

Министерство образования Кировской области
Кировское областное государственное общеобразовательное автономное
учреждение «Гимназия г. Уржума»

Принята на заседании
методического совета
28.08.2022 г. Протокол № 1

Утверждена
приказом № 93-о
от 30.08.2022

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

Возраст обучающихся: 10 - 13 лет
Срок реализации: 1 год

Автор-составитель:
Елеева Ольга Викторовна,
учитель информатики

Уржум, 2022г

Содержание

1. Пояснительная записка	3
2. Цели и задачи программы.....	4
3. Планируемые результаты	5
4. Учебно-тематический план.....	9
5. Содержание программы.....	9
6. Условия реализации программы	14
7. Формы аттестации и оценочные материалы.....	15
8. Список литературы	22
9. Календарно-тематическое планирование.....	25

1. Пояснительная записка

Дополнительная общеразвивающая программа технической направленности «Программирование в Scratch» разработана в соответствии с Федеральным законом Российской Федерации «Об образовании в Российской Федерации» от 29 декабря 2012 г. № 273-ФЗ, Порядком организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам (приказ Минобрнауки РФ от 09.11.2018г. № 196) и отвечает требованиям «Концепции развития дополнительного образования» от 4 сентября 2014 года (Распоряжение Правительства РФ от 04.09.2014 N 1726-р), а также на основе методического инструментария тьютора «ИТ-Квантум. Тулкит», автор Владимир Войков.

Программа имеет **техническую направленность**.

Актуальность программы состоит в том, что мультимедийная среда Scratch позволяет сформировать у детей стойкий интерес к программированию, отвечает всем современным требованиям объектно-ориентированного программирования. Среда Scratch позволяет сформировать навыки программирования, раскрыть технологию программирования.

Новизна программы заключается в изменении подхода к обучению ребят, а именно –внедрению в образовательный процесс новых информационных технологий, побуждающих учащихся решать самые разнообразные логические и конструкторские проблемы.

Педагогическая целесообразность программы заключается в том, что в программу включён разнообразный познавательный и развивающий материал, что позволяет отойти от типовой программы школьной информатики и расширить знания учащихся в работе с компьютером, проводить групповое и индивидуальное сопровождение образовательного процесса.

Обучение основывается на педагогических принципах:

- личностно ориентированного подхода;
- природосообразности (учитывается возраст и уровень развития детей);
- культуросообразности (ориентация на общечеловеческие культурные ценности);
- систематичности, наглядности и последовательности обучения;
- сотрудничества и ответственности.

Отличительные особенности программы

Вариативность содержания программы: в зависимости от предпочтений учащегося в проектной деятельности возможен выбор образовательных модулей в специализированном блоке.

Таким образом, в программе реализуется возможность выбора и построения индивидуальной образовательной траектории для каждого учащегося.

Использование современного оборудования позволяет не только изучать базовые концепции, но и формировать образ мышления в контексте использования технологий будущего. Благодаря междисциплинарности проектной деятельности, учащиеся будут получать навыки работы в команде, распределения ролей при выполнении задания, требующего знаний и умений в различных областях науки и техники, а также навыки управления проектом.

Адресат программы.

Программа рассчитана на учащихся в возрасте от 10 до 13 лет.

Объём программы – Программа рассчитана на 34 учебных часа.

Форма обучения – очная, групповая (занятия в группах по 10-12 человек).

Срок освоения программы – 1 учебный год, 34 учебных недель

Режим занятий. Занятия проходят 1 раза в неделю по 1 часу.

2. Цели и задачи программы

Цель программы: обучение основам программирования, воспитание творческой личности, обогащенной общетехническими знаниями и умениями, развитие индивидуальных творческих способностей, интереса к науке и технике.

Задачи программы:

Обучающие:

- сформировать у детей базовые представления о языке программирования Scratch, алгоритме, исполнителе;
- сформировать навыки разработки, тестирования и отладки несложных программ;
- познакомить с понятием проекта, его структуры, дизайна и разработки;
- освоить навыки планирования, создания проекта, публикации его в сети Интернет;
- сформировать и развить навыки работы в сети для обмена материалами работы;
- •выработать навыки и умения безопасного и целесообразного поведения при работе с компьютерными программами и в сети Интернет, умения соблюдать нормы информационной этики и права.

Развивающие:

- на протяжении всех занятий формировать 4К-компетенции (критическое мышление, креативное мышление, коммуникация, кооперация);
- способствовать расширению словарного запаса;

- способствовать развитию памяти, внимания, технического мышления, изобретательности;
- способствовать развитию алгоритмического мышления;
- способствовать формированию интереса к техническим знаниям;
- способствовать формированию умения практического применения полученных знаний;
- сформировать умение формулировать, аргументировать и отстаивать своё мнение;
- сформировать умение выступать публично с докладами, презентациями и т. п.

Воспитательные:

- воспитывать аккуратность и дисциплинированность при выполнении работы;
- способствовать формированию положительной мотивации к трудовой деятельности;
- способствовать формированию опыта совместного и индивидуального творчества при выполнении командных заданий;
- воспитывать трудолюбие, уважение к труду;
- формировать чувство коллективизма и взаимопомощи;
- воспитывать чувство патриотизма, гражданственности, гордости за достижения отечественной ИТ-отрасли.

3. Планируемые результаты

Предметные результаты

Обучающийся научится (для использования в повседневной жизни и обеспечения возможности успешного продолжения образования на базовом уровне)

в области информационных технологий:

- запускать на выполнение программу Scratch, работать с ней, сохранять созданные файлы, закрывать программу;
- создавать, переименовывать, перемещать, копировать и удалять файлы;
- работать с основными элементами пользовательского интерфейса: использовать меню, обращаться за справкой, работать с окнами (изменять размеры и перемещать окна, реагировать на диалоговые окна);
- вводить информацию в компьютер с помощью клавиатуры и мыши;
- применять встроенный в программу Scratch графический редактор для создания и редактирования простых рисунков;
- осуществлять поиск информации в сети Интернет с использованием простых запросов (по одному признаку);

- ориентироваться на интернет-сайтах (нажать указатель, вернуться, перейти на главную страницу);
- соблюдать требования к организации компьютерного рабочего места, требования безопасности и гигиены при работе со средствами ИКТ;
- разрабатывать и реализовывать собственные творческие проекты в среде Scratch, размещать их на своей странице сайта <http://scratch.mit.edu>, просматривать чужие проекты на данном сайте, оценивать их и скачивать для использования с учётом авторских прав;
- сформировать начальные представления о назначении и области применения проектов; о проектировании как методе научного познания.

в области алгоритмов и элементов программирования:

- понимать смысл понятия «скрипт - алгоритм», приводить примеры алгоритмов;
- понимать термины «спрайт - исполнитель», «среда исполнителя», «блоки скриптов - система команд исполнителя»;
- осуществлять управление имеющимся формальным исполнителем; понимать правила записи и выполнения алгоритмов, содержащих алгоритмические конструкции «следование», «ветвление», «цикл»;
- подбирать алгоритмическую конструкцию, соответствующую заданной ситуации;
- исполнять линейный, разветвляющийся и циклический алгоритмы для формального исполнителя с заданной системой команд.

Личностные результаты:

- критическое отношение к информации и избирательность её восприятия;
- осмысление мотивов своих действий при выполнении заданий;
- развитие любознательности, сообразительности при выполнении разнообразных заданий проблемного и эвристического характера;
- развитие внимательности, настойчивости, целеустремлённости, умения преодолевать трудности;
- развитие самостоятельности суждений, независимости и нестандартности мышления;
- освоение социальных норм, правил поведения, ролей и форм социальной жизни в группах и сообществах;
- формирование коммуникативной компетентности в общении и сотрудничестве с другими обучающимися.

Метапредметные результаты:

Регулятивные универсальные учебные действия:

- умение принимать и сохранять учебную задачу;
- умение планировать последовательность шагов алгоритма для достижения цели;
- умение ставить цель (создание творческой работы), планировать достижение этой цели;
- умение осуществлять итоговый и пошаговый контроль по результату;
- способность адекватно воспринимать оценку наставника и других обучающихся;
- умение различать способ и результат действия;
- умение вносить коррективы в действия в случае расхождения результата решения задачи на основе её оценки и учёта характера сделанных ошибок;
- умение в сотрудничестве ставить новые учебные задачи;
- способность проявлять познавательную инициативу в учебном сотрудничестве;
- умение осваивать способы решения проблем творческого характера в жизненных ситуациях;
- умение оценивать получающийся творческий продукт и соотносить его с изначальным замыслом, выполнять по необходимости коррекции либо продукта, либо замысла.

Познавательные универсальные учебные действия:

- умение осуществлять поиск информации в индивидуальных информационных архивах обучающегося, информационной среде образовательного учреждения, федеральных хранилищах информационных образовательных ресурсов;
- умение использовать средства информационных и коммуникационных технологий для решения коммуникативных, познавательных и творческих задач;
- умение ориентироваться в разнообразии способов решения задач;
- умение осуществлять анализ объектов с выделением существенных и несущественных признаков;
- умение проводить сравнение, классификацию по заданным критериям;
- умение строить логические рассуждения в форме связи простых суждений об объекте;
- умение устанавливать аналогии, причинно-следственные связи;
- умение моделировать, преобразовывать объект из чувственной формы в модель, где выделены существенные характеристики объекта (пространственно-графическая или знаково-символическая);
- умение синтезировать, составлять целое из частей, в том числе самостоятельно достраивать с восполнением недостающих компонентов.

Коммуникативные универсальные учебные действия:

- умение аргументировать свою точку зрения на выбор оснований и критериев при выделении признаков, сравнении и классификации объектов;
- умение выслушивать собеседника и вести диалог;
- способность признавать возможность существования различных точек зрения и право каждого иметь свою;
- умение планировать учебное сотрудничество с наставником и другими обучающимися: определять цели, функции участников, способы взаимодействия;
- умение осуществлять постановку вопросов: инициативное сотрудничество в поиске и сборе информации;
- умение разрешать конфликты: выявление, идентификация проблемы, поиск и оценка альтернативных способов разрешения конфликта, принятие решения и его реализация;
- умение с достаточной полнотой и точностью выражать свои мысли в соответствии с задачами и условиями коммуникации;
- владение монологической и диалогической формами речи.

Предметные результаты

Обучающийся научится (для использования в повседневной жизни и обеспечения возможности успешного продолжения образования на базовом уровне)

в области информационных технологий:

- запускать на выполнение программу Scratch, работать с ней, сохранять созданные файлы, закрывать программу;
- создавать, переименовывать, перемещать, копировать и удалять файлы;
- работать с основными элементами пользовательского интерфейса: использовать меню, обращаться за справкой, работать с окнами (изменять размеры и перемещать окна, реагировать на диалоговые окна);
- вводить информацию в компьютер с помощью клавиатуры и мыши;
- применять встроенный в программу Scratch графический редактор для создания и редактирования простых рисунков;
- осуществлять поиск информации в сети Интернет с использованием простых запросов (по одному признаку);
- ориентироваться на интернет-сайтах (нажать указатель, вернуться, перейти на главную страницу);
- соблюдать требования к организации компьютерного рабочего места, требования безопасности и гигиены при работе со средствами ИКТ;

- разрабатывать и реализовывать собственные творческие проекты в среде Scratch, размещать их на своей странице сайта <http://scratch.mit.edu>, просматривать чужие проекты на данном сайте, оценивать их и скачивать для использования с учётом авторских прав;
- сформировать начальные представления о назначении и области применения проектов; о проектировании как методе научного познания.

в области алгоритмов и элементов программирования:

- понимать смысл понятия «скрипт - алгоритм», приводить примеры алгоритмов;
- понимать термины «спрайт - исполнитель», «среда исполнителя», «блоки скриптов - система команд исполнителя»;
- осуществлять управление имеющимся формальным исполнителем; понимать правила записи и выполнения алгоритмов, содержащих алгоритмические конструкции «следование», «ветвление», «цикл»;
- подбирать алгоритмическую конструкцию, соответствующую заданной ситуации;
- исполнять линейный, разветвляющийся и циклический алгоритмы для формального исполнителя с заданной системой команд.

4. Учебно-тематический план

№ п/п	Наименование разделов	Количество часов			Форма контроля и аттестации
		Всего	Теория	Практика	
1.	Знакомство с программной средой Scratch	4	3	1	Беседа, анкетирование, практическая работа
2.	Компьютерная графика	7	1,5	5,5	
3.	Алгоритмы и исполнители	23	6,5	16,5	Практическая работа, проект, выставки, мастер-классы
	Итого	34	11	23	

5. Содержание программы

Знакомство с программной средой Scratch – 4 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.

Компьютерная графика – 7 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 и импортирование их в программную среду Scratch.

Знакомство с основными графическими примитивами векторного редактора. 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.

Алгоритмы и исполнители – 23 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.

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.

6. Условия реализации программы

Методическое обеспечение программы

Выбирая форму ведения занятий, учитываются индивидуальные и возрастные особенности обучающихся, их потенциальные возможности. Занятия состоят из теоретических и практических частей, которые могут идти параллельно (на усмотрение педагога), чтобы сразу же закреплять теоретические знания на практике. Практическая часть занимает большую часть учебного времени.

Практически каждая тема занятия начинается с постановки задачи — характеристики образовательного продукта, который предстоит создать учащимся. Приветствуется творческое свободное начало, инициатива со стороны детей, умение применять на практике полученные знания.

На протяжении курса обучения по программе происходит постепенное усложнение материала.

Для развития творческого подхода педагог создает проблемные ситуации, с целью поиска детьми различных вариантов решения проблемы, таким образом, используется метод проектной деятельности. Наряду с этим педагог развивает навыки самоконтроля и взаимоконтроля. Одной из его форм является презентация проектов.

Занятия предусматриваются следующие формы организации учебной деятельности:

- индивидуальная (учащемуся дается на его выбор самостоятельное творческое задание с учетом его интересов);
- фронтальная (работа со всеми одновременно);
- групповая (выполнение групповых проектов).

Для успешной реализации программы предусмотрены следующие **методы и средства**:

в обучении:

- словесные (рассказ, беседа, чтение; прослушивание аудиороликов);

- наглядные (показ иллюстраций, просмотр документальных и учебных фильмов; работа с учебной и специальной литературой, дидактическим материалом);
- практические (работа в компьютерных программах);
- игровые (применением познавательных игр (викторины, конкурсы, олимпиады и др.));
- информационные (интерактивное обучение) – электронные образовательные ресурсы, презентаций, компьютерные программы.

в воспитании:

- методы формирования сознания личности, направленные на формирование устойчивых убеждений (рассказ, дискуссия, этическая беседа, пример);
- методы организации деятельности и формирования опыта общественного поведения (воспитывающая ситуация, приучение,
- упражнения, тренинги);
- методы стимулирования поведения и деятельности (соревнования, поощрения, конкурсы).

А также используются в процессе обучения следующие методы (классификация по типу познавательной деятельности): практический, репродуктивный, алгоритмический, метод проблемного изложения.

Кадровое обеспечение

Реализовать программу может педагог, имеющий средне общее или высшее образование, обладающий достаточными теоретическими знаниями и практическими умениями в области ИКТ.

Материально-технические условия реализации программы

Для работы с учебным комплектом необходимо:

- - программное обеспечение: Scratch, Paint;
- - аппаратное обеспечение: компьютерный класс, интерактивная система.

7. Формы аттестации и оценочные материалы

Формы подведения итогов реализации общеобразовательной программы

Подведение итогов реализуется в рамках защиты результатов выполнения заданий по каждой теме.

Формы демонстрации результатов обучения

Представление результатов образовательной деятельности пройдет в форме публичной презентации решений задач и последующих ответов выступающих на вопросы наставника и других команд.

Формы диагностики результатов обучения

Для определения достижения обучающимися планируемых результатов можно применять следующие методики: наблюдение, беседа, опрос, анкетирование, задания на выделение существенных признаков, задания на логические закономерности, задания проблемно-поискового характера, задания на внимание, методики самооценки, создание проектов и другие.

Оценка и требования к содержанию проекта. Проект должен отражать уровень теоретического осмысления предложенных в рамках данной программы тем, а также практические умения, которыми слушатели овладели в процессе обучения. **Критерии оценки проекта и процедура его защиты.** Критерии оценки проектов слушателей:

- адекватность формулировки темы, обоснование актуальности, целей и задач проекта;
- чёткое понимание сущности понятий, терминов, научных подходов, идей, которые лежат в основе разработки заявленной темы;
- реализация теоретических знаний на практике;
- наличие в работе количественных и качественных показателей успешного внедрения полученных знаний;
- качество оформления проекта;
- оригинальность, практическая значимость.

Процедура защиты проекта. Защита проекта проводится на заключительном занятии. Продолжительность выступления разработчика проекта – 5-7 минут.

Материалы контрольных опросов:

Основы работы со средой программирования Scratch

1. Основные элементы пользовательского интерфейса программной среды Scratch.
2. Блочная структура систематизации информации.
3. Блоки команд, состояний, программ, запуска, действий и исполнителей.
4. Основные понятия: спрайт, сцена, скрипт, проект, костюм.
5. Редактирование и изменение размеров спрайтов.
6. Библиотека сцен. Замена сцены.
7. Система команд исполнителя Scratch.
8. Основные приемы программирования в Scratch
9. Создание линейной программы для нескольких спрайтов

Графический редактор Scratch

1. Знакомство с командами рисования. 2. Установка цвета и размера пера
3. Дублирование объектов.
4. Основные признаки линейного алгоритма.
5. Программное управление исполнителем.
6. Создание программ для перемещения исполнителя по экранному полю.
7. Точка отсчета, оси координат, единица измерения расстояния, абсцисса и ордината.
8. Написание программы для исполнителя, чтобы он оставлял пунктирную линию при перемещении по экранному полю.

Смена костюма и фона

1. Управление курсор движения.
2. Импорт костюма, импорт фона.
3. Смена фона, задание координат для траектории движения спрайта.
4. Библиотека персонажей.
5. Сцена и разнообразие сцен, исходя из библиотеки данных. Систематизация данных библиотек персонажей и сцен.
6. Иерархия в организации хранения костюмов персонажа и фонов для сцен.
7. Создание и редактирование фона и костюмов с помощью встроенного графического редактора.

Понятие цикла. Соблюдение условий

1. Знакомство с понятием «цикл» и умение применять это понятие при описании скрипта.
2. Многократное повторение команд как организация цикла.
3. Особенности использования цикла в программе.
4. Вставка и редактирование звука; подборка музыкального сопровождения в соответствии с темой проекта.
5. Знать, как пишется скрипт с использованием блока «звук».
6. Понятие условия. Формулировка условий. Операции сравнения.
7. Простые и составлены условия. Алгоритмическая конструкция ветвления. Команды ветвления Если..., Если...Иначе....
8. Выполнение скриптов с ветвлениями. Вложенные команды ветвления
9. Перемещение спрайта за указателем мыши. Превращение спрайта в указатель мыши.

10. Задание перемещения спрайта с помощью клавиш на клавиатуре для нескольких объектов.
11. Добавление, перезапуск и останов таймера.
12. Разработка программы с одновременным выполнением двух или более скриптов одним объектом.
13. Знакомство с командами: изменить ... эффект на ... убрать графические эффекты.

Самоуправление

1. Понятие сообщения. Передача сообщения, запуск скриптов при условии получения сообщения вызова.
2. Обмен данными между скриптами.
3. Знакомство с командами: ждать передать ... когда я получу ... спрятаться показаться изменить размер на ... установить размер ... %
4. Запуск спрайтов с помощью мыши и клавиатуры.
5. Путешествия объекта по нескольким сценам, команды показаться и спрятаться.

Создание переменных. Списки

1. Создание переменной.
2. Осуществление вычисления при помощи переменных.
3. Применение переменной при написании скрипта.
4. Знакомство с понятием «случайное число».
5. Передвижения объекта в случайное место с использованием команды выдать случайное число от ... до
6. Создание скрипта для очистки холста.
7. Изменение отображения переменной.
8. Список как упорядоченный набор однотипной информации.
9. Понятие списка.
10. Создание списков.
11. Добавление и удаление элементов.
12. Поиск необходимых данных в списке.
13. Вычисление итоговых показателей для списка.

Строковые константы и переменные. Операции со строками

1. Создание списка.
2. Использование операций сравнения.
3. Анализ и вывод строковых переменных на экран.
4. Разработка и создание теста с использованием заранее подготовленных

материалов.

5. Создание сложных условий.
6. Тестирование и отладка проекта.
7. Этапы решения задачи (постановка, алгоритмизация, кодирование, тестирование, отладка).
8. Понятие проект, его структура и реализация в среде Scratch.
9. Работа со звуком.
10. Основные этапы разработки проекта.
11. Дизайн проекта.
12. Разработка и создание небольшой программы с использованием заранее подготовленных материалов.

Материалы для практических работ:

Графический редактор Scratch

1. Запустить программу Scratch.
2. Нарисовать произвольный спрайт с помощью инструмента Кисть. Изменить размер спрайта, скопировать его. Сохранить в своей папке.
3. Отработать команды блока Перо. Установить цвет и размер пера.
4. Создать программу для перемещения исполнителя по заданной траектории.
5. Сохранить полученные результаты.
6. Загрузить фон и исполнителя из сетевой папки.
7. Написать программу для исполнителя, чтобы он оставлял пунктирную линию при перемещении по экранному полю.
8. Создать проект «Морское путешествие».
9. Сохранить проект в своей папке.

Смена костюма и фона

1. Импортировать фон и спрайт из сетевой папки.
2. Задать координаты для траектории движения спрайта.
3. Написать программу полета самолета со сменой фона.
4. Создать и отредактировать фон и костюмы с помощью встроенного графического редактора.
5. Написать программу «Подводный мир».

Понятие цикла. Соблюдение условий

1. Создать программу с определенным количеством повторений команд.
2. Конструкция Всегда. Создать программу с бесконечным повторением

команд.

3. Добавить музыку из библиотеки. Проект «Танцы, танцы...»
6. Создать программу с простыми командами ветвлениями. Проект «Космос».
7. Разработать алгоритм и создать программу перемещения спрайта за указателем мыши и в указатель мыши.
8. Создать программу перемещения спрайта с помощью клавиш на клавиатуре для нескольких объектов. Добавить таймер, перезапустить его и остановить при достижении цели.
9. Разработать программу с одновременным выполнением двух или более скриптов одним объектом. Использовать команды изменить ... эффект на ..., убрать графические эффекты. Создать сценарий путешествия, в котором герой, управляемый пользователем, может переходить от фона к фону.

Самоуправление спрайтов

1. Разработать алгоритм для обмена данными между спрайтами.
2. Разработать программу с последовательным выполнением одного скрипта разными объектами. Познакомиться с командами: ждать..., передать ..., когда я получу ... , спрятаться, показаться.
3. Запустить спрайты с помощью мыши и клавиатуры. Создать проект «Переодевалки».
4. Создать проекта с возможностью взаимодействия между объектами, принадлежащими разным средам с использованием команд: мышка нажата?
мышка по x мышка по y всегда, если ...
5. Создать двухуровневый проект «Лабиринт», изменяя скорость перемещения спрайта и используя несколько условий в программе.

Создание переменных. Списки

1. Создать переменную для осуществления вычислений. Написать программу с использованием переменной.
2. Создать проект передвижения объекта в случайное место с использованием команды выдать случайное число от ... до Создать проект передвижения нескольких объектов с использованием датчика случайных чисел.
3. Создать программы для изменения размера спрайта в зависимости от количества объектов. Создать скрипт для очистки холста. Изменить отображение переменной.
4. Создание список команд исполнителя. Разработать проект «Управление роботом».

Строковые константы и переменные. Операции со строками

1. Использовать операций сравнения в строковых переменных. Проанализировать и вывести строковые переменные на экран.
2. Создать проект «Угадай слово».
3. Разработать тесты с использованием заранее подготовленных материалов. Использовать сложные условия. Протестировать на нескольких примерах и отладить проект.
4. Разработать проект (постановка задачи, алгоритмизация, кодирование, тестирование, отладка), его структуру и реализацию в среде Scratch. Подобрать звук (скачать из интернета). Протестировать и отладить.

Материалы для проведения проектных работ

1. Запустить программу Scratch.
2. Загрузить фон из библиотеки, из файла, нарисовать в графическом редакторе.
3. Загрузить спрайт из библиотеки, из файла, нарисовать.
4. Изменить размеры спрайта, сделать несколько копий.
5. Написать несложную программу движения для одного спрайта.
6. Для остальных сдублировать эту программу и отредактировать (изменить направление движения спрайта и его скорость).
7. Сохранить программу.
8. Нарисовать или выбрать готовый фон «Природа».
9. Добавить спрайты «Лесные жители».
10. Составить программу перемещения объектов с учетом изменения размеров, скорости, направления движения.
11. Вставить звук к каждому объекту и применить конструкцию Если.
12. Добавить несколько сцен. Менять сцены, используя команду Передать... и Когда я получу...
13. Сохранить работу в своей папке.
14. Из Интернета скачать фон для проекта «Галактика».
15. Скачать спрайты и сохранить их в своей папке.
16. Создать проект, в котором используются команды Диалога между спрайтами, с учетом конструкции Если. Создать непрерывный цикл.
17. Создать проект, используя команду Когда спрайт нажат... для нескольких объектов. К каждому объекту вставить свой звук (например, проект «Оркестр»).
18. Создать проект «Пульт управления» для 5 программ ТВ.

19. Создать программу усложненного лабиринта с учетом перемещающихся преград.
20. Составить программу для управления объектом, используя списки.
21. Создать тест по русскому языку.

Тест «Основные понятия Scratch»

1. Как называется подвижный графический объект, который действует на сцене проекта и выполняет разнообразные алгоритмы (сценарии). Исполнитель алгоритмов, которому доступны все команды языка Scratch.
 - а) Скрипт
 - б) Спрайт
 - в) Сцена
 - г) Котенок
2. Блоки команд в программе Scratch разделены на разноцветные категории. Сколько таких категорий?
 - а) 20
 - б) 15
 - в) 10
 - г) 7
3. Как называется алгоритм (или сценарий), составленный из блоков языка Scratch для какого-нибудь объекта?
 - а) Скрипт
 - б) Спрайт
 - в) Сцена
 - г) Код
4. Чему равна ширина сцены?
 - а) 320 точек
 - б) 480 точек
 - в) 260 точек
 - г) Может меняться
5. Сколько костюмов может иметь спрайт?
 - а) 1
 - б) 2
 - в) Любое количество
 - г) Можно не более 7
6. Чему равна высота сцены?
 - а) 320 точек
 - б) 480 точек
 - в) 360 точек
 - г) Может меняться
7. Как называется место, где спрайты двигаются, рисуют и взаимодействуют?
 - а) Скрипт
 - б) Спрайт
 - в) Сцена
 - г) Котенок
8. Можно ли сделать проект, в котором нет сцены?
 - а) Да
 - б) Нет
 - в) Иногда можно

9. Какое расширение имеют файлы, созданные в среде Scratch?
- а) .sb2
 - б) .exe
 - в) .psd
 - г) .bmp
10. Набор команд, которые может выполнять объект, называют ...
- а) СКИ
 - б) Алгоритм
 - в) Скрипт
 - г) Программа

Ответы на тест:

- 1. б
- 2. в
- 3. а
- 4. б
- 5. в
- 6. в
- 7. в
- 8. б
- 9. а
- 10. а

8. Список литературы

Список литературы для педагога

- 1. Модуль «Пропедевтика программирования со Scratch», Сорокина Т.Е;
- 2. Рындак В. Г., Дженжер В. О., Денисова Л. В. Проектная деятельность школьника в среде программирования Scratch. — Оренбург: Оренб. гос. ин-т. менеджмента, 2009.
- 3. «Пропедевтика идей параллельного программирования в средней школе при помощи среды Scratch», В.Г. Рындак, В.О. Джинжер, Л.В. Денисова;
- 4. «Раннее обучение программированию в среде Scratch», В.Г. Рындак, В.О. Джинжер, Л.В. Денисова;
- 5. Патаракин Е.П. Учимся готовить в среде Scratch - Версия 2.0

Список литературы для обучающихся

- 1. Творческие задания в среде Scratch: рабочая тетрадь для 5-6 классов/ Ю.В. Пашковская. – М.: БИНОМ. Лаборатория знаний, 2014. – 200 с.: ил.
- 2. Д.В. Голиков и А.Д. Голиков, «Программирование на Scratch 2. Делаем игры и мультики». Изд. Электронное издание 2014.
- 3. Д.В. Голиков и А.Д. Голиков, «Программирование на Scratch 2. Делаем сложные игры». Изд. Электронное издание 2014.
- 4. Ю.В. Торгашева, «Первая книга юного программиста. Учимся писать программы на Scratch». Изд. Питер 2016.

5. Патаракин Е.П. Учимся готовить в среде Scratch - Версия 2.0

Интернет ресурсы:

1. <http://scratch.mit.edu> – официальный сайт Scratch
2. <http://letopisi.ru/index.php> /Скретч - Скретч в Летописи.ру
3. <http://setilab.ru/scratch/category/commun> - Учитесь со Scratch

9. Календарно-тематическое планирование

№ п.п.	Тема	Количество часов				Форма контроля
		Всего	Теория	Практика	Контроль	
Знакомство с программной средой Scratch – 4 ч.						
1	Вводное занятие. Исполнители, роботы и робототехника, программы для исполнителей. Техника безопасности и правила поведения.	1	1			Беседа, анкетирование
2	Понятия "команды", "программа" и "программирование". Среда конструирования. Языки программирования.	1	1			Беседа
3	Знакомство со средой Scratch. Внешний вид среды, поля. Анимация.	1	0,5	0,5		Практическая работа
4	Исполнитель Scratch, цвет и размер пера.	1	0,5	0,5		Практическая работа
Компьютерная графика – 7 ч.						
5	Основные инструменты встроенного растрового графического редактора.	1	0,5	0,5		Практическая работа
6-7	Линейный алгоритм. Создание блок-схемы. Основные графические примитивы векторного редактора.	2	1	1		Практическая работа
8	Линейный алгоритм. Рисование линий исполнителем Scratch.	1		1		Практическая работа
9	Линейный алгоритм. Исполнитель Scratch рисует квадраты и прямоугольники линейно.	1		1		Практическая работа
10-11	Конечный цикл. Исполнитель Scratch рисует квадраты, линии.	2		1	1	Практическая работа
Алгоритмы и исполнители – 23 ч.						
12	Конечный цикл. Исполнитель Scratch рисует несколько линий и фигур. Копирование фрагментов программы.	1	0,5	0,5		Практическая работа
13	Циклический алгоритм. Цикл в цикле. Вложенные и внешние циклы.	1		1		Практическая работа
14	Цикл в цикле. Повторение пунктирной линии с поворотом. Блок-схема цикла.	1	0,5	0,5		Практическая работа

15-16	Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов.	2	1	1		Практическая работа
17-18	Сцена как исполнитель. Создаем модель таймера.	2	1	1		Практическая работа
19	Бесконечный цикл. Одна программа для исполнителя Scratch, но разные костюмы.	1		1		Практическая работа
20-21	Одинаковые программы для несколько исполнителей.	2	1	1		Практическая работа
22-23	Несколько исполнителей. Параллельное выполнение действий для ускорения процесса выполнения программы.	2		2		Практическая работа
24-25	Разбиение программы на части для параллельного выполнения исполнителями. Таймер. Уменьшение показаний таймера при параллельных вычислениях.	2	1	1		Практическая работа
26-27	Два исполнителя со своими программами. Мини-проект «Часы».	2		1	1	Проект
28	Алгоритмы с ветвлением. Условие ЕСЛИ. Два исполнителя.	1		1		Практическая работа
29	Цикл при условии. Мини-проект «Шарики в лабиринте»	1		1		Проект
30	Цикл при условии. Исполнитель определяет цвета.	1	0,5	0,5		Практическая работа
31	Цикл при условии. Исполнители в разных слоях. Мини-проект «Самолет сквозь облака».	1		1		Проект
32-33	Выставки, мастер-классы, экскурсии	2			2	Выставки, мастер-классы
34	Итоговое занятие	1	1			Рефлексия
ИТОГО		34	11	19	4	