

**ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ АДМИНИСТРАЦИИ г.о. Самара**  
Муниципальное бюджетное общеобразовательное учреждение  
«Школа № 127» городского округа Самара

«Принято»  
на заседании методического совета

Протокол № 1  
от «26» августа 2022 г.

«Утверждаю»  
Директор МБОУ Школа № 127  
г.о. Самара  
А.В. Лебедев  
Приказ № 309/1  
«28» августа 2022 г.



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА  
«ПРОГРАМИРОВАНИЕ СО SCRATCH»**

Направленность: техническая  
Уровень программы: базовая  
Возраст учащихся: 12-14 лет  
Срок реализации: 3 года

Разработчик программы:  
учитель информатики  
Цалевич А.М.

Самара, 2022 год

## **Введение**

Современное поколение детей не представляет себе досуг без компьютерных игр. Хотят взрослые или нет, но индустрия компьютерных игр будет расти и развиваться, так как есть спрос. А значит людей, входящих в мир компьютерных игр, будет все больше.

Психологи на протяжении многих лет пытаются ответить на вопрос, что больше дают компьютерные игры – пользы или вреда, но однозначно на этот вопрос ответить не могут. У виртуального мира, как и у любой медали, две стороны.

Есть данные, утверждающие, что у юных геймеров практически такая же координация и концентрация внимания, как у космонавтов. Во всяком случае, эти показатели сравнимы с показателями тренированных астронавтов из НАСА. Кроме того, у таких подростков хорошо развито воображение. Многие взрослые, которые были когда-то геймерами, сейчас работают в ведущих технических компаниях мира. В то же самое время, не секрет, что погружение в виртуальный мир, побежденные чудища и гоблины, воскрешение после неудачного боя и т.д. искажает восприятие реального мира.

Нет смысла уводить детей от компьютерных игр, вызывая тем самым ещё большее желание получить запретный плод. Есть смысл предлагать ребёнку компьютерные игры согласно его возрасту, а ещё лучше попробовать его заинтересовать вопросом, как появляются компьютерные игры и сможет ли он сам создать свой виртуальный мир. Именно это мы и попытались сделать в своем курсе, предложив учащимся научиться создавать свои компьютерные игры в среде программирования СКРЕТЧ.

СКРЕТЧ – это визуальная объектно-ориентированная среда программирования для обучения школьников младших и средних классов. Название данной среды произошло от слова **scratching** - техники, используемой хип-хоп-ди-джеями, которые крутят виниловые пластинки взад-вперед руками для того, чтобы смешивать музыкальные темы. Скретч был создан как продолжение идей языка Лого и конструктора Лего. Эта среда позволяет детям создавать свои собственные анимированные интерактивные истории, игры и модели. В среде программирования Сретч можно играть с различными объектами – СПРАЙТАМИ, видоизменять их вид, перемещать по экрану, устанавливать формы взаимодействия между объектами. Это объектно-ориентированная среда, в которой блоки программ собираются из разноцветных кирпичиков команд.

Из этих кирпичиков даже самые маленькие дети, имеющие представление о компьютере и умеющие работать с мышью, могут собрать простейшие конструкции. Учащиеся могут собирать из разноцветных блоков свои программы и процедуры. В результате выполнения простых команд со спрайтами может складываться сложная модель, в которой будут взаимодействовать множество объектов, наделенных различными свойствами. Начальный уровень программирования очень прост и доступен, поэтому в данной среде можно начинать обучение программированию с младшими школьниками<sup>1</sup>.

Когда учащиеся создают проекты в Скретче, они осваивают множество навыков 21 века: творческое мышление, предметное общение, системный анализ, беглое использование технологий, эффективное взаимодействие, проектирование, постоянное обучение и т.д. Также изучение Скретч может серьезно помочь учащимся освоить азы алгоритмизации и программирования, а полученные знания пригодятся для дальнейшего и более серьезного изучения программирования и успешной сдачи государственной итоговой аттестации по информатике.

## Пояснительная записка

Дополнительная общеобразовательная (общеразвивающая) программа «Программирование по Scratch» разрабатывалась на основе следующих материалов и документов: Рындак В. Г., Дженжер В. О., Денисова Л. В. Проектная деятельность школьника в среде программирования Scratch. — Оренбург: Оренб. гос. ин-т. менеджмента, 2009; «Пропедевтика идей параллельного программирования в средней школе при помощи среды Scratch», В.Г. Рындак, В.О. Джинжер, Л.В. Денисова; «Ранее обучение программирование в среде Scratch», а так же в соответствии с требованиями Федерального государственного образовательного стандарта второго поколения основного общего образования. Главная цель изучения курса - формирование всесторонне образованной личности, умеющей ставить цели, организовывать свою деятельность, оценивать результаты своего труда, применять знания в жизни.

Содержание курса построено таким образом, что изучение всех последующих тем обеспечивается знаниями по ранее изученным темам курса. Предполагаемая методика изучения и структура программы позволяют наиболее эффективно организовать учебный процесс, в том числе и обобщающее повторение учебного материала. В процессе занятий вводятся новые методы решения, но вместе с тем повторяются, углубляются и закрепляются знания, полученные ранее, развиваются умения применять эти знания на практике в процессе самостоятельной работы.

Программа позволяет учащимся осуществлять различные виды проектной деятельности, оценивать свои потребности и возможности и **сделать обоснованный выбор профиля обучения** в старшей школе.

Программа курса «Программирование по Scratch» содержит все необходимые разделы и соответствует современным требованиям, предъявляемым к программам внеурочной деятельности. Данный курс можно разделить на два блока: первый блок – знакомство со средой Скретч, создание спрайтов и фонов, добавление звуков, управление несколькими объектами, интерактивность, условия и переменные, диалог с программой; второй блок – циклические алгоритмические структуры, координаты в Скретч, создание первых простых мультфильмов и игр, функции в Скретч. Это разбиение достаточно условное, так как много зависит от математической подготовки и заинтересованности учащихся.

**Актуальность** данного курса заключается в том, что в нем четко прослеживается интеграция информатики с математикой. Школьники на занятиях по программированию в Скретч знакомятся с такими математическими понятиями как числовая прямая, положительные и отрицательные числа, координатная плоскость, координаты точки на плоскости, угол, градусная мера угла. Помимо этого, программирование и алгоритмизация способствует формированию и развитию логико-алгоритмического мышления, а значит, ребенокучится мыслить и рассуждать и как следствие будет успешен при изучении математики.

Изучение данной программы способствует формированию навыка *computational thinking* (математического мышления), который будет полезен для совершенно различных профессий. Данный курс рассчитан на освоение начальных знаний об объектно-ориентированном программировании, причем содержание заданий носит практический характер.

Изучение основ программирования связано с целым рядом умений и навыков (организация деятельности, планирование ее и т.д.), которые по праву носят общеинтеллектуальный характер и формирование которых - **одна из приоритетных задач современной школы в рамках реализации концепции развития математического образования.**

На занятиях у обучающихся формируются важные для практико-ориентированной деятельности умения, связанные с представлением, анализом и интерпретацией данных. Содержание заданий курса позволяет развивать и организационные умения: планировать этапы предстоящей работы, определять последовательность учебных действий, осуществлять контроль и оценку их правильности, поиск путей преодоления ошибок.

Общеобразовательная (общеразвивающая) программа – **авторская, технической направленности.**

Программа построена таким образом, чтобы помочь учащимся заинтересоваться программированием вообще и найти ответы на вопросы, с которыми им приходится сталкиваться в повседневной жизни при работе с большим объемом информации; при решении практических и жизненных задач. Программа строится на использовании среды Scratch при обучении детей, что позволяет создавать собственные программы для решения конкретной задачи. Это является **отличительной особенностью** данной 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.

**Новизна программы** 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 в программе мультфильмы, анимацию и даже простейшие игры, делает образовательную программу по программированию практически значимой для современного учащегося, т.к. дает возможность увидеть практическое назначение алгоритмов и программ, что будет способствовать развитию интереса к профессиям, связанным с программированием.

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

**Цель общеобразовательной (общеразвивающей) программы** - воспитание творческой личности, обогащенной общетехническими знаниями и умениями, развитие индивидуальных творческих способностей, интереса к науке и технике.

**Задачи программы:**

**Обучающие:**

- 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, малой группе, коллективе;
- формировать умение демонстрировать результаты своей работы.

**Принципы обучения, реализуемые программой:**

- сознательности;
- наглядности;
- доступности;
- связи теории с практикой;
- творческой активности.

Важным условием развития творческого и познавательного интереса учащегося является индивидуальный подход к нему в процессе обучения.

**Организация образовательного процесса**

**Срок реализации общеобразовательной (общеразвивающей) программы «Программирование по Scratch» - 3 года.**

**Рекомендуемый возраст детей: 12-16 лет.**

**На программу *1-2-3 года обучения* отводится **102 часа****

**Режим занятий:**

- 1 раза в неделю по 1 часу.

**Наполняемость групп:**

- в группе *1-2-3 года обучения* – 25 человек.

В данной программе используется индивидуальная, групповая и фронтальная формы работы.

Содержание практических занятий ориентировано не только на овладение учащимися навыками программирования, но и на подготовку их как грамотных пользователей ПК; формированию навыков участия в дистанционных конкурсах и олимпиадах, умений успешно использовать навыки сетевого взаимодействия.

**Ожидаемые результаты и способы их проверки****Личностные и метапредметные результаты освоения дополнительной общеобразовательной (общеразвивающей) программы*****Личностные:***

- установление связи целью учебной деятельности и ее мотивом — определение того, -  
«какое значение, смысл имеет для меня участие в данном занятии»;
- построение системы нравственных ценностей, выделение допустимых принципов поведения;
- реализация образа Я (Я-концепции), включая самоотношение и самооценку;
- нравственно-этическое оценивание событий и действий с точки зрения моральных норм.

Построение планов во временной перспективе.

***Регулятивные:***

- определение образовательной цели, выбор пути ее достижения;
- рефлексия способов и условий действий; самоконтроль и самооценка; критичность;
- выполнение текущего контроля и оценки своей деятельности; сравнение характеристик запланированного и полученного продукта;
- оценивание результатов своей деятельности на основе заданных критериев, умение самостоятельно строить отдельные индивидуальные образовательные маршруты.

***Коммуникативные:***

- планирование учебного сотрудничества с учителем и сверстниками — определение цели, способов взаимодействия;
- контроль и оценка своей деятельности, обращение по необходимости за помощью к сверстникам и взрослым;
- формирование умения коллективного взаимодействия.

### ***Познавательные:***

- умение актуализировать знания, определять границы своего знания при решении задач практического содержания;
- умение оперировать со знакомой информацией; формировать обобщенный способ действия; моделировать задачу и ее условия, оценивать и корректировать результаты решения задачи.

Изучение курса дает возможность обучающимся достичь следующих результатов развития:

#### *1) в личностном направлении:*

умение ясно, точно, грамотно излагать свои мысли в устной и письменной речи, понимать смысл поставленной задачи;

умение распознавать логически некорректные высказывания, отличать гипотезу от

факта;

креативность мышления, инициатива, находчивость, активность, применение знаний по

информатике и математике для решения конкретных жизненных задач;

#### *2) в метапредметном направлении:*

умение находить в различных источниках информацию, необходимую для решения проблем;

умение понимать и использовать различные средства наглядности (графики, диаграммы, таблицы, схемы и др.);

умение применять индуктивные и дедуктивные способы рассуждений;

умение планировать и осуществлять деятельность, направленную на решение задач исследовательского характера;

#### *3) в предметном направлении:*

умение грамотно применять символику, использовать язык программирования СКРЕТЧ для записи программ, алгоритмический язык для постановки задачи;

овладение знаниями об объектно-ориентированном программировании;

овладение основными способами представления и анализа данных;

умение использовать язык программирования для описания действий объектов программирования, развитие пространственных представлений и изобразительных умений;

овладение навыками быстрого использования технологий;

умение применять изученные понятия, результаты, методы для решения задач практического характера и задач из смежных дисциплин с использованием при необходимости справочных материалов, компьютера.

### **Система оценки достижений учащихся**

Для отслеживания результатов предусматриваются в следующие *формы контроля:*

- *стартовый*, позволяющий определить исходный уровень знаний и умений учащихся;



- *текущий*: - прогностический, то есть проигрывание всех операций учебного действия доначала его реального выполнения;
- *плановый*, то есть контроль за правильностью, полнотой и последовательностью выполнения операций, входящих в состав действия;
- рефлексивный, контроль, обращенный на ориентировочную основу, «план» действия и опирающийся на понимание принципов его построения;
- контроль по результату, который проводится после осуществления учебного действия методом сравнения фактических результатов или выполненных операций с образцом;
- *итоговый* контроль в формах: практические работы, творческие работы учащихся, создание проекта.

Курс «Программирование по Scratch» предполагает без отметочную форму оценивания. Основную роль должна играть словесная оценка товарищей и учителя. При оценивании проектов допускается оценка в баллах, которая заносится в паспорт проекта. Возможна оценка в виде рецензии или отзыва на выполненную работу (вместе с её фотографией, скриншотом), которую следует помещать в портфолио учащегося. Предусматривается самооценка и самоконтроль учащихся - определение учеником границ своего «знания - незнания», своих потенциальных возможностей, а также осознание тех проблем, которые ещё предстоит решить в ходе осуществления деятельности.

Содержательный контроль и оценка результатов учащихся предусматривает выявление индивидуальной динамики качества усвоения предмета ребёнком и не допускает сравнения его с другими детьми. Результаты фиксируются в зачётном листе учителя.

Для оценки эффективности занятий можно использовать следующие показатели:

- степень помощи, которую оказывает учитель учащимся при выполнении заданий: чем помощь учителя меньше, тем выше самостоятельность учеников и, следовательно, выше развивающий эффект занятий;
- поведение учащихся на занятиях: живость, активность, заинтересованность школьников обеспечивают положительные результаты занятий;
- результаты выполнения тестовых заданий и заданий из конкурса эрудитов, при выполнении которых выявляется, справляются ли ученики с этими заданиями самостоятельно;
- косвенным показателем эффективности данных занятий может быть повышение успеваемости по разным школьным дисциплинам, а также наблюдения учителей за работой учащихся на других уроках (повышение активности, работоспособности, внимательности, улучшение мыслительной деятельности).

### **Проверка результативности**

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

Формами педагогического контроля могут быть: итоговые занятия один раз в полугодие, контрольные задания, тематические выставки, устный опрос, тестирование, которые способствуют поддержанию интереса к работе, направляют учащихся к достижению более высоких вершин творчества.

### Учебно - тематический план 1-2-3 год обучения

№ п/п	Разделы программы и темы учебных занятий	Количество часов		
		теория	практика	всего
	<b>1 год обучения</b>			
1.	Знакомство со средой СКРЕТЧ	2	2	4
2.	Начало программирования. Первые программы на Скретч	2	12	14
3.	Графические и звуковые редакторы Скретч. Первый мультик	2	14	16
	<b>ИТОГО 1 год обучения</b>	<b>6</b>	<b>28</b>	<b>34</b>
	<b>2 год обучения</b>			
1.	Знакомство со средой СКРЕТЧ	1	1	2
2.	Начало программирования. Первые программы на Скретч	0	2	2
3.	Проектная деятельность и моделирование процессов и систем	2	8	10
4.	Создание мультфильмов и игр	2	8	10
5.	Знакомство с переменными	1	5	6
6.	Итоговый годовой проект	1	3	4
	<b>ИТОГО 2 год обучения</b>	<b>7</b>	<b>27</b>	<b>34</b>
	<b>3 год обучения</b>			
1.	Начало программирования. Первые программы на Скретч	1	3	4
2.	Графические и звуковые редакторы Скретч. Программа движения	1	7	8
3.	Проектная деятельность и моделирование процессов и систем	2	8	10
4.	Создание мультфильмов и игр	2	8	10
5.	Итоговый годовой проект	1	3	4
	<b>ИТОГО 3 год обучения</b>	<b>7</b>	<b>27</b>	<b>34</b>
	<b>ИТОГО</b>	<b>20</b>	<b>82</b>	<b>102</b>

## Содержание программы 1-2-3 год обучения

### 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 копирование 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.

## **Раздел 5. Создание мультфильмов и игр**

Разработка моделей игр и мультфильмов на основе изученного материала

*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 6. Знакомство с переменными**

Назначение переменных. Создание переменных. Использование переменных для создания игры

*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 7. Итоговый годовой проект**

Разработка плана игры или видео по заданной теме. Создание программного кода для спрайтов.

*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.

### **Методическое обеспечение программы**

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.

### **Материально-техническое обеспечение программы**

**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: Scrath

### 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.
5. Марчи М. scratch для детей. Самоучитель по программированию. М. 2016 – 154с.