

Муниципальное общеобразовательное учреждение
Большенагаткинская средняя школа им. Героя Советского Союза В.А. Любавина
муниципального образования «Цильнинский район» Ульяновской области

Согласовано:

на заседании педагогического совета

Протокол от 27.08.2021

Утверждаю:

Директор МОУ Большенагаткинская СШ

С.В. Ратаев



**Дополнительная общеобразовательная общеразвивающая программа
технической направленности
«Программирование на Scratch»
педагога Романова Алексея Николаевича**

**Возраст обучающихся: 12-15 лет
Срок реализации программы: 1 год
Уровень программы: базовый**

с. Большое Нагаткино
2021

Пояснительная записка

Нормативно-правовое обеспечение программы.

В настоящее время содержание, роль, назначение и условия реализации программ дополнительного образования закреплены в следующих нормативных документах:

- Федеральный Закон Российской Федерации от 29.12.2012 № 273 «Об образовании в Российской Федерации» (далее – ФЗ № 273);
- Приказ Министерства просвещения РФ от 9 ноября 2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- Концепция развития дополнительного образования детей от 04.09.2014 № 1726;
- Письмо Минобрнауки России от 18.11.2015 №09-3242. «Методические рекомендации по проектированию дополнительных общеразвивающих программ»;
- Письмо Минобрнауки России от 28 апреля 2017 г. № ВК – 1232/09 «О направлении методических рекомендаций» вместе с (Методическими рекомендациями по организации независимой оценки качества дополнительного образования детей»);
- СанПин СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- СанПин 2.4.3172-14: «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей».
- Устав, локальные акты МОУ Большенагаткинской средней школы.

Наука и технология развиваются столь стремительно, что образование зачастую не успевает за ними. Одним из самых известных способов организации дополнительной деятельности является метод проектов. Самым подходящим инструментом для организации подобной образовательной деятельности является среда для программирования Scratch. Овладев даже минимальным набором операций, юный пользователь сможет с лёгкостью создавать проекты.

Scratch- это среда, которая позволяет детям создавать собственные анимации, мини-мультфильмы и игры. Работа в среде scratch развивает у детей логику, позволяет сформировать понятие об основах программирования.

Scratch можно рассматривать как инструмент для творчества. Обучающиеся учатся сочинять и реализовывать истории, рисовать и оживлять придуманных ими персонажей, учатся работать с графикой и звуком.

В результате выполнения простых команд может складываться сложная модель, в которой будут взаимодействовать множество объектов, наделенных различными свойствами.

Работая в среде Scratch обучающиеся осваивают множество навыков, таких как: творческое мышление, системный анализ, проектирование, умение обучаться и самообучаться, самостоятельное принятие решений.

Направленность программы. Техническая.

Дополнительность программы. Данный вид деятельности в общеобразовательных учреждениях не изучается.

Актуальность программы. Мультимедийная среда Scratch позволяет сформировать у обучающихся интерес к программированию, отвечает всем современным требованиям объектно-ориентированного программирования. Данная среда позволяет сформировать навыки и освоить технологию программирования. Изучение данного языка значительно облегчает последующий переход к изучению других языков программирования.

Новизна программы заключается в комбинировании исследовательской деятельности с изучением основ программирования и создания проекта в программной среде Scratch. Аспект новизны заключается в том, что Scratch не просто язык программирования, а еще и интерактивная среда, где результаты действий визуализированы, что делает работу с программой понятной, интересной и увлекательной. Особенность среды Scratch, позволяющая создавать мультфильмы, анимацию и даже простейшие игры, делает программу практически значимой для современного школьника. Это дает возможность увидеть практическое назначение алгоритмов и программ, что будет способствовать развитию интереса к профессиям, связанным с программированием.

Педагогическая целесообразность программы: заключается в создании организационных и психолого-педагогических условий для привлечения детей и подростков к занятиям техническим творчеством, обеспечивающих развитие мотивации к познанию, творчеству и труду, конструкторских и изобретательских способностей, формирование инженерно-технических компетенций, как факторов успешного самоопределения и самореализации личности в современном мире.

Отличительная особенность. Особенность среды Scratch- программа позволяет создавать мультфильмы, анимацию, игры. Делает образовательную программу практически значимой для учащегося, так как дает возможность увидеть практическое назначение алгоритмов и программ, что способствует развитию интереса к профессиям, связанным с программированием.

Инновационность программы: Новизна заключается в том, что Scratch не просто язык программирования, а еще и интерактивная среда, где результаты действий визуализированы, что делает работу с программой понятной, интересной и увлекательной.

Адресат программы: Дети в возрасте 12-15 лет, деятельность обучающихся становится предметной. Теперь, кроме игры проступает второе важное направление — учеба и развитие. Этот период связан с активной работой психики. Предстоит многому научиться.

Объём программы: 70 часов.
Модульная программа:
Модуль 1: 35 часов: 11.09.2021-28.12.2021
Модуль 2: 35 часов: 11.01.2022-23.05.2022

График занятий: время занятия составляет 1 ч., занятия проходят 2 раз в неделю по вторникам и средам с 14.30 до 15.30.

Формы обучения: Очная. Постоянное общение с преподавателем, для обмена информацией, возможность заниматься научной деятельностью во время обучения.

Особенности организации образовательного процесса:

В соответствии с учебным планом программы объединении «Программируем на «Scratch»». Состав группы постоянный.

В период дистанционного обучения – организация работы в дистанционном формате, онлайн-уроки, видеопрезентации.

Уровень реализуемой программы: Базовый. Предполагает использование и реализацию общедоступных и универсальных форм организации материала, минимальную сложность предлагаемого для освоения содержания программы.

Цель и задачи программы.

Цель программы: создание условий для развития алгоритмических и креативных способностей обучающихся к творческому самовыражению в проектной деятельности в области программирования, через формирование ключевых компетенций, основанных на создании ценностно-ориентированного, конструктивного стиля мышления и новых способах самостоятельной творческой деятельности по направлению ИТ.

Задачи программы:

Личностные

- формирование ответственного отношения к учению, способности довести до конца начатое дело на примере завершённых творческих учебных проектов;
- формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой благодаря иллюстративной среде программирования мотивации к обучению и познанию;
- развитие опыта участия в социально значимых проектах, повышение уровня самооценки, благодаря реализованным проектам;
- формирование коммуникативной компетенции в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;
- формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;
- формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;
- развитие эстетического сознания через творческую деятельность на базе иллюстрированной среды программирования.

Метапредметные

- умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата, понимая, что в программировании длинная программа не значит лучшая программа;
- умение оценивать правильность решения учебно-исследовательской задачи;
- умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;
- владение основами самоконтроля, принятия решений;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;
- ИКТ-компетенция;
- умение сотрудничества и совместной деятельности со сверстниками в процессе проектной и учебно-исследовательской деятельности.

Предметные

- 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.

Планируемые результаты.

Личностные.

По окончании освоения программы учащийся научится:

- самостоятельно и творчески решать поставленные задачи;
- активно участвовать в коллективной работе.

Метапредметные.

Учащийся научится использовать полученные теоретические знания и практические навыки самостоятельной работы на компьютере

Предметные.

По окончании освоения программы учащийся освоит:

- способы записи алгоритма;
- среду программирования;

- система команд исполнителя Scratch;
- линейный алгоритм, цикл, ветвления, их реализация в среде Scratch;
- понятие переменной;
- понятие проекта, его структура и реализация в среде Scratch.

**В процессе обучения дети:
будут знать:**

правила безопасной работы с компьютером;
основные компоненты среды программирования Scratch;
компьютерную среду, включающую в себя графический язык программирования;

будут уметь:

самостоятельно решать задачи в процессе написания программы (планирование предстоящих действий, самоконтроль, применять полученные знания, приемы и опыт с использованием специальных элементов и т.д.);

создавать простые и сложные проекты по собственному замыслу и по поставленной задаче.

Корректировать программу при необходимости;

у них будут развиты:

пространственное воображение, логическое и визуальное мышление;
познавательные, интеллектуальные и творческие способности;

будут обладать следующими качествами:

самостоятельное мышление, умение отстаивать своё мнение;

потребность в самообразовании, дальнейшем развитии профессиональных умений и навыков в области программирования;

Содержание учебного плана

МОДУЛЬ 1.

Раздел 1. Знакомство с программной средой Scratch – 4ч

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 – 10 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.) и импортирование их в программную среду Scratch.

Знакомство с основными графическими примитивами векторного редактора (LibreOfficeDraw) и импортирование их в программную среду Scratch.. 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 – 21 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) и импортирование их в программную среду Scratch..

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.

МОДУЛЬ 2.

Проектная деятельность и моделирование процессов и систем 35 ч.

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.

Учебный план

№ п/п	Тема	Количество часов	теория	практика	Формы аттестации
	МОДУЛЬ 1.				
	Раздел 1. Знакомство с программной средой Scratch	4ч			
1.	Знакомство со средой Scratch. Внешний вид среды, поля. Анимация.	2	1	1	Устный опрос
2.	Исполнитель Scratch, цвет и размер пера.	2	1	1	беседа
	Раздел 2. Компьютерная графика	10			
3.	Основные инструменты встроенного растрового графического редактора.	2	1	1	Практическое задание
4.	Линейный алгоритм. Создание блок-схемы. Основные графические примитивы векторного редактора (LibreOfficeDraw) и импортирование их в программную среду Scratch..	2	1	1	Практическое задание, беседа
5.	Линейный алгоритм. Рисование линий исполнителем	2	1	1	Практическое задание, беседа
6.	Линейный алгоритм. Исполнитель Scratch рисует квадраты и прямоугольники линейно.	2			

7.	Конечный цикл. Исполнитель Scratch рисует квадраты, линии	2	1	1	
Раздел 3. Алгоритмы и исполнители		21			
8.	Конечный цикл. Исполнитель Scratch рисует несколько линий и фигур. Копирование фрагментов программы.	2	1	1	Практическое задание, беседа
9.	Циклический алгоритм. Цикл в цикле. Вложенные и внешние циклы.	2	1	1	Практическое задание, беседа
10.	Цикл в цикле. Повторение пунктирной линии с поворотом. Блок-схема цикла.	2	1	1	Практическое задание, беседа
11.	Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов	2	1	1	Практическое задание, беседа
12.	Сцена как исполнитель. Создаем модель таймера.	2	1	1	Практическое задание, беседа
13.	Бесконечный цикл. Одна программа для исполнителя Scratch, но разные костюмы	2	1	1	Практическое задание, беседа
14.	Одинаковые программы для несколько исполнителей.	2	1	1	Практическое задание, беседа
15.	Несколько исполнителей. Параллельное выполнение действий для ускорения процесса выполнения программы.	2	1	1	Практическое задание, беседа
16.	Разбиение программы на части для параллельного выполнения исполнителями. Таймер. Уменьшение показаний таймера при параллельных вычислениях.	2	1	1	Практическое задание, беседа
17.	Два исполнителя со своими программами. Мини-проект «Часы».	2	1	1	Практическое задание, беседа
18.	Алгоритмы с ветвлением. Условие ЕСЛИ. Два исполнителя.	2	1	1	Практическое задание, беседа
МОДУЛЬ 2.					
Раздел 4. Проектная деятельность и моделирование процессов и систем		35			
19.	Цикл при условии. Мини-проект «Шарики в лабиринте»	2	1	1	Практическое задание, беседа
20.	Цикл при условии. Исполнитель определяет цвета.	2	1	1	Практическое задание, беседа
21.	Цикл при условии. Исполнители в разных слоях. Мини-проект «Самолет сквозь облака». Действия исполнителей в разных слоях. Мини-проект «Дорога».	2	1	1	Практическое задание, беседа

23.	Алгоритмы с ветвлением. Условие ЕСЛИ. Взаимодействие исполнителей. Блок-схема с условием	2	1	1	Практическое задание, беседа
24.	Сцена как исполнитель. Последовательное выполнение команд исполнителями	2	1	1	Практическое задание, беседа
25.	Алгоритмы с ветвлением. Программирование клавиш.	2	1	1	Практическое задание, беседа
26.	Алгоритмы с ветвлением. Если касается цвета.	2	1	1	Практическое задание, беседа
27.	Интерактивность исполнителей. Создание мини-проекта «Лабиринт».	2	1	1	Практическое задание, беседа
28.	Игра «Лабиринт». Усложнение.	2	1	1	Практическое задание, беседа
29.	Моделирование ситуации. Мини-проект «Пешеходный переход».	2	1	1	Практическое задание, беседа
30.	Моделирование ситуации. Интерактивность исполнителей. Мини-проект «Водолей».	2	1	1	Практическое задание, беседа
31.	Моделирование. Учебные модели «Рисующий карандаш», «Затухание».	2	1	1	Практическое задание, беседа
32.	Моделирование. Тестовая модель «Комнатные растения».	2	1	1	Практическое задание, беседа
33.	Моделирование. Обучающий проект по маршрутам географических открытий.		1	1	Практическое задание, беседа
34.	Подготовка и презентация проектов	2	1	1	Практическое задание
35.	Подготовка и презентация проектов	2	1	1	Практическое задание

Календарный учебный график

№п/п	Тема	Количество часов		
	МОДУЛЬ 1		Форма занятия	Форма контроля
1.	Знакомство со средой Scratch.	2	Лекция	Практическое задание

	Внешний вид среды, поля. Анимация.			
2.	Исполнитель Scratch, цвет и размер пера.	2	Лекция	Практическое задание
3.	Основные инструменты встроенного растрового графического редактора.	2	Лекция	Практическое задание
4.	Линейный алгоритм. Создание блок-схемы. Основные графические примитивы векторного редактора (LibreOfficeDraw) и импортирование их в программную среду Scratch..	2	Лекция	Практическое задание
5.	Линейный алгоритм. Рисование линий исполнителем Scratch.	2	Лекция	Практическое задание
6.	Линейный алгоритм. Исполнитель Scratch рисует квадраты и прямоугольники линейно.	2	Лекция	Практическое задание
7.	Конечный цикл. Исполнитель Scratch рисует квадраты, линии.	2	Лекция	Практическое задание
8.	Конечный цикл. Исполнитель Scratch рисует несколько линий и фигур. Копирование фрагментов программы.	2	Лекция	Практическое задание
9.	Циклический алгоритм. Цикл в цикле. Вложенные и внешние циклы.	2	Лекция	Практическое задание
10.	Цикл в цикле. Повторение пунктирной линии с поворотом. Блок-схема цикла.	2	Лекция	Практическое задание
11.	Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов.	2	Лекция	Практическое задание
12.	Сцена как исполнитель. Создаем модель таймера.	2	Лекция	Практическое задание
13.	Бесконечный цикл. Одна программа	2	Лекция	Практическое задание

	для исполнителя Scratch, но разные костюмы.			
14.	Одинаковые программы для несколько исполнителей.	2	Лекция	Практическое задание
15.	Несколько исполнителей. Параллельное выполнение действий для ускорения процесса выполнения программы.	2	Лекция	Практическое задание
16.	Разбиение программы на части для параллельного выполнения исполнителями. Таймер. Уменьшение показаний таймера при параллельных вычислениях.	2	Лекция	Практическое задание
17.	Два исполнителя со своими программами. Мини-проект «Часы».	2	Лекция	Практическое задание
18.	Алгоритмы с ветвлением. Условие ЕСЛИ. Два исполнителя. МОДУЛЬ 2.	2	Лекция	Практическое задание
19.	Цикл при условии. Мини-проект «Шарики в лабиринте»	2	Лекция	Практическое задание
20.	Цикл при условии. Исполнитель определяет цвета.	2	Лекция	Практическое задание
21.	Цикл при условии. Исполнители в разных слоях. Мини-проект «Самолет сквозь облака».	2	Лекция	Практическое задание
22.	Перемещение исполнителя из одного слоя в другой. Действия исполнителей в разных слоях. Мини-проект «Дорога».	2	Лекция	Практическое задание
23.	Алгоритмы с ветвлением. Условие	2	Лекция	Практическое задание

	ЕСЛИ. Взаимодействие исполнителей.			
	Блок-схема с условием.			
24.	Сцена как исполнитель.	2	Лекция	Практическое задание
	Последовательное выполнение команд исполнителями.			
25.	Алгоритмы с ветвлением.	2	Лекция	Практическое задание
	Программирование клавиш.			
26.	Алгоритмы с ветвлением. Если касается цвета.	2	Лекция	Практическое задание
27.	Интерактивность исполнителей. Создание мини-проекта «Лабиринт».	2	Лекция	Практическое задание
28.	Игра «Лабиринт». Усложнение.	2	Лекция	Практическое задание
29.	Моделирование ситуации. Мини-проект «Пешеходный переход».	2	Лекция	Практическое задание
30.	Моделирование ситуации. Интерактивность исполнителей. Мини-проект «Водолей».	2	Лекция	Практическое задание
31.	Моделирование. Учебные модели «Рисующий карандаш», «Затухание».	2	Лекция	Практическое задание
32.	Моделирование. Тестовая модель «Комнатные растения».	2	Лекция	Практическое задание
33.	Моделирование. Обучающий проект по маршрутам географических открытий.	2	Лекция	Практическое задание
34.	Подготовка и презентация проектов	2	Лекция	Практическое задание
35.	Подготовка и презентация проектов	2	Лекция	Практическое задание

2. Условия реализации программы Материально-техническое обеспечение

Кружок располагается в Точке Роста. Кабинет обеспечен соответствующей мебелью: рабочими столами, стульями, ноутбуками, программным обеспечением, выходом в интернет, мультимедийной доской, столом для руководителя.

Группа обучающихся состоит из 8-10 человек. Дети работают индивидуально.

Рабочее место оснащено столом, стульями, ноутбуком, компьютерной мышью.

К работе в кружке дети приступают после проведения руководителем соответствующего инструктажа по правилам техники безопасной работы каким-либо инструментом или приспособлением.

Информационное обеспечение

Персональный компьютер со специальной программой SCRATCH 3 и выходом в сеть Интернет.

Кадровое обеспечение

В реализации программы занят один педагог Романов А.Н (учитель информатики).. Образование: педагогическое, Ульяновский Государственный Педагогический Университет им. И.Н. Ульянова 2003.

Формы проведения занятий:

- занятие с использованием игровых технологий;
- занятие-игра;
- занятие-исследование;
- творческие практикумы (сбор скриптов с нуля);
- занятие-испытание игры;
- занятие-презентация проектов;
- занятие с использованием тренинговых технологий (работа на редактирование готового скрипта в соответствии с поставленной задачей).

Формы организации учебного занятия:

Форма организации деятельности групповая, при этом отдельные вопросы и ошибки рассматриваются в индивидуальном порядке с каждым обучающимся, исходя из особенностей каждого обучающегося в усвоении пройденного материала.

Первая часть занятия предполагает получение обучающимся нового материала. Во время второй части занятия обучающийся пытается

самостоятельно реализовать полученную теоретическую базу в рамках собственного проекта. Оценка результатов производится коллективно всей группой.

Некоторые занятия полностью отведены на реализацию проектной работы.

Общение на занятии ведётся в свободной форме — каждый обучающийся в любой момент может задать интересующий его вопрос без поднятия руки. Данный момент очень важен в процессе обучения, так как любой невыясненный вопрос, может превратиться в препятствие для получения обучающимся последующих знаний и реализации им собственных проектов.

Формы аттестации

Процесс обучения по дополнительной общеразвивающей программе предусматривает следующие формы диагностики и контроля:

1. Первичная диагностика, проводится в начале года
2. Промежуточная диагностика, проводится по завершению полугодия
3. Итоговая диагностика проводится после завершения всей учебной программы.

Оперативный контроль учебных достижений осуществляется на протяжении всех занятий и имеет своей целью оценку систематичности учебной работы обучающихся по формированию знаний и умений в рамках освоения данного материала. Проводится в процессе устного опроса, проведения практических работ, выполнения индивидуальных заданий и т.п.

Задачи текущего контроля:

- повышение мотивации обучающихся к регулярной учебной работе;
- развитие навыков самостоятельной работы;
- обеспечение обратной связи между обучающимися и преподавателем, на основании которой устанавливается, как обучающиеся воспринимают и усваивают учебный материал;
- дифференциация итоговой оценки знаний.

Для оценки планируемых результатов применяется **текущий и итоговый** контроль.

Текущий контроль – опросы , просмотры работ.

Итоговый контроль – презентация творческих работ учащихся.

По мере получения теоретических знаний учащиеся будут реализовывать их на практике.

1. Промежуточная аттестация – тестирование и просмотр работ.

2. Итоговая аттестация – просмотр, защита проектов.

В качестве способов проверки результатов в процессе обучения применяются тестирование по изучаемым темам, конкурсы между учащимися на скорость и качество решения поставленной задачи. Результаты практической деятельности обучающихся оцениваются педагогом. При оценке учитывается правильность, оптимальность, скорость решения задачи и уровень самостоятельности при решении задачи.

Предметом диагностики и контроля являются внешние образовательные продукты обучающихся (созданные проекты), а также их внутренние личностные качества

и компетенции (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам программы.

Основой для оценивания деятельности обучающихся являются результаты анализа проектов и деятельности по их созданию. Оценка имеет различные способы выражения — устные суждения педагога, письменные качественные характеристики, систематизированные по заданным параметрам аналитические данные, в том числе и рейтинги. Одна из задач педагога - обучение детей навыкам самооценки. С этой целью педагог выделяет и поясняет критерии оценки, учит детей формулировать эти критерии в зависимости от поставленных целей и особенностей образовательного продукта — создаваемого мультимедийного проекта.

Проверка достигаемых образовательных результатов производится в следующих формах:

- 1) текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий;
- 2) оценка учащимися работ друг друга или работ, выполненных в группах;
- 3) публичная защита выполненных учащимися творческих работ (индивидуальных и групповых);
- 4) текущая диагностика и оценка педагогом деятельности учащихся;
- 5) итоговая оценка деятельности по образовательной программе в форме защиты портфолио в рамках итоговой конференции;
- 6) итоговая оценка индивидуальной деятельности учащегося педагогом, выполняемая в форме образовательной характеристики;
- 7) независимая экспертная оценка творческих работ (работы) обучающегося в рамках конкурсов, олимпиад, конференций различного ранга.

Список литературы для учащихся

1. Денис Голиков «Scratch для юных программистов».
2. Интернет-источник [<https://scratch.mit.edu>]
4. Могилев А.В. и др. Информатика. – М., 1999. – 816 с.
5. Гурин Н.И. Работа на персональном компьютере. М., 1994.
6. Поспелов Д.А. Информатика: Энциклопедический словарь для начинающих. – М.: Педагогика-Пресс, 1994. – 352 с.

Литература и интернет источники для педагога

1. Scratch по-русски. <http://scratchrus.w>) и импортрование их в программную среду Scratch. ordpress.com/
2. Scratch. Идея, программа, общество. <http://scratch.mit.edu/>
3. Дженжер В.О., Денисова Л.В. Пропедевтика идей параллельного программирования в средней школе при помощи среды Scratch. <http://e-learning.uvk6.info/skretc/propedevtikaidej-parallelnogo-programmirovaniya-v-srednej-skole-pri-pomosi-sredy-scratch>
4. Скретч. Материал из Letopisi.ru. <http://letopisi.ru/index.php/Scratch>
5. Что нам стоит Scratch построить? <http://scratch.ucoz.net/publ/5>

Литература для родителей

1. Голиков Д. В. Scratch для юных программистов. — СПб.: БХВ-Петербург, 2017. — 192 с.: ил.
2. Шпынева С. М. Методическое пособие Технологии Scratch. – Тамбов, 2014. – 29с: ил.
3. <http://scratch.mit.edu/pages/source> – страница разработчиков.
4. <http://scratch.mit.edu/> - официальный сайт проекта Scratch.
5. <http://setilab.ru/scratch/category/commun/> Сайт «Учитесь со Scratch»
http://minecraftnavideo.ru/play/vd20J2r5wUQ/scratch_lesson_01_znakomstvo_so_sr_edoj_programmirovaniya_scratch.html

