

МУНИЦИПАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ИРКУТСКОГО РАЙОННОГО МУНИЦИПАЛЬНОГО ОБРАЗОВАНИЯ
«Ревякинская средняя общеобразовательная школа»

«СОГЛАСОВАНО»

«29» августа 2024г.

Заведующая центром

«Точка роста»

Степанова А.Г.

«УТВЕРЖДАЮ»

Приказ №86/21

от «29» августа 2024 г

Директор МОУ ИРМО

«Ревякинская СОШ»

Шилинскас Г.В.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
детского объединения

«Среда программирования Scratch»

Направленность: техническая

Уровень: базовый

Возраст обучающихся: 9 – 11 лет

2024г

Содержание

1. Пояснительная записка
 - 1.1. Информационные материалы и литература
 - 1.2. Направленность программы
 - 1.3. Актуальность и педагогическая целесообразность программы
 - 1.4. Отличительные особенности программы
 - 1.5. Адресат программы
 - 1.6. Срок освоения программы
 - 1.7. Форма обучения
 - 1.8. Форма организации обучения
 - 1.9. Режим занятий
 - 1.10. Условия реализации программы
 - 1.11. Цель и задачи программы
 - 1.12. Планируемые результаты
 - 1.13. Формы, методы контроля результативности обучения
 - 1.14. Формы подведения итогов.
2. Учебно-тематический план.
 - 2.1. Содержание программы.
3. Методическое обеспечение.
 - 3.1. Методические виды продукции.
 - 3.2. Дидактический материал.
 - 3.3. Описание методов обучения
 - 3.4. Применяемые технологические технологии
 - 3.5. Формы проведения занятий.
4. Список литературы

1. Пояснительная записка

1.1. Информационные материалы и литература:

1. Федеральный Закон «Об образовании в Российской Федерации» от 29.12.2012 №273-ФЗ.
2. Концепция развития дополнительного образования детей (утверждена распоряжением Правительства РФ от 04.09.2014 №1726-р).
3. Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам (утвержден приказом Министерства образования и науки РФ от 9.11.2018 № 196).
4. Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей (утверждено постановлением главного государственного санитарного врача РФ от 04.07.2014 № 41).
5. Общих требований к определению нормативных затрат на оказание государственных (муниципальных) услуг в сфере образования, науки и молодежной политики, применяемых при расчете объема субсидии на финансовое обеспечение выполнения государственного (муниципального) задания на оказание государственных (муниципальных) услуг (выполнения работ) государственным (муниципальным) учреждением (утверждены приказом Министерства образования и науки РФ от 22.09.2015 № 1040).
6. Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы) (Приложение к письму Департамента государственной политики в сфере воспитания детей и молодежи Министерства образования и науки РФ от 18.11.2015 № 09-3242).

1.2. Направленность: научно-технические технологии

1.3. Актуальность и педагогическая целесообразность программы

Мультимедийная среда Scratch позволяет сформировать у детей интерес к программированию, отвечает всем современным требованиям объектно-ориентированного программирования. Среда Scratch позволяет формировать навыки программирования, раскрыть технологию программирования. Изучение языка значительно облегчает последующий переход к изучению других языков программирования. Преимуществом Scratch, среди подобных сред программирования, является наличие версий для различных операционных систем, к тому же программа является свободно распространяемой, что немало важно для образовательных учреждений России. Именно в настоящее время имеет смысл рассматривать программы с открытым кодом, что позволяет сформировать у учащихся более широкое представление о возможностях работы с цифровой техникой.

1.4. Отличительные особенности программы

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

1.5. Адресат программы

Программа рассчитана на учащихся 9 – 11 лет и предполагает, что учащиеся владеют навыками работы с клавиатурой, мышью, приемами работы с графическими

изображениями, программа не требует первоначальных знаний в области программирования.

1.6. Срок освоения программы

Программа рассчитана на один год. Количество учебных часов по программе: 162ч.

1.7. Форма обучения

Форма обучения – очная.

1.8. Форма организации обучения

Занятия будут проводиться на базе центра образования цифрового и гуманитарного профилей «Точка роста».

1.10. Условия реализации программы.

Изучение данного курса позволит детям получить общее представление об основах программирования, овладеть навыками программирования в среде Scratch.

1.11. Цель и задачи программы

1.11.1. Цель: содействие развитию социально активной, творческой, успешной личности в условиях современного общества посредством применения знаний и умений основ программирования в среде программирования Scratch.

1.11.2. Задачи программы

- Овладение базовыми понятиями объектно-ориентированного программирования и применение их при создании проектов в визуальной среде программирования Scratch;
- Приобщение обучающихся к новым технологиям, способным помочь им в реализации собственного творческого потенциала;
- Развитие познавательной деятельности учащихся в области новых информационных технологий;
- Способствовать развитию интереса к профессиям, связанным с программированием, анимацией, мультипликацией;
- Совершенствование навыков работы на компьютере и повышение интереса к программированию.

1.12. Планируемые результаты

После изучения данной программы, обучающиеся должны приобрести следующие знания, умения и навыки:

Знать:

- функциональное устройство программной среды Scratch и основных структурных элементах пользовательского интерфейса;
- назначение и использование основных блоков команд, состояний, программ;
- правила сохранения документа и необходимости присвоения правильного имени;
- возможности и способы отладки написанной программы;
- сущности понятий «спрайт», «сцена», «скрипт»;
- исполнители и системы их команд, возможности непосредственного управления исполнителем;
- наличие заготовок для персонажей и сцен в соответствующих библиотеках, иерархическом устройстве библиотек и возможности импортирования их элементов;
- возможности использования встроенного растрового редактора, наличия и назначения основных инструментов;
- использование других программ (например, LibreOfficeDraw) для создания собственных изображений;
- алгоритм как формальный язык описания последовательности действий исполнителя, приводящих от исходных данных к конечному результату;

- использование схематического описания алгоритма;
- программное управление исполнителем и линейных алгоритмах;
- написание программ для исполнителей, создающих геометрические фигуры на экране в процессе своего перемещения;
- необходимость программного прерывания;
- использование циклических команд при необходимости повторений однотипных действий;
- виды циклических алгоритмов и их применение;
- достижения 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;
- виды и формы разветвленных алгоритмов, включая циклы с условием;
- управление событиями.
- использование метода проектов для моделирования объектов и систем;
- возможности описания реальных задач средствами программной среды;
- создание анимационных, игровых, обучающих проектов, а также систем тестирования в программной среде Scratch.

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.

1.13. Формы, методы контроля результативности обучения.

Уровень обученности учащихся выявляется по трем параметрам и определяется как минимальный, общий, продвинутый.

1. Теоретические знания

Критериями оценки являются: степень усвоения теоретического материала, глубина, широта и системность теоретических знаний, грамотное использование компьютерных терминов.

2. Знание технологии

Критериями оценки являются: степень усвоения материала, глубина, широта и системность знания технологии.

3. Уровень овладения практическими умениями и навыками

Критериями являются: разнообразие умений и навыков, грамотность (соответствие существующим нормативам и правилам, технологиям) практических действий, свобода владения специальным компьютерным оборудованием и программным обеспечением, качество творческих проектов учащихся: грамотность исполнения, использование творческих элементов.

Текущий контроль усвоения материала планируется осуществлять путем устного и письменного опроса, в виде различных тестов, в том числе в электронном виде, самостоятельных, практических и творческих работ; путем использования игровой формы проведения контроля знаний в виде ребусов, кроссвордов, конкурсов.

Итоговый контроль – в виде конкурсов, защиты и представления творческих работ.

1.14. Форма подведения итогов

Итог – творческая работа на свободную тему, выполненная в изученной среде программирования Scratch. Учащиеся самостоятельно или в парах выполняют индивидуальный проект, согласованный с учителем. Как итог, можно провести конкурс проектов.

При оценке творческих работ педагог должен руководствоваться следующим критериям:

- общая художественная выразительность;
- самостоятельность и оригинальность замысла;
- содержательность сюжета;
- проявление наблюдательности, воображения;
- использование циклических алгоритмов;
- управление одновременной работой нескольких исполнителей;
- внедрение звуковых эффектов в алгоритмы исполнителей;
- создание алгоритмов, которые будут выполняться одновременно (параллельно) несколькими исполнителями.

2. Учебно-тематический план.

№ п/п	Наименование разделов и тем	Количество аудиторных часов		
		Всего	в том числе	
			Теоретически занятия	Практические занятия
1	Знакомство с программной средой Scratch	4	4	0
1.1	Знакомство со средой Скретч. Создание и редактирование спрайтов и фонов для сцены.	2		2
1.2	Знакомство со средой Скретч. Пользуемся помощью Интернета. Поиск, импорт и редакция спрайтов и фонов из Интернета.	2		2
2	Компьютерная графика	12	4	24
2.1	Управление спрайтами: команды идти, повернуться на угол, опустить перо, поднять перо, очистить.	2		2
2.2	Координатная плоскость. Точка отсчёта, оси координат, единица измерения расстояния, абсцисса и ордината.	2	1	1
2.3	Навигация в среде Скретч. Определение координат спрайта.	4		4
2.4	Создание проекта «Кругосветное путешествие Магеллана».	4		4
3	Алгоритмы и исполнители	40	4	24
3.1	Понятие цикла. Команда повторить . Рисование узоров и орнаментов.	3	1	2
3.2	Конструкция всегда . Создание проектов «Берегись автомобиля!» и «Гонки по вертикали». Команда если край, оттолкнуться .	4		4
3.3	Ориентация по компасу. Управление курсом движения. Команда повернуть в направление . Проект «Полёт самолёта».	2		2
3.4	Спрайты меняют костюмы. Анимация. Создание проектов «Осьминог», «Девочка, прыгающая на скакалке» и «Бегущий человек».	4		4
3.5	Создание мультипликационного сюжета «Кот и птичка».	2		2
3.6	Соблюдение условий. Сенсоры. Блок если . Управляемый стрелками спрайт.	3	1	2
3.7	Создание коллекции игр: «Лабиринт», «Кружащийся котёнок».	3		3
3.8	Пополнение коллекции игр: «Опасный лабиринт».	3		3
3.9	Составные условия. Проекты «Хождение по коридору», «Слепой кот», «Тренажёр памяти».	4		4
3.10	Датчик случайных чисел. Проекты «Разноцветный экран», «Хаотичное движение», «Кошки-мышки», «Вырастим цветник».	6		6

3.11	Циклы с условием. Проект «Будильник».	2		2
3.12	Запуск спрайтов с помощью мыши и клавиатуры. Проекты «Переодевалки» и «Дюймовочка».	4		4
4	Проектная деятельность и моделирование процессов и систем	106		
4.1	Самоуправление спрайтов. Обмен сигналами. Блоки передать сообщение и Когда я получу сообщение . Проекты «Лампа» и «Диалог».	6		6
4.2	Доработка проектов «Магеллан», «Лабиринт».	4		4
4.3	Датчики. Проекты «Котёнок-обжора»	6		6
4.4	Переменные. Их создание. Использование счётчиков. Анимация. Разворачиваем Пчелу в направление движения.	7		7
4.5	Ввод переменных. Проект «Цветы». Доработка проекта «Лабиринт» - запоминание имени лучшего игрока.	5		5
4.6	Ввод переменных с помощью рычажка. Проекты «Цветы» (вариант-2), «Правильные многоугольники».	5		5
4.7	Список как упорядоченный набор однотипной информации. Создание списков. Добавление и удаление элементов. Проекты «Гадание», «Назойливый собеседник».	5		5
4.8	Поиграем со словами. Строковые константы и переменные. Операции со строками.	6		6
4.9	Организация интерактивного диалога с пользователем	5		5
4.10	Мини-проект «Шарики в лабиринте»	3		3
4.11	Мини-проект «Смена времени суток».	4		4
4.12	Создание игры «Угадай слово».	7		7
4.13	Создание тестов – с выбором ответа и без.	9		9
4.14	Программирование исполнителя Перо	12		12
4.15	Свободное проектирование. Создание проектов по собственному замыслу. Регистрация в Скретч-сообществе.	19		19
4.16	Презентация творческих проектов.	3		3
Итого:		162		

2.1. Содержание программы

Раздел 1. Знакомство с программной средой Scratch

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

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

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

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.

3. Методическое обеспечение программы

3.1. Методические виды продукции:

- рабочие места по количеству обучающихся, оснащенные моноблоками или ноутбуками с установленным программным обеспечением, находящемся в свободном доступе;
- выход в интернет с каждого рабочего места.
- рабочее место преподавателя, оснащенное ноутбуком с установленным программным обеспечением;
- интерактивная доска;
- комплект учебно-методической документации: рабочая программа объединения, раздаточный материал, задания,
- цифровые компоненты учебно-методических комплексов (презентации).

3.2. Дидактический материал: компьютерные презентации

3.3. Описание методов обучения.

словесный, наглядный практический, объяснительно-иллюстративный, частично-поисковый, исследовательский, проблемный, проектный и воспитания (убеждение, поощрение, упражнение, стимулирование, мотивация и др.)

3.4. Применяемые технологические технологии

3.5. Формы проведения занятий:

Объяснение, беседа, задания. Основой данного кружка является выполнение большого объема практических заданий. Именно практика способствует наиболее крепкому усвоению материала.

4. Список литературы

Литература для педагога:

1. Рындак В. Г., Дженжер В. О., Денисова Л. В. Проектная деятельность школьника в среде программирования Scratch. — Оренбург: Оренб. гос. ин-т. менеджмента, 2009.
2. «Пропедевтика идей параллельного программирования в средней школе при помощи среды Scratch», В.Г. Рындак, В.О. Джинжер, Л.В. Денисова;
3. «Ранее обучение программирование в среде Scratch», В.Г. Рындак, В.О. Джинжер, Л.В.Денисова;

Интернет ресурсы:

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