового графического редактора.	<ul style="list-style-type: none"> • инструменты растрового графического редактора — кисточка, ластик, заливка, квадрат, круг, линия; • копирование, поворот, горизонтальное отражение, вертикальное отражение во встроенном редакторе программной среды Scratch; • командный блок внешность (фиолетовый) — команды начальной установки эффекта цвет УСТАНОВИТЬ ЭФФЕКТ ЦВЕТ В ЗНАЧЕНИЕ 0 и команда начальной установки 	<ul style="list-style-type: none"> • приобретение умений самостоятельно создавать графические изображения в среде Scratch, используя предоставленный инструментарий; • применение умений использовать эффекты при написании программ в среде Scratch; 	Кейс 3. Свободное рисование Придумай, чем можно дополнить проект с прошлого урока (Нарисовать дерево? Воздушный шар? Дом?) и реализуй это.	3,4,5

		<p>размера УСТАНОВИТЬ РАЗМЕР 100%;</p> <ul style="list-style-type: none"> • командный блок управления пером (зелёный) — команда ПЕЧАТЬ для копирования графического изображения исполнителя Scratch в нужном месте экрана; 	<ul style="list-style-type: none"> • приобретение навыка грамотного использования возможностей инструментов программной среды Scratch; • приобретение навыка устанавливать начальные значения для изменяющихся параметров; • приобретение навыка использовать геометрические примитивы для составления сложных геометрических форм 		
4.	<p>Линейный алгоритм. Создание блок-схемы. Основные графические примитивы векторного редактора LibreOfficeDraw.</p>	<ul style="list-style-type: none"> • алгоритм; • виды элементов блок-схемы алгоритма; • блок-схема линейного алгоритма; • графические примитивы векторного редактора LibreOffice.Draw; • параметры графических примитивов редактора векторной графики — заливка и обводка, цвет, толщина линии; • инструмент для создания надписей в векторном редакторе — текст; • выравнивание графических объектов по центру 	<ul style="list-style-type: none"> • приобретение умения использовать графические примитивы редактора векторной графики; • приобретение навыков создания векторных моделей в графическом редакторе LibreOffice.Draw; • приобретение умения реализовывать блок-схемы линейных алгоритмов; • приобретение навыков использования горячих клавиш при копировании объектов. 	<p>Кейс 4. Запиши мой алгоритм! Ребята разбиваются на пары, в паре формулируют друг другу какую-то повседневную задачу, для решения которой нужен линейный алгоритм, придумывают этот алгоритм и рисуют блок-схему для задачи, заданной им напарником.</p>	3,4,5
5.	<p>Линейный алгоритм. Рисование линий исполнителем Scratch.</p>	<ul style="list-style-type: none"> • последовательное выполнение команд; • линейный алгоритм; • блок-схема линейного алгоритма; • изменение параметров пера. 	<ul style="list-style-type: none"> • формирование навыка решения поставленной задачи в виде последовательного выполнения команд; • приобретение навыков создания блок-схемы линейного алгоритма средствами редактора векторной графики; • использование навыков изменения параметров команд программной среды Scratch для выполнения поставленной задачи; • применение созданной программы для широкого спектра возможных задач.. 	<p>Кейс 5. Что бывает полосатое? Придумайте линейный алгоритм для создания полосатого объекта (заранее придумайте, что это будет – шарф, зебра, забор?) Создайте блок-схему этого алгоритма и реализуйте алгоритм в среде Scratch.</p>	3,4,5

6.	<p>Линейный алгоритм. Исполнитель Scratch рисует квадраты и прямоугольники линейно.</p>	<ul style="list-style-type: none"> • команда «повернуть в направление»; • пошаговое выполнение программы для её отладки; • единичный шаг редактирования; центр костюма исполнителя Scratch. 	<ul style="list-style-type: none"> • формирование навыка создания алгоритма для рисования исполнителем квадрата путем последовательного выполнения команд; • приобретение навыка рисования линейного алгоритма, состоящего из двух колонок блоков команд; • использование навыков создания графических изображений в программной среде Scratch; • тренировка навыка использования векторного редактора офисного пакета LibreOffice в качестве инструмента для создания блок-схем; • формирование навыка выбора нужного значения из предлагаемого списка вариантов; • формирование навыка отладки программы для получения верного результата. 		3,4,5
7.	<p>Конечный цикл. Исполнитель Scratch рисует квадраты, линии.</p>	<ul style="list-style-type: none"> • команда открыть... из пункта меню File; • команда сохранить как... из пункта меню File; • циклический алгоритм; • блок-схема циклического алгоритма; • команда «повернуться на 90° по часовой стрелке»; • команда «повернуться на 90° против часовой стрелки»; • команда конечного цикла ПОВТОРИТЬ 10. 	<ul style="list-style-type: none"> • формирование навыка сохранения готовых программ для дальнейшего использования; • приобретение навыка рисования блок-схемы циклического алгоритма; • тренировка навыка использования векторного редактора офисного пакета LibreOffice в качестве инструмента для создания блок-схем; • тренировка навыка использования команд поворота на прямой угол (90°) по часовой и против часовой стрелки; • формирование навыка использования циклического алгоритма для рисования исполнителем квадрата; • формирование навыка 	<p>Упражнение. Сравнение двух алгоритмов, рисующих квадрат. Идти 100 шагов Повторить 4 Повернуться на 90° Идти 100 шагов Идти 100 шагов Повернуться на 90° Повернуться на 90° Идти 100 шагов Повернуться на 90° Идти 100 шагов Повернуться на 90° <i>Рекомендуется объяснить, что команды, обеспечивающие повторяющееся выполнение одной или нескольких команд называются циклическими, показать, что использование команды цикла всегда сокращает количество команд в программе, а значит делает её более красивой. В более</i></p>	3,4,5

			оптимизации линейного алгоритма за счёт использования циклической конструкции в программной среде Scratch;	<i>короткой программе легче обнаружить ошибку или неточность.</i>	
Тема 3 «Алгоритмы и исполнители» (14 ч.)					
8.	Конечный цикл. Исполнитель Scratch рисует несколько линий и фигур. Копирование фрагментов программы.	<ul style="list-style-type: none"> • тело цикла; • конечный цикл; • блок-схема бесконечного цикла; • имя костюма; • имя спрайта; • изменение костюма исполнителя; • копирование фрагмента программы 	<ul style="list-style-type: none"> • тренировка навыка использования операции цикла для решения учебных задач; • формирование пространственных представлений о двумерных фигурах; • тренировка навыка применения поворота на прямой угол (90°) при создании геометрических фигур и перемещении исполнителя; • тренировка навыка оптимизации линейного алгоритма за счёт использования циклической конструкции в программной среде Scratch; 	Кейс 6 Создай картинку из квадратов. Предложите ребятам нарисовать свою картинку из квадратов, может быть, это будет лестница, может быть, ковер, может быть, что-то еще.	3,4,5
9.	Циклический алгоритм. Цикл в цикле. Вложенные и внешние циклы.	<ul style="list-style-type: none"> • цикл в цикле; • блок-схема конструкции цикл в цикле; 	<ul style="list-style-type: none"> • тренировка навыка использования операции цикла в цикле для решения учебных задач; • формирование пространственных представлений о перемещении на плоскости; • тренировка навыка использования поворота на прямой угол (90°) по часовой и против часовой стрелки; • формирование навыка оптимизации алгоритма за счёт использования конструкции «цикл в цикле» в программной среде Scratch; 	Упражнение. Использование циклов «повторить _» Нарисуйте квадраты, как показано на рисунке, используя конструкцию цикл в цикле. 	3,4,5
10.	Цикл в цикле. Повторение пунктирной линии с поворотом. Блок-схема	<ul style="list-style-type: none"> • несплошные линии; • алгоритм рисования несплошных линий с использованием циклических конструкций; • рисование квадрата несплошными линиями, используя конструкцию «цикл в цикле»; 	<ul style="list-style-type: none"> • совершенствование навыка использования конструкции «цикл в цикле» при решении учебных задач; • формирование пространственных 	Кейс 7 Мой необычный дом. Предложите ребятам нарисовать дом и что-то рядом с ним, используя линейные и	3,4,5

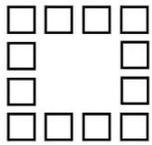
	цикла		<p>представлений о видах линий;</p> <ul style="list-style-type: none"> • формирование навыка создания и реализации алгоритма рисования квадрата несплошными линиями; • совершенствование навыка оптимизации алгоритма за счёт использования конструкции «цикл в цикле» в программной среде Scratch; • использование операции копирования внешности исполнителя путём копирования костюма. 	циклические алгоритмы, а также смену костюмов Исполнителя. Может быть, над домом будут звезды разного размера, а около дома будет пунктирная разноцветная дорога, или деревья разного размера.	
11.	Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов.	<ul style="list-style-type: none"> • бесконечный цикл; • анимация исполнителя с помощью смены костюмов; • эффект «призрак»; • изменение размера исполнителя; • интерактивное взаимодействие с исполнителем с помощью клавиатуры; • команды «спросить» и «думать» 	<ul style="list-style-type: none"> • тренировка навыка использования операции бесконечного цикла для создания анимации; • совершенствование навыка использования встроенной библиотеки персонажей 		3,4,5
12.	Сцена как исполнитель. Создаем модель таймера.	<ul style="list-style-type: none"> • анимация сцены (фона); • синхронная анимация сцены и исполнителя 	<ul style="list-style-type: none"> • совершенствование навыка использования операции бесконечного цикла для создания анимации; • совершенствование навыка оптимизации алгоритма за счёт использования циклических конструкций в программной среде Scratch; 		3,4,5
13.	Бесконечный цикл. Одна программа для исполнителя Scratch, но разные костюмы.	<ul style="list-style-type: none"> • анимация с помощью вращения; • дублирование исполнителей и их скрипты; • использование одинаковых программ, но разных костюмов у исполнителей. 	<ul style="list-style-type: none"> • совершенствование навыка использования операции бесконечного цикла для создания анимации; • совершенствование навыка использования встроенной библиотеки персонажей 		3,4,5
14.	Одинаковые программы для несколько исполнителей	<ul style="list-style-type: none"> • дублирование исполнителей; • синхронное выполнение. 	<ul style="list-style-type: none"> • формирование навыка создания и реализации алгоритма при параллельном выполнении скриптов результат работы • использование операции дублирования исполнителя; • совершенствование навыка 		3,4,5

			оптимизации алгоритма за счёт использования синхронного выполнения скриптов в программной среде Scratch;		
15.	Разбиение программы на части для параллельного выполнения исполнителями. Таймер.	<ul style="list-style-type: none"> • планирование действий разных исполнителей во времени для решения общей задачи; • сенсор «таймер» • сброс «таймера»; • команда «ждать до ...» 	<ul style="list-style-type: none"> • получение различного эффекта воспроизведения программы при изменении костюма исполнителя Scratch. • тренировка навыка использования команды «ждать до ...» 	Кейс 8 Мини-проект «Смена времени суток». Создайте скрипт, в котором будет четыре сцены: «утро», «день», «вечер», «ночь». Используйте «таймер» для своевременного изменения сцен, своевременного запуска скриптов разных исполнителей.	3,4,5
16.	Два исполнителя со своими программами. Мини-проект «Часы с кукушкой»..	<ul style="list-style-type: none"> • координаты в Scratch.; • команда «идти в x: _ y:_» (переместиться в точку с координатами (x,y)); • команда «плыть _ секунд в точку x: _ y:_» (плавно двигаться в точку с координатами (x,y)); • команды «спрятаться», «показаться». 	<ul style="list-style-type: none"> • тренировка навыка использования команд «идти в x: _ y:_», «плыть _ секунд в точку x: _ y:_», «спрятаться», «показаться»; • формирование навыка использования координат исполнителя 	Кейс 9 Мини-проект «Часы с кукушкой». Создайте сцену в виде часов, и три исполнителя: часовую стрелку, минутную стрелку и кукушку. Стрелки должны вращаться на циферблате. При наступлении каждого часа (т.е. в _ часов : 00 минут) должна появляться кукушка.	3,4,5
17.	Алгоритмы с ветвлением. Условие ЕСЛИ. Два исполнителя	<ul style="list-style-type: none"> • блок-схема алгоритма с ветвлением; • полная форма ветвления; • команда «если _ или»; • неполная форма ветвления; • команда «если _»; • сенсор «касается» (касания края, или другого спрайта, или указателя мыши) 	<ul style="list-style-type: none"> • тренировка навыка использования операции ветвления в алгоритмах; • тренировка навыка использования сенсора «касается» 		3,4,5
18.	Цикл с условием. Мини-проект «Шарики в лабиринте»	<ul style="list-style-type: none"> • сенсор «касается цвета»; • программируем отскок шариков от стен 	<ul style="list-style-type: none"> • совершенствование навыка использования сенсора «касается цвета» • тренировка навыка оптимизации алгоритма за счёт использования циклической конструкции с условием в программной среде Scratch 	Кейс 10 Мини-проект «Шарики в лабиринте» Создайте сцену в виде бильярдного стола, на котором три шарика будут двигаться с разными скоростями и отталкиваться от бортиков.	3,4,5
19.	Цикл с условием. Исполнитель определяет цвет. Сенсор	<ul style="list-style-type: none"> • сенсор «касается цвета»; • программируем поведение исполнителя в зависимости от цвета фона. 	<ul style="list-style-type: none"> • совершенствование навыка использования сенсора «касается цвета» • тренировка навыка 		3,4,5

	«касается цвета»		алгоритмизации поведения исполнителя в зависимости от цвета фона в программной среде Scratch;		
20.	Оператор случайных чисел.	<ul style="list-style-type: none"> команда «Выдать случайное число от _ до _»; случайные перемещения исполнителя в координатной плоскости; случайное количество шагов; случайные координаты; поворот на случайный угол. 	<ul style="list-style-type: none"> тренировка навыка использования команды «Выдать случайное число от _ до _» совершенствование навыка использования координат исполнителя; тренировка навыка использования поворота на случайный угол 		3,4,5
21.	Перемещение исполнителей между слоями.	<ul style="list-style-type: none"> команда «Перейти в верхний слой»; команда «Перейти назад на _ слоев»; эффект «Призрак» 	<ul style="list-style-type: none"> тренировка навыка использования команд «Перейти в верхний слой», «Перейти назад на _ слоев» тренировка навыка использования перемещения исполнителей между слоями; 		3,4,5

Тема 4 «Проектная деятельность» (13 ч.)

22.	Перемещение исполнителя из одного слоя в другой. Действия исполнителей в разных слоях. Мини-проект «Дорога».	<ul style="list-style-type: none"> команда «Перейти в верхний слой»; команда «Перейти назад на _ слоев»; эффект «Призрак» 	<ul style="list-style-type: none"> совершенствование навыка использования перемещения исполнителей между слоями; формирование навыка отладки программы для получения верного результата. 	Кейс 10 Мини-проект «Дорога». Создайте сцену, на которой изображена дорога с двумя полосами, по обочинам дороги расположите кусты и деревья. По дороге должны двигаться автомобили. Правильно расположите автомобили, деревья и кусты по слоям.	1, 3, 4, 5, 6, 7
23.	Алгоритмы с ветвлением. Условие ЕСЛИ. Взаимодействие исполнителей . Блок-схема с условием.	<ul style="list-style-type: none"> исполнители касаются друг друга; команда «Касается _»; поведение исполнителей при столкновении. 	<ul style="list-style-type: none"> совершенствование навыка использования сенсора «касается»; совершенствование навыка алгоритмизации поведения исполнителя при столкновении в программной среде Scratch; 		1, 3, 4, 5, 6, 7
24.	Сцена как исполнитель. Последовательное выполнение	<ul style="list-style-type: none"> команда «Передать»; команда «Когда я получу»; связи между программами разных исполнителей. 	<ul style="list-style-type: none"> тренировка навыка использования команд «Передать», «Когда я получу» совершенствование навыка отладки программы для 		1, 3, 4, 5, 6, 7

	команд исполнителя ми.		получения верного результата.		
25.	Алгоритмы с ветвлением. Программирование клавиш.	<ul style="list-style-type: none"> • взаимодействие пользователя с программой; • клавиши управления перемещением исполнителя; • команда «Изменить значение x на _»; • команда «Изменить значение y на _» 	<ul style="list-style-type: none"> • тренировка навыка использования команд для программирования клавиш; • тренировка навыка оптимизации алгоритма за счёт использования алгоритмической конструкции с условием в программной среде Scratch 	Кейс 10 Игра «Лабиринт» Создайте сцену в виде лабиринта. С помощью клавиш со стрелками управляйте движением маленькой мышки. Мышка должна добраться до выхода, не задевая стенок лабиринта. В случае касания стенок мышка возвращается в исходную точку.	1, 3, 4, 5, 6, 7
26.	Последовательное выполнение команд исполнителя ми.	<ul style="list-style-type: none"> • команда «Передать»; • команда «Когда я получу»; • связи между программами разных исполнителей 	<ul style="list-style-type: none"> • тренировка навыка использования команд «Передать», «Когда я получу»; • совершенствование навыка отладки программы для получения верного результата. 		1, 3, 4, 5, 6, 7
27.	Координатная плоскость. Геометрические фигуры.	<ul style="list-style-type: none"> • рисование геометрических фигур на координатной плоскости; • последовательное выполнение команд одним исполнителем; • параллельное выполнение команд несколькими исполнителями. 	<ul style="list-style-type: none"> • совершенствование навыка создания алгоритма для рисования исполнителем квадрата путем последовательного выполнения команд; • использование навыков создания графических изображений в программной среде Scratch 	Кейс 11. Исследование времени выполнения программ. Создайте одного исполнителя, рисующего квадраты, как показано на рисунке. Затем создайте четыре исполнителя, которые будут параллельно рисовать по три квадрата. Сравните время работы в этих двух случаях. 	1, 3, 4, 5, 6, 7
28.	Координатная плоскость. Переменные.	<ul style="list-style-type: none"> • блок «Переменные»; • рисование геометрических фигур на координатной плоскости с использованием переменных; • построение перпендикуляров к координатным осям. 	<ul style="list-style-type: none"> • тренировка навыка создания алгоритма для рисования исполнителем геометрических фигур на координатной плоскости с использованием переменных; • совершенствование навыка отладки программы для 		1, 3, 4, 5, 6, 7

			получения верного результата.		
29.	Создание списков.	<ul style="list-style-type: none"> • название списка; • элементы списка; • длина списка; • команда «Создать список»; • выбор элемента списка. 	<ul style="list-style-type: none"> • тренировка навыка создания алгоритма со списками; • тренировка навыка использования команды «Создать список». 	<p>Кейс 11. Мини-проект «Викторина».</p> <p>Создайте проект «Викторина» в котором будет задано 10 вопросов.</p> <p>Правильный ответ надо выбрать из списка. В случае правильного ответа очки увеличиваются на единицу. В конце выставляется оценка: «отлично» за 9 или 10, «хорошо» за 7 или 8, «удовлетворительно» за 5 или 6 правильных ответов.</p>	1, 3, 4, 5, 6, 7
30.	Использование подпрограмм.	<ul style="list-style-type: none"> • структурирование программ; • команда «Передать _ и ждать»; • команда «Играть звук _»; • команда «Ноту _ играть _ тактов» 	<ul style="list-style-type: none"> • тренировка навыка использования команд «Передать _ и ждать», «Играть звук _», «Ноту _ играть _ тактов»; • совершенствование навыка отладки программы для получения верного результата. 		1, 3, 4, 5, 6, 7
31.	Сообщество Scratch.	<ul style="list-style-type: none"> • Scratch 2.0; • регистрация на сайте; • личный кабинет; • публикация проектов 	<ul style="list-style-type: none"> • знакомство с сообществом Scratch; • формирование навыков создания личного информационного пространства 		1, 3, 4, 5, 6, 7
32.	Работа над индивидуальными и групповыми проектами	<ul style="list-style-type: none"> • основные требования к проекту; • критерии оценивания проекта. 	<ul style="list-style-type: none"> • работа, направленная на решение конкретной проблемы, на достижение оптимальным способом заранее запланированного результата; • формирование навыков использования возможностей программной среды Scratch для реализации проекта 	<p>Кейс 12. Итоговый проект. Учащиеся самостоятельно или в группах выполняют индивидуальный проект, согласованный с учителем. Как итог, конкурс проектов</p>	1, 3, 4, 5, 6, 7
33.					
34.	Представление индивидуальных и групповых работ	<ul style="list-style-type: none"> • защита индивидуальных и групповых проектов; • итоги конкурса 	<ul style="list-style-type: none"> • презентация проекта; • рассмотрение примеров проектов 	<p>Конкурс проектов</p>	1, 3, 4, 5, 6, 7

<i>Всего</i>	<i>34 ч.</i>
---------------------	---------------------