


Автономная некоммерческая организация
дополнительного профессионального образования
«Лицейская ассоциация «Лица»

Ижевск, Удмуртская Республика, 426000
ОГРН 1191832016936; ИНН 1831195417

Рассмотрено на Педагогическом совете
от 02 09 2024г.
Протокол № 1

Утверждено:
Приказом директора АНО ДПО «Лица»
 И.Ю.Шкляевой
Приказ № 1 от 02 09 2024 г.



**Дополнительная общеобразовательная общеразвивающая программа
технической направленности
«Программирование Scratch»
для детей 9-10 лет
на 2024-2025 учебный год**

Составили:
педагоги дополнительного образования
Филимонова Л.В.
Мухутдинова А.Н.

Ижевск, 2024 год

Пояснительная записка

Глобальная информатизация и компьютеризация общества предъявляют высокие требования к подрастающему поколению, которому необходимо обладать высоким уровнем компьютерной грамотности, уметь быстро находить необходимую информацию, оперативно ее обрабатывать, передавать, хранить и грамотно представлять.

Программа «Программирование в среде Scratch» посвящена знакомству с основными понятиями, используемыми в языках программирования, решению большого количества творческих задач, многие из которых моделируют процессы и явления из различных предметных областей.

Реализация программы направлена на формирование научного мировоззрения, развитие мышления посредством изучения вопросов программирования и алгоритмизации, на формирование у учеников информационной культуры и грамотности, приобщение к информационным технологиям.

Использование метода проектов позволяет обеспечить условия для развития у ребят навыков самостоятельной постановки задач и выбора оптимального варианта их решения, самостоятельного достижения цели, анализа полученных результатов с точки зрения решения поставленной задачи. Приобретенные в данном курсе знания и умения могут быть использованы при участии в олимпиадах по программированию, при решении задач по физике, химии и другим наукам.

Курс «Программирование в среде Scratch» рассчитан на 30 часов и предназначен для учеников 3 класса.

Направленность программы

Технической направленности. Программа предназначена для использования в системе дополнительного образования детей.

Новизна данной образовательной программы

Данная программа охватывает алгоритмическое направление, а также вопросы практического использования полученных знаний при решении задач. Предоставляется возможность командной разработки, создания коллективных проектов.

Актуальность программы

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.

Педагогическая целесообразность разработки и внедрения данной образовательной программы обусловлена тем, что назрела необходимость комплексного подхода в обучении учеников современным языкам программирования.

Отличительная особенность. Отличительной особенностью данной программы являются компетенции, которые приобретает ученик по окончании курса:

- умение объяснять и использовать на практике как простые, так и сложные структуры данных и конструкции для работы с ними;
- умение искать и обрабатывать ошибки в коде;
- умение разбивать решение задачи на подзадачи;
- способность писать грамотный, красивый код;

- способность анализировать как свой, так и чужой код;
- способность работать с информацией: находить, оценивать и использовать информацию из различных источников, необходимую для решения профессиональных задач (в том числе на основе системного подхода);

- способность грамотно строить коммуникацию, исходя из целей и ситуации.

Адресат программы: Возраст детей, участвующих в реализации данной программы: 9-10 лет. Условия набора учащихся: принимаются все желающие. Наполняемость в группах до 14 человек.

Объем программы: 30 часов

Форма обучения: очная.

Методы обучения:

- ✓ Технология продуктивного чтения, которая основана на природосообразной технологии формирования типа правильной читательской деятельности;
- ✓ ИКТ-технологии, которые помогают сделать обучение более эффективным. Использование данного вида технологии является одним из требований ФГОС ООО;
- ✓ Технологии проектов, форма организации занятий, предусматривающая комплексный характер деятельности всех обучающихся по получению образовательной продукции за определённый промежуток времени – от одного урока до нескольких месяцев;
- ✓ Навыки и компетенции XXI века, которая основана на отработке умений работать в малой группе, взаимообучение;
- ✓ Проблемно-диалоговое обучение, тип обучения, обеспечивающий творческое усвоение материала посредством специально организованного диалога, формирование познавательной мотивации с целью понимания учеником нового материала.

Тип занятий: комбинированный

Формы проведения занятий: лекция, учебная игра, тематические задания по подгруппам, практические занятия, семинар, тематическая дискуссия, групповая консультация.

Срок освоения программы: 1 год – 30 учебных недель

Режим занятий: Занятия проводятся 1 раз в неделю по 1 часу, недельная нагрузка 1 час (30 часов в год). Продолжительность занятия - 40 минут.

Особенности организации образовательного процесса

Образовательный процесс осуществляется в группах с детьми одного возраста. Состав группы постоянный; количество учащихся 14 человек.

Программа предоставляет учащимся возможность освоения учебного содержания занятий с учетом их уровней общего развития, способностей, мотивации. В рамках программы предполагается реализация параллельных процессов освоения содержания программы на разных уровнях доступности и степени сложности, с опорой на диагностику стартовых возможностей каждого из участников.

1.1 Цель и задачи программы

Цель: воспитание творческой личности, обогащенной общетехническими знаниями и умениями, развитие индивидуальных творческих способностей, интереса к науке и технике. Данная программа решает следующие основные задачи:

Образовательные:

- овладеть навыками составления алгоритмов;
- изучить функциональность работы основных алгоритмических конструкций;
- сформировать представление о профессии «программист»;
- сформировать навыки разработки программ;
- познакомить с понятием проекта и алгоритмом его разработки;
- сформировать навыки разработки проектов: интерактивных историй, квестов, интерактивных игр, обучающих программ, мультфильмов, моделей и интерактивных презентаций.

Развивающие:

- способствовать развитию критического, системного, алгоритмического и творческого мышления;
- развивать внимание, память, наблюдательность, познавательный интерес;
- развивать умение работать с компьютерными программами и дополнительными источниками информации;
- развивать навыки планирования проекта, умение работать в группе.

Воспитательные:

- формировать положительное отношение к информатике и ИКТ;
- развивать самостоятельность и формировать умение работать в паре или в малой группе;
- формировать умение, демонстрировать результаты своей работы.

1.2 Содержание программы

Учебный план

Название раздела	Количество часов		
	всего	теоретических	практических
Интерфейс программы Scratch (1 ч)	1	0,75	0,25
Начало работы в среде Scratch (2 ч)	2	1,25	0,75
Основные скрипты программы Scratch (18 ч).	18	6	12
Работа с несколькими объектами. Синхронизация их работы (4 ч).	4	1	3
Использование программы Scratch для создания мини-игр (7 ч).	5	1	4
	30	10	20

Содержание учебно-тематического плана

Интерфейс программы Scratch (1 ч)

Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.

Теория. История создания среды Scratch. Основные базовые алгоритмические конструкции (линейные алгоритмы, с условным оператором, циклического типа с предусловием и постусловием) и их исполнение в среде Scratch. Понятие исполнителя, алгоритма и программы, их назначение, виды и использование. Виды управления исполнителем. Способы записи алгоритма. Основные характеристики исполнителя. Система команд исполнителя. Понятие проект, его структура и реализация в среде Scratch. Основные компоненты проекта Scratch: спрайты и скрипты. Принцип создания анимации и движения объектов. Листинг программы. Сцена. Текущие данные о спрайте. Стил поворота. Закладки. Панель инструментов, Новый спрайт. Координаты мышки. Режим представления. Окно скриптов. Окно блоков. Блоки стека. Блоки заголовков. Блоки ссылок. Самодостаточные и открытые скрипты.

Начало работы в среде Scratch (2 ч)

Сцена. Редактирование фона. Добавление фона из файла.

Теория. Сцена. Ширина и высота сцены. Текущие координаты объекта. Редактирование текущего фона. Вставка нового фона из файла. Вставка стандартного фона из

библиотечного модуля среды. Рисование фона в графическом редакторе. Создание нескольких фонов в одной сцене.

Практика. Создание фона сцены на выбранную учащимся тему.

Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.

Теория. Стандартный объект. Спрайты. Список спрайтов. Редактор рисования для создания новых спрайтов. Инструменты рисования (кисточка, линия, текст, эллипс) и редактирования объекта (ластик, заливка, поворот, выбор, печать, пипетка). Центрирование костюма. Масштабирование спрайта. Загрузка на сцену спрайтов из стандартной коллекции среды Scratch. Вставка спрайтов из файлов форматов JPG, BMP, PNG, GIF. Выбор случайного спрайта. Удаление спрайтов.

Практика. Создание фона сцены и прорисовка основных спрайтов для Scratch-истории.

Основные скрипты программы Scratch (18 ч).

Синий ящик – команды движения. Темно-зеленый ящик – команды рисования.

Теория. Команды – *идти; повернуться направо (налево); повернуть в направлении; повернуться к; изменить x (y) на; установить x (y) в; если край, оттолкнуться*. Принципиальное различие действия команд *идти в* и *плыть в*. Назначение сенсоров *положение x, положение y и направлении*. Команды – *очистить, опустить перо, поднять перо, установить цвет пера, изменить цвет пера на, установить цвет пера, изменить тень пера, установить тень пера, изменить размер пера на, установить размер пера, печать*.

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

Фиолетовый ящик – внешний вид объекта. Оживление объекта с помощью добавления костюмов.

Теория. Костюмы спрайта. Копирование и редактирование костюма спрайта с помощью редактора рисования. Переупорядочивание костюмов. Команды – *перейти к костюму, следующий костюм, говорить...в течении...секунд, сказать, думать, думать...секунд, изменить... эффект на, установить эффект...в значение, убрать графические эффекты, изменить размер на, установить размер, показаться, спрятаться, перейти в верхний слой, перейти назад на...1 слов*. Назначение сенсоров *костюм* и *размер*. Понятие раскадровки движения. Изменение костюма спрайта для имитации движения.

Практика. Создание программы для управления внешним видом объекта. Создание Scratch-историй с имитацией хождения и движения объектов.

Желтый ящик – контроль. Лиловый ящик – добавление звуков.

Теория. Кнопка с зеленым флажком и ее назначение. Управление последовательностью выполнения скриптов. Понятие управляющих сообщений. Команды – *передать, передать и ждать, когда я получу*. Скрипты для создания условных конструкций программы – *если, если...или*. Скрипты для управления циклами – *всегда, повторить, всегда, если, повторять до..* Команды – *когда клавиша...нажата, когда щелкнут по, ждать...секунд, ждать до, остановить скрипт, остановить все*. Загрузка звуков из стандартной коллекции и из файлов жесткого диска. Запись звука через микрофон. Принципиальная разница работы команд *играть звук* и *играть звук до завершения*. Команды – *остановить все звуки, барабану играть...тактов, оставшиеся...тактов, ноту...играть...тактов, выбрать*

инструмент, изменить громкость, установить громкость, изменить темп на, установить темп. Назначение сенсоров *громкость* и *темп*.

Практика. Создание программ с элементами управления объектом. Озвучивание Scratch-историй.

Использование в программах условных операторов.

Теория. Базовая конструкция ветвление, назначение, виды (полная и неполная форма). Понятие условия. Изменение порядка выполнения скриптов в зависимости от условия. Разветвление листинга программы. Скрипты условных операторов. Использование неполной формы ветвления в системе Scratch.

Практика. Создание программ с изменением последовательного выполнения скриптов при наличии условий.

Функциональность работы циклов. Цикличность выполнения действий в зависимости от поставленных условий.

Теория. Циклы с фиксированным числом повторений. Заголовок цикла. Тело цикла. Циклы с условным оператором. Заголовок цикла. Тело цикла. Предусловие и постусловие. Зацикливание.

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

Зеленый ящик – операторы. Использование арифметических и логических блоков вместе с блоками управления.

Теория. Числа. Строинги. Логические величины. Логические выражения. Арифметические операции. Логические операции. Операции сравнения. Команды для работы со строингами – *слить, буква...в, длинна строки*. Команда *выдать случайное от...до*. Использование арифметических и логических блоков в листинге программы. Просмотр полученного результата.

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

События. Оранжевый ящик – переменные.

Теория. События в проектах Scratch. Понятие переменных и необходимость их использования в листинге программы. Глобальные и локальные переменные. Имя переменной и правила его формирования. Команды для переменных - *поставить...в, изменить...на, показать переменную, спрятать переменную*. Удаление переменных. Создание счетчиков с помощью переменных.

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

Списки.

Теория. Создание списков и необходимость их использования в проектах Scratch. Добавление в список данных. Удаление данных из списка. Удаление списка. Команды работы со списками – *добавить...к, удалить...из, поставить...в...из, заменить элемент...в...на, элемент...из, длина списка*.

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

Голубой ящик – сенсоры. Ввод-вывод данных.

Теория. Понятие сенсора. Правила применения и область действия команд *касается*, *касается цвета* и *цвет.касается*. Функционал команды *спросить...и ждать*. Сенсоры *мышка по x*, *мышка по y*, *мышка нажата?*, *клавиша...нажата?*, *расстояние до*, *перезапустить таймер*. Сенсоры, значение которых можно выводить на экран – *ответ*, *таймер*, *громкость*, *громко?* ...*значение сенсора* и *сенсор*.... Необходимость ввода данных для их обработки в программе. Ввод данных с помощью команды *спросить*. Вывод конечного результата обработки с помощью команд *говорить* и *сказать*.

Практика. Создание проектов с использованием значений сенсоров и команды *спросить*. Создание программ для обработки данных пользователя с выводом на экран конечного результата.

Работа с несколькими объектами. Синхронизация их работы (4 ч).

Последовательность и параллельность выполнения скриптов.

Теория. Последовательные и параллельные потоки в программах Scratch. Одновременная и попеременная работа нескольких исполнителей.

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

Взаимодействие между спрайтами. Управление через обмен сообщениями.

Теория. Решение проблемы появления новых исполнителей только после того, как старые исполнители выполнили свои действия. Взаимодействие спрайтов с неподвижными объектами с помощью команд *касается* и *касается цвета*. Взаимодействие спрайтов с помощью команд *передать* и *когда я получу*. Использование сообщений для создания событий.

Практика. Создание Scratch-историй с взаимодействием нескольких исполнителей и неподвижных объектов. Создание Scratch-историй с взаимодействием нескольких исполнителей.

Использование программы Scratch для создания мини-игр (5 ч).

Виды компьютерных игр. Алгоритмическая разработка листинга программы.

Теория. Компьютерные игры – вред или польза. Виды компьютерных игр. Этапы разработки игр программистами.

Практика. Алгоритмическая разработка проекта, запись на естественном языке событий и точек взаимодействия героев будущей игры.

Разработка базовых спрайтов для игры. Формирование базовых скриптов.

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

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

Синхронизация работы скриптов для разных спрайтов.

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

Переход из одной сцены в другую. Создание интерфейса игры.

Теория. Односторонний (без возможности вернуться назад) переход из одного пространства в другое. Понятие интерфейса. Элементы интерфейса. Основные принципы дизайна интерфейсов. Обратная связь. Необходимые элементы меню.

Практика. Создать программу для перемещения объекта по игровой карте и разработать интерфейс для Scratch-проекта.

Сообщество Scratch в Интернете. Просмотр и публикация проектов.

Теория. Правила работы в сети. Интернет-сообщества. Сообщество Scratch. Регистрация на сайте. Использование заимствованных кодов и объектов. Авторские права. Публикация проектов Scratch.

Практика. Регистрация на сайте сообщества Scratch. Просмотр проектов сообщества и публикация собственных проектов.

Учебно-тематический план

Дата	Название раздела	№ урока	Название темы	Кол-во часов		
				всего	теоретических	практических
	Интерфейс программы Scratch (1 ч)	1	Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.	1	0,75	0,25
		Начало работы в среде Scratch (2 ч)	2	Сцена. Редактирование фона. Добавление фона из файла.	1	0,75
			3	Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.	1	0,5
	Основные скрипты программы Scratch (18 ч).	4	Синий ящик – команды движения.	1	0,5	0,5
		5	Темно-зеленый ящик – команды рисования.	1	0,5	0,5
		6	Фиолетовый ящик – внешний вид объекта.	1	0,5	0,5
		7	Оживление объекта с помощью добавления костюмов.	1	0,5	0,5
		8	Желтый ящик – контроль.	1	0,5	0,5
		9	Лиловый ящик – добавление звуков.	1	0,5	0,5
		10	Использование в программах условных операторов.	1	0,25	0,75
		11	Использование в программах условных операторов.	1	0,25	0,75
		12	Функциональность работы циклов.	1	0,25	0,75
		13	Цикличность выполнения действий в зависимости от поставленных условий.	1	0,25	0,75
		14	Зеленый ящик – операторы.	1	0,25	0,75
		15	Использование арифметических и логических блоков вместе с блоками управления.	1	0,25	0,75
		16	События. Оранжевый ящик – переменные.	1	0,25	0,75
		17	Списки.	1	0,25	0,75

25		18	Голубой ящик – сенсоры. Ввод-вывод данных.	1	0,25	0,75
		19	Голубой ящик – сенсоры. Ввод-вывод данных.	1	0,25	0,75
		20	Последовательность и параллельность выполнения скриптов.	1	0,25	0,75
		21	Последовательность и параллельность выполнения скриптов.	1	0,25	0,75
	Работа с несколькими объектами и. Синхронизация их работы (4 ч).	22	Взаимодействие между спрайтами.	1	0,25	0,75
		23	Взаимодействие между спрайтами. Управление через обмен сообщениями.	1	0,25	0,75
		24	Виды компьютерных игр. Алгоритмическая разработка листинга программы.	1	0,25	0,75
		25	Виды компьютерных игр. Алгоритмическая разработка листинга программы.	1	0,25	0,75
	Использование программы Scratch для создания мини-игр (5 ч).	26	Разработка базовых спрайтов для игры. Формирование базовых скриптов.	1	0,25	0,75
		27	Разработка базовых спрайтов для игры. Формирование базовых скриптов.	1	0,25	0,75
		28	Синхронизация работы скриптов для разных спрайтов.	1	0,25	0,75
		29	Синхронизация работы скриптов для разных спрайтов.	1	0,25	0,75
		30	Переход из одной сцены в другую. Создание интерфейса игры.	1		1

1.4 Планируемые результаты

Личностные результаты:

- широкие познавательные интересы, инициатива и любознательность, мотивы познания и творчества; готовность и способность обучающихся к саморазвитию и реализации творческого потенциала в предметнопродуктивной деятельности за счет развития их образного, алгоритмического и логического мышления;
- готовность к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики и ИКТ;
- интерес к информатике и ИКТ, стремление использовать полученные знания в процессе обучения другим предметам и в жизни;
- способность увязать учебное содержание с собственным жизненным опытом и личными смыслами, понять значимость подготовки в области информатики и ИКТ в условиях развития информационного общества;
- готовность к самостоятельным поступкам и действиям, принятию ответственности за их результаты; готовность к осуществлению индивидуальной и коллективной информационной деятельности;
- способность к избирательному отношению к получаемой информации за счет умений ее анализа и критичного оценивания; ответственное отношение к информации с учетом правовых и этических аспектов ее распространения;
- развитие чувства личной ответственности за качество окружающей информационной среды;
- способность и готовность к принятию ценностей здорового образа жизни за счет знания основных гигиенических, эргономических и технических условий безопасной эксплуатации средств ИКТ.

Метапредметные результаты:

- владение умениями организации собственной учебной деятельности, включающими: целеполагание как постановку учебной задачи на основе соотнесения того, что уже известно, и того, что требуется установить;
- планирование – определение последовательности промежуточных целей с учетом конечного результата, разбиение задачи на подзадачи, разработка последовательности и структуры действий, необходимых для достижения цели при помощи фиксированного набора средств;
- прогнозирование – предвосхищение результата;
- контроль – интерпретация полученного результата, его соотнесение с имеющимися данным и с целью установления соответствия или несоответствия (обнаружения ошибки);
- коррекция – внесение необходимых дополнений и корректив в план действий в случае обнаружения ошибки;
- оценка – осознание учащимся того, насколько качественно им решена учебно-познавательная задача;
- владение основными универсальными умениями информационного характера: постановка и формулирование проблемы;
- поиск и выделение необходимой информации, применение методов информационного поиска;
- структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных условий;
- самостоятельное создание алгоритмов деятельности при решении проблем творческого и поискового характера;
- владение основами продуктивного взаимодействия и сотрудничества со сверстниками и взрослыми: умение правильно, четко и однозначно сформулировать мысль в понятной собеседнику форме;

- умение осуществлять в коллективе совместную информационную деятельность, в частности при выполнении проекта;
- умение выступать перед аудиторией, представляя ей результаты своей работы с помощью средств ИКТ;
- использование коммуникационных технологий в учебной деятельности и повседневной жизни.

Предметные результаты:

- умение использовать термины «информация», «сообщение», «данные», «алгоритм», «программа»; понимание различий между употреблением этих терминов в обыденной речи и в информатике;
- умение составлять линейные, разветвляющиеся и циклические алгоритмы управления исполнителями на языке программирования Scratch;
- умение использовать логические значения, операции и выражения с ними;
- овладение понятиями спрайт, объект, скрипт, обработка событий;
- умение формально выполнять алгоритмы, описанные с использованием конструкций ветвления (условные операторы) и повторения (циклы), вспомогательных алгоритмов;
- умение создавать и выполнять программы для решения несложных алгоритмических задач в программе Scratch;
- навыки выбора способа представления данных в зависимости от поставленной задачи.

2.1 Календарный учебный график

Продолжительность учебного года: 30 учебных недель, в учебной неделе 5 дней.

Дата начала учебного периода: 03.10.2023 г.

Дата окончания учебного периода: 30.05.2024г.

2.2 Условия реализации программы

Аппаратные средства:

Наиболее рациональным с точки зрения организации деятельности детей в школе является установка в компьютерном классе 13–15 компьютеров (рабочих мест) для школьников и одного компьютера (рабочего места) для педагога.

Предполагается объединение компьютеров в локальную сеть с возможностью выхода в Интернет, что позволяет использовать сетевые цифровые образовательные ресурсы.

Минимальные требования к техническим характеристикам каждого компьютера следующие:

- ✓ процессор – не ниже *Celeron* с тактовой частотой 2 ГГц;
- ✓ оперативная память – не менее 256 Мб;
- ✓ жидкокристаллический монитор с диагональю не менее 15 дюймов;
- ✓ жёсткий диск – не менее 80 Гб;
- ✓ клавиатура;
- ✓ мышь;
- ✓ устройство для чтения компакт-дисков (желательно);
- ✓ аудиокарта и акустическая система (наушники или колонки).

Кроме того, в кабинете информатики должны быть:

- ✓ принтер на рабочем месте учителя;
- ✓ проектор на рабочем месте учителя;
- ✓ сканер на рабочем месте учителя

Программные средства:

На компьютерах, которые расположены в кабинете информатики, должна быть установлена операционная система *Windows* или *Linux*, а также необходимое программное обеспечение:

- ✓ текстовый редактор (*Блокнот* или *Gedit*) и текстовый процессор (*Word* или *OpenOffice.orgWriter*);
- ✓ табличный процессор (*Excel* или *OpenOffice.orgCalc*);
- ✓ графический редактор *Gimp* (<http://gimp.org>);
- ✓ среда разработки: *Scratch* и другие программные средства.

Кадровое обеспечение:

Реализовывать программу могут педагоги дополнительного образования, знающие особенности обучения программированию

2.3 Формы аттестации

Контроль знаний и умений:

Текущий контроль уровня усвоения материала осуществляется по результатам выполнения учащимися тестов, практических работ. Итоговой аттестацией является выполнение творческого проекта.

Формы организации учебных занятий:

Занятия организуются в форме лекций, семинаров, дискуссий, мозговых атак, практикумов, самостоятельного решения заданий.

В ходе изучения проводятся краткие теоретические опросы по знанию основных понятий. Используется принцип непрерывного повторения, что улучшает процесс запоминания.

2.4 Методические материалы

- ✓ Методические рекомендации, дидактический материал (игры, сценарии, задания, задачи, способствующие включению внимания, восприятия, мышление, воображения учащихся);
- ✓ Учебно-планирующая документация;
- ✓ Диагностический материал (кроссворды, анкеты, тестовые и кейсовые задания);
- ✓ Наглядный материал, аудио и видео материал

3. Список литературы

Литература для педагога:

1. Л.Л.Босова, Сорокина Т.Е. Методика применения интерактивных сред для обучения младших школьников программированию: Информатика и образование № 7(256) сентябрь 2014 г.
2. Сорокина Т.Е. Пропедевтика программирования со Scratch: Слово учителю, сетевое издание ГМЦ <http://slovo.mosmetod.ru/avtorskie-materialy/item/238-sorokina-t-e-propedevtikaprogramirovaniya-so-scratch>
3. Рындак В. Г., Дженжер В. О., Денисова Л. В. Проектная деятельность школьника в среде программирования Scratch. — Оренбург: Оренб. гос. ин-т. менеджмента, 2009.
4. «Пропедевтика идей параллельного программирования в средней школе при помощи среды Scratch», В.Г. Рындак, В.О. Дженжер, Л.В. Денисова;
5. «Раннее обучение программированию в среде Scratch», В.Г. Рындак, В.О. Дженжер, Л.В. Денисова;
6. Книга юных программистов на Scratch. Голиков Денис и Голиков Артём - Издательство Smashwords, 2013
7. Программы внеурочной деятельности для начальной и основной школы. 3-6 классы. Цветкова М.С., Богомолова О.Б., БИНОМ. Лаборатория знаний, 2013.

Литература для учащихся:

1. Творческие задания в среде Scratch: рабочая тетрадь для 5-6 классов/ Ю.В. Пашковская. – М.: БИНОМ. Лаборатория знаний, 2014. – 200 с.: ил.

Интернет ресурсы

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