

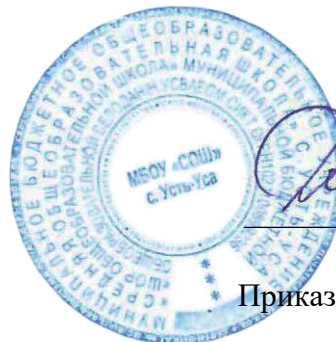
УПРАВЛЕНИЕ ОБРАЗОВАНИЯ АДМИНИСТРАЦИИ  
МУНИЦИПАЛЬНОГО ОБРАЗОВАНИЯ  
ГОРОДСКОГО ОКРУГА «УСИНСК»  
«УСИНСК» КАР КЫТШЫН МУНИЦИПАЛЬНОЙ ЮКОНЛОН  
АДМИНИСТРАЦИЯСА ЙӖЗӖС ВЕЛӖДӖМӖН ВЕСЬКӖДЛАНӖН

Муниципальное бюджетное  
общеобразовательное учреждение  
"Средняя общеобразовательная  
школа" с. Усть-Уса



Муниципальной бюджетной  
общеобразовательной велӖданӖн  
«Общеобразовательной шӖр школа»  
Усавом сикт

Рассмотрено  
методическим советом школы  
Протокол № 4 от 06.06.2023



Утверждаю  
\_\_\_\_\_  
Директор школы  
Е.В.  
Дьячкова  
Приказ № 235 от 06. 06.2023

**Дополнительная  
общеобразовательная общеразвивающая программа  
«Программирование в среде Scratch»**

Возраст детей 14-17 лет  
Срок реализации 1 год

Составитель:  
учитель информатики, педагог дополнительного образования  
Волков Вячеслав Аркадьевич

село Усть-Уса  
2023 г.

## Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Программирование в среде Scratch» разработана на основе программы курса по выбору «Творческие задания в среде программирования Скретч», изданной в сборнике «Информатика. Математика. Программы внеурочной деятельности для начальной и основной школы: 3-6 класс» М.С.Цветкова, О.Б.Богомолова. – М.: БИНОМ. Лаборатория знаний, 2015.

Программа составлена с учётом:

- Закона РФ от 29.12.2012 N 273-ФЗ "Об образовании в Российской Федерации",
- Концепции развития дополнительного образования детей до 2030 года (утверждена распоряжением Правительства Российской Федерации от 31 марта 2022 г. № 678-р)
- Постановления Главного государственного санитарного врача Российской Федерации от 28 января 2021 г. № 2 «Об утверждении санитарных правил и норма СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания» (раздел VI «Гигиенические нормативы по устройству, содержанию и режиму работы организаций воспитания и обучения, отдыха и оздоровления детей и молодежи»),
- Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утверждённым приказом Министерства просвещения РФ от 09 ноября 2018 г. №196 (в действующей редакции).

**Направленность** - техническая.

**Актуальность программы** В настоящее время информатизация обучения отводится ответственная роль в развитии и становлении активной, самостоятельно мыслящей личности, готовой конструктивно и творчески решать возникающие перед обществом задачи. Поэтому одна из основных задач дополнительного образования состоит в том, чтобы помочь учащимся в полной мере проявлять свои способности, развить творческий потенциал, инициативу, самостоятельность. Формирование интереса к овладению знаний и умений в области информационных технологий является важным средством повышения качества обучения.

**Новизна программы** Можно ли научиться программировать играя? Оказывается, можно. Американские ученые, задумывая новую учебную среду для обучения школьников программированию, стремились к тому, чтобы она была понятна любому ребенку, умеющему читать.

Назначение «Scratch» в переводе с английского имеет несколько значений. Это и царапина, которую оставляет Котенок – символ программы, и каракули, символизирующие первый, еще неуклюжий самостоятельный опыт, и линия старта. Со Скретчем удобно стартовать. Сами разработчики характеризуют программу так: «Скретч предлагает низкий пол (легко начать), высокий потолок (возможность создавать сложные проекты) и широкие стены (поддержка большого многообразия проектов)».

**Отличительная особенность программы**

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

Скретч легко перекидывает мостик между программированием и другими школьными науками. Так возникают межпредметные проекты. Они могут сделать наглядными понятия отрицательных чисел и координат или, например, уравнения плоских

фигур, изучаемых на уроках геометрии. В них оживут исторические события и географические карты. А тесты по любым предметам сделают процесс обучения веселым и азартным.

Скретч – свободно распространяемая программа. Она одинаково устанавливается в Windows, в Macintosh, в Ubuntu.

**Адресат программы:** программа составлена для учащихся 14-17 лет.

**Вид программы по уровню освоения** - ознакомительный

**Объем программы** - 37 недель, 37 часов в год.

**Срок освоения** данной дополнительной общеобразовательной общеразвивающей программы составляет 1 год.

**Формы обучения:** проведение комбинированных занятий - занятия состоят из теоретической и практической частей, причём большее количество времени занимает практическая часть.

При проведении занятий традиционно используются три формы работы:

- демонстрационная, когда обучающиеся слушают объяснения педагога и наблюдают за демонстрационным экраном на ученических рабочих местах;
- фронтальная, когда обучающиеся синхронно работают под управлением педагога;
- самостоятельная, когда обучающиеся выполняют индивидуальные задания в течение части занятия или нескольких занятий.

**Режим занятий:** занятия проходят один раз в неделю, количество детей в группе – 10 чел.

**Особенности организации образовательного процесса** – состав группы постоянный, виды занятий - групповые

## 1.2. Цели и задачи программы

**Цель программы:** обучение программированию через создание творческих проектов по информатике. Курс развивает творческие способности учащихся, а также закладывает пропедевтику наиболее значимых тем курса информатики и позволяет успешно готовиться к участию в олимпиадах по математике и информатике.

**Задачи:**

Образовательные:

- формирование у обучающихся базовых представлений о языках программирования, алгоритме, исполнителе, способах записи алгоритма;
- изучение основных базовых алгоритмических конструкций;
- знакомство с понятием переменной и команды присваивания;
- овладение навыками алгоритмизации задачи;
- овладение понятиями класс, объект, обработка событий;
- освоение основных этапов решения задачи;
- формирование навыков разработки, тестирования и отладки несложных программ;
- знакомство с понятием проекта, его структуры, дизайна и разработки;

Развивающие:

- предоставление возможности самовыражения в компьютерном творчестве.

Воспитательные:

- освоение навыков планирования проекта, умение работать в группе;
- выработка навыков работы в сети для обмена материалами работы;

## 1.3. Содержание программы

### Учебный план

№	Тема	Количество часов		
		Теория	Практика	Всего

1	Знакомство со средой Scratch	1	1	2
2	Управление спрайтами. Координатная плоскость	1	1	2
3	Навигация в среде Scratch. Управление командами	4	20	24
4	Создание игры		1	1
5	Создание тестов		1	1
6	Публикация проектов		2	2
7	Резерв		5	5
Итого:		8	29	37

## Содержание учебного плана

### Знакомство со средой Scratch (2 часа)

Ознакомление с учебной средой программирования Скретч. Элементы окна среды программирования. Спрайты. Хранилище спрайтов. Понятие команды. Разновидности команд. Структура и составляющие скриптов - программ, записанных языком Скретч. Понятие анимации. Команды движения и вида. Анимация движением и изменением вида спрайта.

Создание самого простого проекта, его выполнения и сохранения. Хранилище проектов. Создание и редактирование скриптов. Перемещение и удаление спрайтов.

*Ученик описывает:*

- интерфейс среды программирования Скретч;
- понятие программного проекта;
- методику создания, сохранения и выполнения проекта;
- понятие спрайта, как управляемого графического объекта;
- понятие скрипта, как программы управления спрайтом;
- понятие события;
- методику редактирования скриптов;
- понятие команды;
- понятие стека, как последовательности команд;
- понятие команд управления, вида и движения;
- общую структуру Скретч-проекта;

*приводит примеры:*

- команд в языке программирования Скретч;

*умеет:*

- открывать среду программирования;
- создавать новый проект, открывать и хранить проекты;
- запускать разработанный Скретч-проект;
- пользоваться элементами интерфейса среды программирования;
- открывать и закрывать окна инструментов, которые есть в среде программирования Скретч;
- перемещать, открывать и удалять спрайты;
- редактировать скрипты за предоставленным образцом
- реализовать самые простые алгоритмы перемещения спрайтов в виде скриптов среды программирования Скретч;

*понимает:*

- содержание скриптов, записанных языком программирования Скретч та суть событий, которые происходят во время выполнения скрипта.

### Управление спрайтами. Координатная плоскость (2 ч.)

Создание спрайтов, изменение их характеристик (вида, размещения). Графический редактор Скретч. Понятие о событиях, их активизации и обработке. Понятие сцены, налаживания вида сцены. Обработка событий сцены

*Ученик описывает:*

- процесс создания спрайтов;
- назначение элементов управления спрайтов;
- процесс добавления спрайта;
- процесс активации события и ее обработки;

*называет:*

- параметры спрайта;

*умеет:*

- создавать и редактировать спрайты
- называть спрайт, задавать ему место на сцене;
- налаживать сцену

## **Навигация в среде Scratch. Управление командами (24 ч.)**

*Величины и работа с ними*

Понятие переменной и константы. Создание переменных. Предоставление переменным значений, пересмотр значений переменных. Команды предоставления переменных значений. Использование переменных

*Ученик описывает:*

- понятие переменной, ее имени и значения;
- понятие константы;
- правила создания переменные;
- использование команд предоставления значений;
- способы пересмотра значений переменных;

*называет:*

- параметры спрайтов и Сцены;
- датчики среды программирования Скретч;

*умеет:*

- создавать переменные;
- использовать датчики для предоставления значений переменным и управления параметрами спрайтов и сцены;
- предоставлять переменным значений параметров спрайтов и участков сцены, других переменных;
- передавать командам управления значения переменные, параметры спрайтов и сцены;
- осуществлять обмен значениями между двумя переменными;
- руководить отображением значений переменных;
- использовать слайдеры для предоставления переменным значений

*Арифметические операции и выражения*

Понятие операции и выражения. Арифметические операции . Основные правила построения, вычисления и использования выражений. Присвоение значений выражений переменным. Понятие локальной и глобальной переменной. Генератор псевдослучайных чисел

*Ученик описывает:*

- понятие операции и операнда;
- способы использования операций в программе;
- понятие выражения;
- приоритет операций и порядок вычисления значений выражений;
- порядок записи выражений;
- назначение генератора псевдослучайных чисел;

*приводит примеры:*

- арифметических выражений;

*умеет:*

- записывать языком программирования Скретч арифметические выражения;
- использовать в выражениях переменные пользователя и ссылки на атрибуты спрайтов и сцены;
- придавать значение выражений переменным;
- использовать генератор псевдослучайных чисел;

*Команды ветвления*

Понятие условия. Формулировка условий. Операции сравнения. Простые и составлены условия. Алгоритмическая конструкция ветвления. Команды ветвления *Если...*, *Если...Иначе....* Выполнение скриптов с ветвлениями. Вложенные команды ветвления

*Ученик описывает:*

- понятие условия, значений «истинность» и «ложь»;
- использование логических операций *И*, *Или*, *Не*;
- порядок записи составных условий;
- алгоритмические конструкции ветвлений разных видов, их обозначения на блок-схемах;
- создание команд ветвления разных видов;
- создание вложенных ветвлений;

*умеет:*

- записывать языком программирование Скретч простые и составные логические выражения;
- конструировать алгоритмы с разными видами ветвлений и строить соответствующие блок-схемы;
- создавать скрипты с простыми и вложенными ветвлениями;
- создавать проекты, которые требуют проверки соответствия параметров спрайта или среды определенной величине;
- создавать проекты, которые предусматривают выбор варианта поведения спрайта в зависимости от выполнения определенного условия;
- анализировать ход выполнения скриптов, которые имеют команды ветвления

*Команды повторения*

Команда повторения и ее разновидности: циклы с известным количеством повторений, циклы с предусловием и постусловием. Команды повторения в Скретче: *Повторить...*, *Всегда если...*, *Повторять пока...* . Вложенные циклы. Операторы прерывания циклов.

*Ученик описывает:*

- разные виды команд повторения и способ их изображения на блок-схеме;
- порядок выбора оптимальной для данного случая команды повторения;
- порядок использования разных команд повторения;

*объясняет:*

- отличие между командами повторений с предусловием, постусловием и известным количеством повторений;

*умеет:*

- составлять скрипты, в которых используются конструкции повторения;
- использовать циклы для создания анимации;
- использовать вложенные циклы

*Обмен сообщениями между скриптами*

Понятие сообщения. Передача сообщения, запуск скриптов при условии получения сообщения вызова. Обмен данными между скриптами.

*Ученик описывает:*

- понятие сообщения ;

- команды передачи сообщения и запуска скриптов при условии получения сообщения;
- принцип обмена данными между скриптами;

*объясняет отличие:*

- между командами передачи сообщения разных видов;

*умеет:*

- вызывать запуск скриптов передачей им сообщений;
- передавать между скриптами значение параметру;
- применять вызовы скриптов во время создания проектов, в которых многократно выполняются одинаковые последовательности команд;

*Программируемое построение графических изображений*

Команды рисования. Создание проектов с программируемым построением изображений на сцене путем перемещением спрайтов. Использование команды *Штамп*.

*Ученик описывает:*

- способ программируемого построения изображений;
- команды рисования в Скретче;

*объясняет отличие:*

- между программируемым рисованием и построением изображения в графическом редакторе;

*умеет:*

- создавать скрипты для построения изображений;
- использовать команду Штамп;
- передавать между скриптами значение параметра.

*Списки*

Понятие списка. Создание списков. Понятие индекса, как номера элемента списка. Предоставление значений элементам списка и отображения его содержания. Поиск необходимых данных в списке. Вычисление итоговых показателей для списка. Вычисление итоговых показателей для элементов списка, которые отвечают определенным критериям. Алгоритмы сортировки списков.

*Ученик описывает:*

- понятие списка, как одномерного массива;
- правила создание списков в Скретче;
- понятие индекса элемента списка и порядок обращения к элементу списка за его индексом;
- правила введение/выведение значений элементов списка;
- алгоритм поиска необходимых данных в списке;
- алгоритмы вычисления итоговых показателей для списка и для тех его элементов, которые отвечают заданным критериям;

*умеет:*

- создавать в Скретч-проектах списки (одномерные массивы);
- предоставлять и считывать значение элементов списка;
- реализовать в Скретч алгоритмы поиска данных в списке, которые удовлетворяют определенному условию; вычисление итоговых показателей для всего списка и для тех его элементов, которые отвечают заданным критериям;
- реализовать самые простые алгоритмы упорядочивания элементов списка;

**Создание игры (1 ч.)**

Разработка и создание небольшой программы с использованием заранее подготовленных материалов. Тестирование и отладка проекта. Защита проекта

*Ученик описывает:*

- понятие игры;
- понятие отладки проекта.

*умеет:*

- разрабатывать и создавать простейшую логическую игру;
- проводить тестирование игры с последующим исправлением допущенных логических неточностей.
- представлять публично проект.

### **Создание тестов (1 ч.)**

Разработка и создание теста с использованием заранее подготовленных материалов. Тестирование и отладка проекта. Защита проекта

*Ученик описывает:*

- понятие игры;
- понятие отладки проекта.

*умеет:*

- разрабатывать и создавать простейшую логическую игру;
- проводить тестирование игры с последующим исправлением допущенных логических неточностей.
- представлять публично проект.

### **Публикация проектов (2 ч.)**

Использование заимствованных кодов и объектов, авторские права. Правила работы в сети. Дизайн проекта. Работа со звуком. Основные этапы разработки проекта.

*Ученик описывает:*

- понятие авторского права;
- правила работы в сети;
- правила публикации проектов в сети;
- этапы разработки проекта.

*умеет:*

- разрабатывать дизайн проекта;
- публиковать проект в сети;
- оформлять проект звуковым сопровождением;
- вести работу в соответствии с этапами разработки проекта.

### **Резерв (5 ч.)**

#### **1.4. Планируемые результаты**

В результате реализации программы у обучающихся будут сформированы:

##### Предметные результаты:

- формирование умений формализации и структурирования информации, умения выбирать способ представления данных в соответствии с поставленной задачей с использованием соответствующих программных средств обработки данных;
- формирование навыков и умений безопасного и целесообразного поведения при работе с компьютерными программами и в Интернете, умения соблюдать нормы информационной этики и права.

##### Личностные результаты:

- формирование ответственного отношения к учению;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, творческой и других видов деятельности.

##### Метапредметные результаты:

- умение самостоятельно планировать пути достижения целей, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;
- умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата;
- умение оценивать правильность выполнения учебной задачи, собственные возможности ее решения;



- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;
- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
- умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками; работать индивидуально и в группе;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий.

## 2.1.Календарно-тематический план

№ занятия	Тема занятия	Кол-во часов (теория)	Кол-во часов (практика)	Дата проведения занятия (план)	Дата проведения занятия (факт)
1	Знакомство со средой Scratch. Понятие спрайта и объекта. Создание и редактирование спрайтов и фонов для сцены. Техника безопасности и правила поведения в компьютерном классе	1			
2	Знакомство со средой Scratch (продолжение). Пользуемся помощью Интернета. Поиск, импорт и редакция спрайтов и фонов из Интернета.		1		
3	Управление спрайтами: команды Идти, Повернуться на угол, Опустить перо, Поднять перо, Очистить	1			
4	Координатная плоскость. Точка отсчета, оси координат, единица измерения расстояния, абсцисса и ордината		1		
5	Навигация в среде Scratch. Определение координат спрайта. Команда Идти в точку с заданными координатами	1			
6	Создание проекта «Кругосветное путешествие Магеллана». Команда Плыть в точку с заданными координатами	1			
7	Создание проекта «Кругосветное путешествие Магеллана» (продолжение). Режим презентации	1			
8	Понятие цикла. Команда Повторить. Рисование узоров и орнаментов	1			
9	Конструкция Всегда. Создание проектов «Берегись автомобиля!» и «Гонки по вертикали». Команда Если край, оттолкнуться		1		
10	Ориентация по компасу. Управление курсором движения. Команда Повернуть в направлении. Проект «Полет самолета»		1		

11	Спрайты меняют костюмы. Анимация. Создание проектов «Осьминог», «Девочка, прыгающая через скакалку» и «Бегущий человек»		1		
12	Создание мультипликационного сюжета «Кот и птичка»		1		
13	Создание мультипликационного сюжета «Кот и птичка»		1		
14	Соблюдение условий. Сенсоры. Блок Если. Управляемый стрелками спрайт		1		
15	Создание коллекции игр: «Лабиринт», «Кружащийся котенок»		1		
16	Пополнение коллекции игр: «Опасный лабиринт»		1		
17	Составные условия. Проекты «Хождение по коридору», «Слепой кот», «Тренажер памяти»		1		
18	Датчик случайных чисел. Проекты «Разноцветный экран», «Хаотичное движение», «Кошки-мышки», «Вырастим цветник»		1		
19	Циклы с условием. Проект «Будильник»		1		
20	Запуск спрайтов с помощью мыши и клавиатуры. Проекты «Переодевалки» и «Дюймовочка»		1		
21	Самоуправление спрайтов. Обмен сигналами. Блоки Передать сообщение и Когда я получу сообщение. Проекты «Лампа» и «Диалог»		1		
22	Доработка проектов «Магеллан», «Лабиринт»		1		
23	Датчики. Проекты «Котенок-обжора», «Презентация»		1		
24	Переменные. Их создание. Использование счетчиков. Проект «Голодный кот»		1		
25	Ввод переменных. Проект «Цветы». Доработка проекта «Лабиринт» - запоминание имени лучшего игрока		1		
26	Ввод переменных с помощью рычажка. Проекты «Цветы» (вариант 2), «Правильные многоугольники»		1		
27	Список как упорядоченный набор однотипной информации. Создание списков. Добавление и удаление элементов. Проекты «Гадание», «Назойливый собеседник»		1		
28	Поиграем со словами. Строковые константы и переменные. Операции со строками		1		

29	Создание игры «Угадай слово»		1		
30	Создание тестов – с выбором ответа и без		1		
31 32	Создание проектов по собственному замыслу. Регистрация в Скретч-сообществе. Публикация проектов в сети		2		
33- 37	Повторение. Создание и защита итогового проекта		5		

## 2.2. Условия реализации программы

Программа реализуется в учебном кабинете Центра цифровых и гуманитарных технологий «Точка роста», оборудованном всем необходимым для эффективного проведения занятий.

### Аппаратное обеспечение:

Процессор не ниже Pentium II  
Оперативная память не менее 512 Мб  
Дисковое пространство не меньше 800 Мб  
Монитор с 16-битной видеокартой  
Разрешение монитора не ниже 800x600

### Программное обеспечение:

Операционная система: Windows 7 или Windows 8  
Open Office  
Компьютерные программы: Scratch

## Комплекс организационно-педагогических условий

Занятия проводит педагог дополнительного образования Волков В.А., образование высшее, пройдены курсы повышения квалификации:

- федеральное государственное автономное учреждение «Фонд новых форм развития образования» «Гибкие компетенции проектной деятельности»;  
ГОУДПО «КРИРО» «Методика преподавания основ робототехники». Удостоверение 11 348688, р/н 714

## 2.3. Формы аттестации/контроля

Знания, умения, навыки, полученные на занятиях, необходимо подвергать педагогическому контролю, с целью выявления качества усвоенных детьми знаний в рамках программы обучения.

### Аттестация учащихся:

- начальная аттестация (сентябрь);
- промежуточная аттестация (декабрь);
- промежуточная аттестация (май).

При наборе учащихся в объединение по интересам проводится начальная аттестация, в ходе которой педагог проводит устный опрос и практическая работа, по результатам которого узнает уровень подготовки учащихся к занятиям.

Формы промежуточной аттестации: теоретическая часть – письменный опрос, практическая часть - практическая работа.

Письменный опрос состоит из перечня вопросов по содержанию разделов программы, каждому из учащихся предлагается ответить письменно на 7 вопросов. Практическая работа предполагает задания по пройденному материалу.

### Формы аттестации учащихся в течение учебного года

Аттестация	Сроки	Теория	Практика
Начальная аттестация	сентябрь	устный опрос	практическая работа
Промежуточная	декабрь	письменный опрос	практическая работа
Промежуточная	май	письменный опрос	практическая работа

### 2.4. Оценочные материалы

Оценка теоретических знаний и практических умений и навыков учащихся по теории и практике проходит по трем уровням: высокий, средний, низкий.

**Высокий уровень** – учащиеся должны знать правила техники безопасности при работе, грамотно излагать программный материал, знать основные блоки команд, уметь выполнять алгоритмы, описанные с использованием конструкций ветвления и повторения и уметь самостоятельно создавать и выполнять программы для решения алгоритмических задач в программе Scratch.

**Средний уровень** – учащиеся должны знать основные блоки команд, уметь выполнять алгоритмы, описанные с использованием конструкций ветвления и повторения, грамотно и по существу излагать программный материал, не допуская существенных неточностей в ответе.

**Низкий уровень** – учащиеся не знают значительной части материала, допускают существенные ошибки, с большими затруднениями выполняют практические задания.

При обработке результатов учитываются **критерии** для выставления уровней:

**Высокий уровень** – выполнение 100% - 70% заданий;

**Средний уровень** – выполнение от 50% до 70% заданий;

**Низкий уровень** - выполнение менее 50% заданий.

### 2.5. Методические материалы

Для реализации программы используются следующие методы обучения:

- по источнику полученных знаний: словесные, наглядные, практические.
- по способу организации познавательной деятельности:
  - развивающее обучение (проблемный, проектный, творческий, частично-поисковый, исследовательский, программированный);
  - дифференцированное обучение (уровневые, индивидуальные задания).
  - игровые методы (конкурсы, игры-конструкторы, турниры с использованием мультимедиа, дидактические).

Средства обучения:

- дидактические материалы (опорные конспекты, проекты примеры, раздаточный материал для практических работ).
- методические разработки (презентации, видеоуроки, flash-ролики).
- сетевые ресурсы Scratch.
- видеохостинг Youtub (видеоуроки «работа в среде Scratch»).
- учебно-тематический план.

### Список литературы:

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. Дженжер, Л.В. Денисова.
3. «Раннее обучение программированию в среде Scratch», В.Г. Рындак, В.О. Дженжер, Л.В. Денисова.
4. Голиков Д.И. «Scratch для юных программистов», «БХВ-Петербург», Санкт-Петербург, 2017.