

Фроловского муниципального района Волгоградской области

Принята на заседании  
педагогического совета  
№ 1 от 25.08. 2021г.



«Утверждаю»

Директор школы

Приказ № 145 от 02.09.2021

Л.В.Котельникова

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

**«Занимательное программирование»**

Направленность: техническая

Возраст: 7-8 класс

Срок реализации: 1 год (34 часа)

Автор составитель:  
Быстрова М.А.,  
учитель информатики.

п.Образцы , 2021г

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Основное курса внеурочной деятельности «Занимательное программирование» - изучение алгоритмов и исполнителей, первое знакомство с основными алгоритмическими конструкциями, используемыми в языках программирования; получение позитивного опыта отладки и написания первых завершённых программных продуктов.

Программа курса предполагает знакомство с основными понятиями, используемыми в языках программирования высокого уровня, решение большого количества творческих задач, многие из которых моделируют процессы и явления из таких предметных областей, как информатика, алгебра, геометрия, география, физика, русский язык и др. Многие задания составлены таким образом, чтобы они решались методами учебно-исследовательской и проектной деятельности. Большинство заданий встречаются в разных темах для того, чтобы показать возможности решения одной и той же задачи или проблемы различными средствами, обеспечивающими достижение требуемого результата, что в итоге приведет к способности выбирать оптимальное решение данной задачи или проблемы.

**Цель:** сформировать у учащихся базовые представления о языках программирования, алгоритме, исполнителе, способах записи алгоритма; организовать проектную научно-познавательную деятельность творческого характера; сформировать у школьника познавательный интерес к учёбе и исследовательские навыки.

Занятия направлены на решение следующих задач.

### **Задачи программы:**

#### **Обучающие:**

- ✓ овладеть навыками составления алгоритмов;
- ✓ овладеть понятиями «объект», «событие», «управление», «обработка событий»;
- ✓ изучить функциональность работы основных алгоритмических конструкций;
- ✓ сформировать представление о профессии «программист»;
- ✓ сформировать навыки разработки, тестирования и отладки несложных 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.

## **Общая характеристика курса**

Мы живем в век информатизации общества. Информационные технологии проникают в нашу жизнь с разных сторон. Одно из самых удивительных и увлекательных занятий настоящего времени - программирование.

Обучение основам программирования школьников должно осуществляться на специальном языке программирования, который будет понятен детям, будет легок для освоения и соответствовать современным направлениям в программировании.

В данном курсе предполагается вести изучение программирования в игровой, увлекательной форме.

Scratch – это объектно-ориентированная среда, в которой блоки программ собираются из разноцветных кирпичиков.

Scratch – это мультимедийная система. Большая часть операторов языка направлена на работу с графикой и звуком, создание анимационных и видеоэффектов. Манипуляции с медиаинформацией – главная цель создания Scratch.

## **Основные особенности Scratch**

**Блочное программирование.** Для создание программ в Scratch, вы просто совмещаете графические блоки вместе в стеках. Блоки сделаны так, чтобы их можно было собрать только в синтаксически верных конструкциях, что исключает ошибки. Различные типы данных имеют разные формы, подчеркивая несовместимость. Вы можете сделать изменения в стеках, даже когда программа запущена, что позволяет больше экспериментировать с новыми идеями снова и снова.

**Манипуляции данными.** Со Scratch вы можете создать программы, которые управляют и смешивают графику, анимацию, музыку и звуки. Scratch расширяет возможности управления визуальными данными, которые популярны в сегодняшней культуре — например, добавляя программируемость, похожих на Photoshop фильтров.

**Совместная работа и обмен.** Сайт проекта Scratch предлагает вдохновение и аудиторию: вы можете посмотреть проекты других людей, использовать и изменить их картинки и скрипты, и добавить ваш собственный проект. Самое большое достижение — это общая среда и культура, созданная вокруг самого проекта.

Scratch предлагает низкий пол — легко начать; высокий потолок -возможность создавать сложные проекты; и широкие стены — поддержка большого многообразия проектов. В работе над Scratch уделили особое внимание простоте, иногда даже в ущерб функциональности, но с целью большей доступности.

***Когда учащиеся работают над проектом в Scratch, они имеют шанс выучить важные вычислительные концепции, такие как повторения, условия, переменные, типы данных, события и процессы.*** Scratch уже был использован для представления этих понятий детям различных возрастов, от элементарной школы до колледжа. Некоторые студенты перешли от традиционных, текстовых, языков программирования, после того, как им показали программирование в Scratch.

***Scratch*** - это проект с открытым исходным кодом, но с «закрытой» командой разработчиков. Исходный код доступен в свободном доступе, но приложение разрабатывается маленькой командой ученых из MIT Media Lab.

Одной из главных концепций языка Scratch, является развитие собственных задумок с первой идеи до конечного программного продукта. Для этого в Scratch имеются все необходимые средства:

- стандартные для языков процедурного типа: следование, ветвление, циклы, переменные, типы данных (целые и вещественные числа, строки, логические, списки — динамические массивы), псевдослучайные числа;
- объектно-ориентированные: объекты (их поля и методы), передача сообщений и обработка событий;
- интерактивные: обработка взаимодействия объектов между собой, с пользователем, а также событий вне компьютера (при помощи подключаемого сенсорного блока);
- параллельное выполнение: запуск методов объектов в параллельных потоках с возможностью координации и синхронизации;
- создание простого интерфейса пользователя.

### **Актуальность и новизна 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.

Изучение языка значительно облегчает последующий переход к изучению других языков программирования. Преимуществом Scratch, среди подобных сред программирования, также является наличие версий для различных операционных систем: для Windows, Mac OS, GNU/Linux. В настоящее время имеет смысл рассматривать программы с открытым кодом, что позволяет сформировать у учащихся более широкое представление о возможностях работы с цифровой техникой.

Скретч — свободно распространяемая программа, скачать которую можно, к примеру, с сайта: [http://info.scratch.mit.edu/Scratch 1.4 Download](http://info.scratch.mit.edu/Scratch%201.4%20Download).

**Аспект новизны** заключается в том, что Scratch не просто язык программирования, а еще и интерактивная среда, где результаты действий визуализированы, что делает работу с программой понятной, интересной и увлекательной.

Особенность среды Scratch, позволяющая создавать в программе мультфильмы, анимацию и даже простейшие игры, делает образовательную программу «Увлекательное программирование» практически значимой для современного школьника, т.к. дает возможность увидеть практическое назначение алгоритмов и программ, что будет способствовать развитию интереса к профессиям, связанным с программированием.

В зависимости от поставленных задач на занятиях используются различные методы обучения. Все задания расположены по возрастанию степени их сложности. Доступная детям практическая деятельность помогает избежать умственного переутомления.

#### **Организация деятельности учащихся:**

- формы организации деятельности обучающихся:
  - ✓ групповые,
  - ✓ индивидуальные;
- методы обучения
  - ✓ словесные методы (лекция, объяснение);
  - ✓ демонстративно-наглядные (демонстрация работы в программе, схем, скриптов, таблиц);
  - ✓ проектные методы (разработка проекта по спирали творчества, моделирование, планирование деятельности)
  - ✓ работа с Интернет - сообществом (публикация проектов в Интернет - сообществе скретчеров).
  - ✓ объяснительно – иллюстративные (методы обучения, при использовании которых, дети воспринимают и усваивают готовую информацию);
  - ✓ частично-поисковые методы обучения (участие детей в

- ✓ коллективном поиске, решение поставленной задачи совместно с педагогом). исследовательские методы обучения (овладение детьми методами научного познания, самостоятельной творческой работы).

Отличительные особенности. Данная программа обучения основана на преимуществах дополнительного образования и призвана дать необходимые знания и умения в области изучения компьютерных технологий для школьников, а также выявить способных, талантливых детей и развить их способности, дает большие возможности для творческого развития детей, предусматривая индивидуальный подход к ребенку.

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 моделирования.

Набор обучающихся – свободный, по желанию детей и их родителей.

Режим организации занятий:

- общее количество часов в год — 34 часа;
- количество часов в неделю 1 час.

### **Прогнозируемые результаты.**

В результате обучения учащиеся будут уметь:

- самостоятельно планировать пути достижения целей, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;
- соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата;
- оценивать правильность выполнения учебной задачи, собственные возможности ее решения;
- 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;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий.
- 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.
- на 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.

## Требования к уровню освоения материала.

В результате школьник, участвующий в проектной научно-познавательной деятельности, будет:

### **Знать:**

- ✓ отдельные способы планирования деятельности;
- ✓ составление плана предстоящего проекта в виде рисунка, схемы;
- ✓ составление плана предстоящего проекта в виде таблицы объектов, их свойств и взаимодействий;
- ✓ разбиение задачи на подзадачи;
- ✓ распределение ролей и задач в группе;

### **Уметь:**

- ✓ составить план проекта, включая: выбор темы; анализ предметной области; разбиение задачи на подзадачи; проанализировать результат и сделать выводы;
- ✓ найти и исправить ошибки;
- ✓ подготовить небольшой отчет о работе; публично выступить с докладом;
- ✓ наметить дальнейшие пути развития проекта;

### **Получат навыки:**

- ✓ работы в группе;
- ✓ ведения дискуссии;
- ✓ донесения своих мыслей до других.

### **Получат возможность:**

- ✓ постепенно учиться программированию;
- ✓ реализовать свои творческие порывы;
- ✓ участвовать в интерактивном процессе создания игр и анимирования разнообразных историй как индивидуально, так и вместе со своими сверстниками из разных стран;
- ✓ получать живой отклик от единомышленников; оценить свои творческие способности.

### **Описание места курса в учебном плане**

По учебному плану образовательного учреждения отводит 34 часа. Курс «Занимательное программирование» изучается за счет части часов внеурочной деятельности, из расчета 1 час в неделю в 5-7 классах.

## РЕЗУЛЬТАТЫ ОСВОЕНИЯ КУРСА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ

### **Личностные:**

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

результату его деятельности;

- развитие эстетического сознания через творческую деятельность на базе иллюстрированной среды программирования.

#### **Метапредметные:**

- умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;

- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата, понимая, что в программировании длинная программа не значит лучшая программа;

- умение оценивать правильность решения учебно-исследовательской задачи;

- умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;

- владение основами самоконтроля, принятия решений;

- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;

- ИКТ-компетенцию;

- умение сотрудничества и совместной деятельности со сверстниками в процессе проектной и учебно-исследовательской деятельности.

#### **Предметные:**

- осознание значения математики и информатики в повседневной жизни человека;

- формирование представлений об основных предметных понятиях — «информация», «алгоритм», «модель» и их свойствах;

- развитие логических способностей и алгоритмического мышления, умений составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами — линейной, условной и циклической;

- развитие представлений о числах, числовых системах;

- овладение символьным языком алгебры, умение составлять и использовать сложные алгебраические выражения для моделирования учебных проектов, моделировать реальные ситуации на языке алгебры;

- развитие пространственных представлений, навыков геометрических построений и моделирования таких процессов, развитие изобразительных умений с помощью средств ИКТ;

- формирование информационной и алгоритмической культуры, развитие основных навыков использования компьютерных устройств и программ;

- формирование умения соблюдать нормы информационной этики и права.

**В результате освоения курса «Занимательное программирование» школьники получают представление о:**

- свободно распространяемых программах;

- функциональном устройстве программной среды Scratch и основных структурных элементах пользовательского интерфейса;

- назначении и использовании основных блоков команд, состояний, программ;

- правилах сохранения документа и необходимости присвоения правильного имени;

- возможности и способах отладки написанной программы;

- сущности понятий «спрайт», «сцена», «скрипт»;

- исполнителях и системах их команд, возможности непосредственного управления исполнителем;
- наличии заготовок для персонажей и сцен в соответствующих библиотеках, иерархическом устройстве библиотек и возможности импортирования их элементов;
- возможности использования встроенного растрового редактора, наличии и назначении основных инструментов;
- использовании других программ (например, LibreOfficeDraw) для создания собственных изображений;
- алгоритме как формальном описании последовательности действий исполнителя, приводящих от исходных данных к конечному результату;
- использовании схематического описания алгоритма;
- программном управлении исполнителем и линейных алгоритмах;
- написании программ для исполнителей, создающих геометрические фигуры на экране в процессе своего перемещения;
- необходимости программного прерывания;
- использовании циклических команд при необходимости повторений однотипных действий;
- видах циклических алгоритмов и их применении;
- достижении 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;
- видах и формах разветвленных алгоритмов, включая циклы с условием;
- управлении событиями.
- использовании метода проектов для моделирования объектов и систем;
- возможности описания реальных задач средствами программной среды;
- создании анимационных, игровых, обучающих проектов, а также систем тестирования в программной среде Scratch.

### **Школьники будут 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;



# СОДЕРЖАНИЕ КУРСА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ С УКАЗАНИЕМ ФОРМ ОРГАНИЗАЦИИ И ВИДОВ ДЕЯТЕЛЬНОСТИ

## 1. Знакомство со средой Scratch.

*Теоретические знания и умения:*

Знакомство со средой программирования Scratch. Интерфейс Scratch. Главное меню Scratch. Понятие алгоритма, способы записи алгоритма. Понятие спрайта и объекта. Коллекции спрайтов и фонов. Использование интернета для импорта объектов. Особенности графического редактора среды Scratch.

*Практическая работа:*

Установка программы Scratch, составление алгоритмов. Изучение интерфейса среды. Создание и редактирование спрайтов и фонов для сцены, создание новых спрайтов и сцен. Поиск, импорт и редактирование спрайтов из интернета. Сохранение и открытие проектов.

## 2. Управление спрайтами.

*Теоретические знания и умения:*

Команды и блоки. Программные единицы: процедуры и скрипты. Навигация в среде Scratch. Система координат. Оси координат. Команды управления. Градусная мера угла. Средства рисования группы перо. Знакомство с правильными геометрическими фигурами.

*Практическая работа:*

Определение координат спрайта. Ориентация по координатам. Проект «Робот рисует», «Кругосветное путешествие Магелана».

## 3. Циклы.

*Теоретические знания и умения:*

Понятие цикла. Команда **повторить**. Конструкция **всегда**. Управление курсором движения. Команда **повернуть в направлении**.

*Практическая работа:*

Рисование узоров и орнаментов. Проект «берегись автомобиля». Проект «Гонки по вертикали». Проект «Полет самолета».

## 4. Создание анимационных проектов.

*Теоретические знания и умения:*

Анимация. Этапы создания проекта. Смена костюмов.

*Практическая работа:*

Проект «Осьминог». Проект «Девочка, прыгающая на скакалке». Проект «Бегущий человек». Создание мультипликационного сюжета «Кот и птичка».

## 5. Разветвляющиеся алгоритмы.

*Теоретические знания и умения:*

Знакомства с командами ветвления. . Понятие условия. Простые и составные условия. Сенсоры. Блок **если**. Выполнение скриптов с ветвлением. Вложенные команды ветвления.

*Практическая работа:*

Проект «Осторожно лужи». Проект «Управляемый робот».

Игры «Лабиринт», «Опасный лабиринт». Проект «Хождение по коридору». Проект «Слепой кот». Проект «Тренажер памяти».

## **6. Случайности по заказу.**

*Теоретические знания и умения:*

Датчик случайных чисел, сенсоры.

*Практическая работа:*

Проект «Кошки-мышки». Проект «Цветок». Проект «Таймер»

## **7. Запуск и самоуправление спрайтов.**

*Теоретические знания и умения:*

Запуск спрайтов с помощью клавиатуры. Запуск спрайтов с помощью мыши.

Передача сигналов. Датчики.

*Практическая работа:*

Проект «Переродевайка». Проект «Дюймовочка». Проект «Лампа». Проект «Презентация»

## **8. Переменные и рычажки.**

*Теоретические знания и умения:*

Переменные. Типы переменных. Рычажки.

*Практическая работа:*

Игра «Голодный кот». Проект «Цветы». Проект «Правильные многоугольники»

## **9. Списки.**

*Теоретические знания и умения:*

Понятие списка. Элементы списков. Создание списка. Добавление и удаление элементов в список.

*Практическая работа:*

Проект «Гадание». Проект «Назойливый собеседник». Проект «интерактивный собеседник». »Проект «Тест». Игра «Угадай слово».

## **10. Музыка чисел.**

*Теоретические знания и умения:*

Знакомство с музыкальными возможностями Scratch. Громкость. Тон. Тембр. Запись звука. Форматы звуковых файлов. Озвучивание проектов Scratch.

*Практическая работа:*

Проект « музыкальный инструмент». Проект «Композитор»

## **11. Свободное проектирование.**

*Теоретические знания и умения:*

Знакомство с этапами проектирования. Составление таблицы объектов, их свойства и взаимодействие. Создаём собственную игру, мультфильм. Хранилище проектов.

*Практическая работа:*

Включение в деятельность обсуждения проектов. Публичные презентаций результатов деятельности

## ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№ п/п	Тема	Количество часов		
		Теория	Практика	Всего
1	Знакомство с программной средой Scratch	1	1	2
2	Компьютерная графика	2	3	5
3	Алгоритмы и исполнители	4	10	14
4	Проектная деятельность и моделирование процессов и систем	3	10	13
	<b>ИТОГО:</b>	<b>10</b>	<b>24</b>	<b>34</b>

### Поурочное планирование курса «Занимательное программирование»

№ урока	Тема урока	Количество часов
1.	Знакомство со средой Scratch. Внешний вид среды, поля. Анимация.	1
2.	Исполнитель Scratch, цвет и размер пера.	1
3.	Основные инструменты встроенного растрового графического редактора.	1
4.	Линейный алгоритм. Создание блок-схемы. Основные графические примитивы векторного редактора LibreOfficeDraw.	1
5.	Линейный алгоритм. Рисование линий исполнителем Scratch.	1
6.	Линейный алгоритм. Исполнитель Scratch рисует квадраты и прямоугольники линейно.	1
7.	Конечный цикл. Исполнитель Scratch рисует квадраты, линии.	1
8.	Конечный цикл. Исполнитель Scratch рисует несколько линий и фигур. Копирование фрагментов программы.	1
9.	Циклический алгоритм. Цикл в цикле. Вложенные и внешние циклы.	1
10.	Цикл в цикле. Повторение пунктирной линии с поворотом. Блок-схема цикла.	1
11.	Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов.	1
12.	Сцена как исполнитель. Создаем модель таймера.	1
13.	Бесконечный цикл. Одна программа для исполнителя Scratch, но разные костюмы.	1
14.	Одинаковые программы для несколько исполнителей.	1
15.	Несколько исполнителей. Параллельное выполнение действий для ускорения процесса выполнения программы.	1
16.	Разбиение программы на части для параллельного выполнения исполнителями. Таймер. Уменьшение показаний таймера при параллельных вычислениях.	1
17.	Два исполнителя со своими программами. Мини-проект «Часы».	1
18.	Алгоритмы с ветвлением. Условие ЕСЛИ. Два исполнителя.	1

№ урока	Тема урока	Количество часов
19.	Цикл при условии. Мини-проект «Шарики в лабиринте»	1
20.	Цикл при условии. Исполнитель определяет цвета.	1
21.	Цикл при условии. Исполнители в разных слоях. Мини-проект «Самолет сквозь облака».	1
22.	Перемещение исполнителя из одного слоя в другой. Действия исполнителей в разных слоях. Мини-проект «Дорога».	1
23.	Алгоритмы с ветвлением. Условие ЕСЛИ. Взаимодействие исполнителей. Блок-схема с условием.	1
24.	Сцена как исполнитель. Последовательное выполнение команд исполнителями.	1
25.	Алгоритмы с ветвлением. Программирование клавиш.	1
26.	Алгоритмы с ветвлением. Если касается цвета.	1
27.	Интерактивность исполнителей. Создание мини-проекта «Лабиринт».	1
28.	Игра «Лабиринт». Усложнение.	1
29.	Моделирование ситуации. Мини-проект «Пешеходный переход».	1
30.	Моделирование ситуации. Интерактивность исполнителей. Мини-проект «Водолей».	1
31.	Моделирование. Учебные модели «Рисующий карандаш», «Затухание».	1
32.	Моделирование. Тестовая модель «Комнатные растения».	1
33.	Моделирование. Обучающий проект по маршрутам географических открытий.	1
34.	Итоговый проект	1

## **Материально-техническая база**

Оборудование центра "Точка роста"

### **Список литературы**

1. Сорокина Т.Е. Развитие алгоритмического мышления школьников с использованием среды программирования SCRATCH: Мат. Конф./Междунар. научно-практич. конф. 1 апреля 2013 г. в 6 частях. Часть III. Мин-во обр и науки. М.: АР-Консалт, 2013. С. 39–40.
2. Т.Е. Сорокина, поурочные разработки «Пропедевтика программирования со Scratch» для 5-го класса, 2015 г.
3. Голиков Д.В. “Scratch для юных программистов”, ВHV, 2017 г. Стр.192.
4. Голиков Д.В. “40 проектов на Scratch для юных программистов”. ВHV, 2018 г.

### **Литература для обучающихся**

1. Пашковская Ю.В. Творческие задания в среде Scratch. 5-6 класс. Рабочая тетрадь. – М.: Изд-во «БИНОМ. Лаборатория знаний», 2014. – 200 с.

### **Ресурсы в Интернете**

1. Scratch - среда визуального программирования <https://scratch.mit.edu/>
2. Голиков Д. В. и Голиков А. Д. Книга юных программистов на Scratch. Электронная книга, 2013г. — 134 с.: ил. Режим доступа: <http://scratch4russia.com/store/#!/Книга-юных-программистов-на-Scratch-1-4/c/11294030/>
3. Голиков Д. В. и Голиков А. Д. Программирование на Scratch 2. Делаем игры и мультики. Подробное пошаговое руководство для самостоятельного изучения ребёнком. Электронная книга, 2014. — 295 с.: ил. Режим доступа: [http://scratch4russia.com/store/#!/Программирование-на-Scratch-2-Часть-1 - Формат-PDF/p/43882754/category=11294026](http://scratch4russia.com/store/#!/Программирование-на-Scratch-2-Часть-1-Формат-PDF/p/43882754/category=11294026)
4. Сорокина Т.Е. Пропедевтика программирования со Scratch: Слово учителю.