МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ   
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ

ГОУ ДПО «Донецкий республиканский институт

дополнительного педагогического образования»

программа

курса по выбору

*«УВЛЕКАТЕЛЬНОЕ ПРОГРАММИРОВАНИЕ НА SCRATCH»*

*5-7 классы*

*Для образовательных организаций, осуществляющих образовательную деятельность на освобожденных территориях Донецкой Народной Республики*

*Рекомендовано*

*научно-методическим советом   
ГОУ ДПО «Донецкий РИДПО»*

*(протокол № 5 от 22.03.2022 г.)*

Донецк

2022

СОДЕРЖАНИЕ

[**ВВЕДЕНИЕ 4**](#_Toc48206020)

[**I. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА 4**](#_Toc48206021)

[**II. ОБОСНОВАНИЕ НЕОБХОДИМОСТИ РЕАЛИЗАЦИИ ПРОГРАММЫ 5**](#_Toc48206022)

[**III. ПРИНЦИПЫ РЕАЛИЗАЦИИ ПРОГРАММЫ 6**](#_Toc48206023)

[**IV. ЦЕЛИ И ЗАДАЧИ ПРОГРАММЫ 7**](#_Toc48206024)

[**VI. Условия реализации программы 9**](#_Toc48206025)

[**VII. Планируемые результаты изучения учебного курса 9**](#_Toc48206026)

[**VIII. СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА И ТРЕБОВАНИЯ К УЧЕБНЫМ ДОСТИЖЕНИЯМ 12**](#_Toc48206027)

[**IX. Тематическое планирование 15**](#_Toc48206028)

[**X. Обязательные формы контроля и критерии оценивания 18**](#_Toc48206029)

[**XI. Методическое, учебно-методическое и материально-техническое обеспечение образовательного процесса 20**](#_Toc48206030)

[**СПИСОК РЕКОМЕНДОВАННОЙ ЛИТЕРАТУРЫ 21**](#_Toc48206031)

# ВВЕДЕНИЕ

Программа «Увлекательное программирование на Scratch» разработана для организации курсов по выбору, элективных, факультативных, профильных курсов, кружковой работы общеинтеллектуальной направленности в 5-7 классах и расширения знаний по темам, формирующим базовые концепции программирования в общем курсе «Информатика». В основу примерной программы положено изучение **языка программирования Scratch**.

Данная программа отвечает требованиям Государственного образовательного стандарта основного общего образования (далее – ГОС ООО), обеспечивает формирование личностных, метапредметных и предметных результатов, предопределяющих дальнейшее успешное обучение в старшей школе. Программа соответствует стратегической линии развития общего образования и имеет все основания для широкого использования в преподавании предмета «Информатика» в школе.

**Программа курса по выбору содержит**:

* пояснительную записку;
* основное содержание курса с указанием форм организации и видов деятельности с примерным распределением учебных часов;
* требования к результатам освоения программы в системе требований ГОС ООО к результатам освоения курса;
* рекомендуемое тематическое планирование;
* рекомендации по материально-техническому обеспечению, учебно-методическому и материально-техническому обеспечению образовательного процесса.

# I. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Примерная программа курса по выбору «Увлекательное программирование на Scratch» разработана в соответствии с требованиями Государственного образовательного стандарта основного общего образования Донецкой Народной Республики, утвержденного приказом Министерства образования и науки Донецкой Народной Республики от 07.08.2020 г. № 120-НП (в ред. приказа Министерства образования и науки ДНР от 23.06.2021 №79-НП) и направлена на реализацию требований предметной области «Математика и информатика» и требований к общеобразовательной подготовке обучающихся по предмету «Информатика».

Примерная программа «Увлекательное программирование на Scratch» направлена на удовлетворение потребностей обучающихся в интеллектуальном совершенствовании, учитывает индивидуальные особенности детей, обеспечивает поддержку каждого ребенка, его интеллектуальное, творческое, эстетическое развитие с использованием новейших достижений современной науки.

Данный курс посвящён знакомству обучающихся 5–7-х классов с основными алгоритмическими конструкциями, развитию логических способностей школьников на базе среды программирования **SCRATCH** (<http://scratch.mit.edu/>).

**Scratch** – это творческая среда, разработанная специально для развития мышления, творческих и исследовательских способностей детей и подростков.

Название «**Scratch**» в переводе с английского имеет несколько значений. Это и царапина, которую оставляет Котёнок – символ программы, и каракули, символизирующие первый, ещё неуклюжий самостоятельный опыт, и линия старта. Со **Scratch** удобно начинать изучение азов программирования. Сами разработчики характеризуют программу так: «**Scratch** предлагает низкий пол (легко начинать), высокий потолок (возможность создавать сложные проекты) и широкие стены (поддержка большого многообразия проектов)».

Подобно тому, как дети только-только начинающие говорить, учатся складывать из отдельных слов фразы, так **Scratch** обучает из отдельных кирпичиков-команд собирать целые программы.

Программа **Scratch** имеет понятный интерфейс, встроенный графический редактор, меню готовых программ (кирпичиков), широкие возможности работы с мультимедийными объектами.

**Scratch** приятен «на ощупь». Его блоки, легко соединяемые друг с другом и так же легко, если надо, разбираемые. Они могут многократно растягиваться и снова ужиматься. **Scratch** **зовёт к экспериментам!**

В **Scratch** можно сочинять истории, рисовать и оживлять на экране придуманные персонажи, создавать презентации, игры, в том числе и интерактивные, исследовать параметрические зависимости.

Любой персонаж в среде **Scratch** может выполнять параллельно несколько действий – двигаться, поворачиваться, изменять цвет, форму и т.д.; благодаря чему юные скретчисты учатся осмысливать 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.

В **Scratch** легко осуществлять интеграцию с другими школьными науками. Так возникают межпредметные проекты. Они помогут сделать наглядными понятия отрицательных чисел и координат, уравнения плоских фигур, изучаемых на уроках математики. В них оживут исторические события и географические карты.

**Scratch** – свободно распространяемая программа. Она одинаково хорошо устанавливается и в Windows, и в Ubuntu, и в Macintosh.

# II. ОБОСНОВАНИЕ НЕОБХОДИМОСТИ РЕАЛИЗАЦИИ ПРОГРАММЫ

Курс построен таким образом, чтобы помочь обучающимся 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 «Увлекательное программирование на Scratch» позволяет создавать собственные программы для решения конкретной задачи. Это является отличительной особенностью данной программы.

Актуальность проектной деятельности сегодня осознаётся всеми. ГОС ООО нового поколения требует использования в образовательном процессе технологий деятельного типа, методы проектно-исследовательской деятельности определены как одно из условий реализации основной образовательной программы образования.

Следует иметь в виду, что возрастные особенности школьника среднего возраста не позволяют в полной мере реализовать проведение полноценных научных исследований. Раннее включение в организованную специальным образом проектную деятельность творческого характера позволяет сформировать у школьника познавательный интерес и исследовательские навыки.

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

**Педагогическая целесообразность** данной 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 активной поисковой деятельности, предоставляются широкие возможности для разнообразного программирования.

**Scratch** не просто среда для программирования, через нее можно выйти на многие другие темы предмета «Информатика». Создавая свои собственные игры и мультфильмы, дети научатся разрабатывать проекты, ставить цели и задачи. Сначала в текстовом редакторе оформляется проект, затем надо нарисовать героя, окружение. Разработать алгоритм действий героя, алгоритмы его реакций на события. Нужно озвучить героя и события (записать, обработать звук). Важно и то, что ребенок имеет возможность поделиться результатами своего творчества с друзьями или другими пользователями: кнопка для размещения созданного проекта в сети Интернет находится непосредственно в программе.

**Основные виды деятельности**: учебная, познавательная, игровая, проектная и творческая деятельность.

# III. ПРИНЦИПЫ РЕАЛИЗАЦИИ ПРОГРАММЫ

**Обучение в** **активной познавательной деятельности**. Все темы обучающиеся изучают на практике, выполняя различные творческие задания, сотрудничая в парах и группах друг с другом.

**Индивидуальное обучение**. Обучение обучающихся работе на компьютере дает возможность организовать деятельность обучающихся с индивидуальной скоростью и в индивидуальном объеме. Данный принцип реализован через организацию практикума по освоению навыков работы на компьютере.

**Принцип природосообразности**. Основной вид деятельности школьников – игра, через нее дети познают окружающий мир, поэтому в занятия включены игровые элементы, способствующие успешному освоению курса.

**Преемственность**. Программа курса построена так, что каждая последующая тема логически связана с предыдущей. Данный принцип помогает обучающимся понять важность уже изученного материала и значимость каждого отдельного занятия.

**Практико-ориентированность**, обеспечивает подбор содержания, направленного на решение простейших практических задач планирования деятельности, поиска нужной информации, инструментирования всех видов деятельности на базе общепринятых средств информационной деятельности, реализующих основные пользовательские возможности информационных технологий. При этом исходным является положение о том, что компьютер может многократно усилить возможности человека, но не заменить его.

**Принцип дидактической спирали** как важнейший фактор структуризации в методике обучения информатике: вначале общее знакомство с понятием с учетом имеющегося опыта обучаемых, затем его последующее развитие и обогащение, создающее предпосылки для научного обобщения в старших классах.

**Принцип развивающего обучения** (обучение ориентировано не только на получение новых знаний в области информатики и информационных технологий, но и на активизацию мыслительных процессов, формирование и развитие у школьников обобщенных способов деятельности, формирование навыков самостоятельной работы).

# IV. ЦЕЛИ И ЗАДАЧИ ПРОГРАММЫ

**Цели программы:**

* пропедевтика наиболее значимых тем предмета «Информатика и ИКТ» посредством обучения программированию через создание творческих проектов в среде программирования **Scratch**;
* развитие логического мышления, творческого и познавательного потенциала подростка.

**Задачи программы:**

**Обучающие:**

* овладеть базовыми понятиями объектно-ориентированного программирования («объект», «событие», «управление», «обработка событий») и применение их при создании проектов в визуальной среде программирования Scratch;
* сформировать навыки разработки, тестирования и отладки несложных программ;
* приобщить обучающихся к новым технологиям, способным помочь им в реализации собственного творческого потенциала;
* познакомить с понятием проекта и алгоритмом его разработки;
* сформировать навыки разработки проектов: интерактивных историй, интерактивных игр, мультфильмов, интерактивных презентаций;
* развитие познавательной деятельности обучающихся в области новых информационных технологий;
* совершенствовать навыки работы на компьютере и повышение интереса к программированию.

**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;
* создать условия для повышения самооценки обучающегося, реализации его как личности;
* формировать потребность в саморазвитии;
* 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 культуру и навыки сетевого взаимодействия;
* способствовать развитию коммуникативных умений и навыков обучающихся;
* способствовать развитию творческих способностей и эстетического вкуса подростков;
* формировать умение демонстрировать результаты своей работы.

**V. Личностные, метапредметные и предметные результаты освоения курса**

Основные **личностные** результаты, формируемые в процессе освоения программы:

* 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.

К основным **метапредметным** результатам (осваиваемым обучающимися межпредметным понятиям и универсальным учебным действиям, способности их использования, как в учебной, так и в познавательной и социальной практике), формируемым в процессе освоения программы, можно отнести:

* 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.

Основные **предметные** результаты, формируемые в процессе изучения программы направлены на:

* осознание значения математики и информатики в повседневной жизни человека;
* осознание роли техники и технологий для прогрессивного развития общества;
* 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.

# VI. Условия реализации программы

Программа рассчитана на обучающихся **5-7 классов** – 11-13 лет и предполагает, что обучающиеся владеют навыками работы с клавиатурой, мышью, приемами работы с графическими изображениями, умеют сохранять работы, знают логическую структуру диска, программа не требует первоначальных знаний в области программирования.

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

Курс является практикоориентированным, поэтому каждое занятие содержит как теоретический, так и практический материал. Содержание практических занятий ориентировано не только на овладение обучающимися 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.

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

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

Полученные по окончании программы знания и умения могут способствовать развитию интереса к профессиям, связанным с программированием, анимацией, мультипликацией.

На занятиях обращается внимание на соблюдение требований безопасности труда, пожарной безопасности и личной гигиены.

# VII. Планируемые результаты изучения учебного курса

В результате освоения программы обучающиеся **получат 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, paint.net) 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.

Обучающиеся **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;
* создавать собственные изображения в других программах (например, Paint.net) и импортировать их в программную среду Scratch;
* использовать графические примитивы векторного редактора для создания объектов;
* создавать изображения из пунктирных и штрих-пунктирных линий с изменением цвета и толщины линии;
* упрощать программы за счёт использования циклических команд и применять их;
* составлять простые параллельные алгоритмы;
* создавать программы и игры с использованием интерактивных технологий;
* моделировать ситуации с использованием необходимых форм ветвления алгоритма, включая цикл по условию;
* передавать сообщения исполнителям для выполнения последовательности команд (включая разные типы исполнителей).
* планировать и создавать анимации по определенному сюжету;
* создавать игры, используя интерактивные возможности программной среды Scratch;
* планировать и создавать обучающие программы для иллюстрации пройденного материала других предметных областей;
* продумывать и описывать интерактивное взаимодействие для создания простейших тренажеров;
* подходить творчески к построению моделей различных объектов и систем.

# VIII. СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА И ТРЕБОВАНИЯ К УЧЕБНЫМ ДОСТИЖЕНИЯМ

| *Содержание учебного материала* | *Учебные достижения обучающихся* |
| --- | --- |
| **Раздел 1. Знакомство с программной средой Scratch (4 часа)** | |
| **Введение (1 час)** | *Обучающиеся должны знать:*   * 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 дальней 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.   *Обучающиеся должны уметь:*   * 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. |
| Знакомство со средой Scratch. Основные элементы пользовательского интерфейса программной среды Scratch. Функциональные блоки. Рабочее поле. Поле скриптов. Ящики команд: команды контроля, команды движения. Анимация движения объекта. Создание и сохранение документа. Техника безопасности |
| **Управление спрайтами (3 часа)** |
| Понятие спрайта и объекта. Исполнитель Scratch. Управление спрайтами: команды Идти, Повернуться на угол, Опустить перо, Поднять перо, Очистить  Координатная плоскость. Точка отсчёта, оси координат, единица измерения расстояния, абсцисса и ордината. Навигация в среде Scratch. Определение координат спрайта. Команда Идти в точку с заданными координатами  Создание проекта «Кругосветное путешествие Магеллана». Команда Плыть в точку с заданными координатами. Режим презентации |
| **Раздел 2. Компьютерная графика (2 часа)** | |
| Основные инструменты встроенного растрового графического редактора. Редактирование костюма спрайта. Редактирование фона. Создание нового фона.  Создание и редактирование спрайтов и фонов для сцены | *Обучающиеся должны знать:*   * выделять фрагменты изображения для дальней 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.   *Обучающиеся должны уметь:*   * 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. Конструкции управления в Scratch. Алгоритмы и исполнители (16 часов)** | |
| Линейный алгоритм. Рисование линий исполнителем Scratch. Исполнитель Scratch рисует квадраты и прямоугольники линейно  Понятие цикла. Многократное повторение команд как организация цикла. Команда Повторить Циклические алгоритмы. Упрощение программы путём сокращения количества команд при переходе от линейных алгоритмов к циклическим. Исполнитель Scratch рисует квадраты, линии  Конечный цикл. Исполнитель Scratch рисует несколько линий и фигур. Рисование узоров и орнаментов.  Циклический алгоритм. Цикл в цикле. Вложенные и внешние циклы. Копирование фрагментов программы. Повторение пунктирной линии с поворотом  Бесконечный цикл. Конструкция Всегда. Создание проектов «Автомобиль на дороге» и «Гонки по трассе». Команда Если край, Оттолкнуться  Ориентация по компасу. Управление курсом движения. Команда Повернуть в направление. Проект «Полёт мяча»  Бесконечный цикл. Одна программа для исполнителя Scratch, но разные костюмы. Анимация исполнителя Scratch на основе готовых костюмов.  Создание сюжета «Осьминожка», «Прыжки на скакалке» и «Порхание мотылька»  Несколько исполнителей. Параллельное выполнение действий исполнителями. Одинаковые программы для нескольких исполнителей. Создание мультипликационного сюжета «Аквариум». Создание мультипликационного сюжета «Кот и птичка»  Два исполнителя со своими программами. Мини-проект «Часы»  Алгоритмы с ветвлением. Условие ЕСЛИ. Соблюдение условий. Сенсоры. Блок если. Управляемый стрелками спрайт  Создание коллекции игр: «Лабиринт», «Кружащийся котёнок»  Составные условия. Проекты «Хождение по коридору», «Слепой кот», «Тренажёр памяти».  Датчик случайных чисел. Проекты «Разноцветный экран», «Хаотичное движение», «Кошки-мышки», «Вырастим цветник»  Циклы с условием. Проект «Будильник»  Запуск спрайтов с помощью мыши и клавиатуры. Проекты «Переодевалки» и «Дюймовочка»  Самоуправление спрайтов. Обмен сигналами. Блоки Передать сообщение и Когда я получу сообщение. Проекты «Лампочка» и «Диалог» | *Обучающиеся должны знать:*   * 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.   *Обучающиеся должны уметь:*   * 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 (2 часа)** | |
| Создание проектов по собственному замыслу | *Обучающиеся должны знать:*  • 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.  *Обучающиеся должны уметь:*  • использовать возможности программной среды Scratch для создания мультимедийных проектов;  • создавать имитационные модели, интерактивные проекты и игры средствами программной среды |

# IX. Тематическое планирование

| **№** | **Дата** | | | **Название раздела, темы урока** | | **Формы организа-ции занятия** | **Формы аттеста-ции/кон-троля** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **план** | **факт** | |
| **Раздел 1. Знакомство с программной средой Scratch (4 часа)** | | | | | | | |
|  |  | | | **Введение (1 час)** | |  |  |
|  |  |  | | Знакомство со средой **Scratch**. Основные элементы пользовательского интерфейса программной среды **Scratch**. Функциональные блоки. Рабочее поле. Поле скриптов. Ящики команд: команды контроля, команды движения. Анимация движения объекта. Создание и сохранение документа. Техника безопасности | | Индивиду-альные |  |
|  |  |  | | **Управление спрайтами (3 часа)** | |  |  |
|  |  |  | | Понятие спрайта и объекта. Исполнитель **Scratch**. Управление спрайтами: **команды** ***Идти*, *Повернуться на угол*, *Опустить перо*, *Поднять перо*, *Очистить*** | | Индивиду-альные |  |
|  |  |  | | Координатная плоскость. Точка отсчёта, оси координат, единица измерения расстояния, абсцисса и ордината. Навигация в среде **Scratch**. Определение координат спрайта. **Команда** ***Идти в точку с заданными координатами*** | | Индивиду-альные |  |
|  |  |  | | Создание проекта «Кругосветное путешествие Магеллана». **Команда** ***Плыть в точку с заданными координатами***. Режим презентации | | Подгруп-повые | Творчес-кая работа |
| **Раздел 2. Компьютерная графика (2 часа)** | | | | | | | |
|  |  | |  | | Основные инструменты встроенного растрового графического редактора. Редактирование костюма спрайта. Редактирование фона. Создание нового фона | Подгруп-повые | Творчес-кая работа |
|  |  | |  | | Создание и редактирование спрайтов и фонов для сцены | Индивиду-альные | Творчес-кая работа |
| **Раздел 3. Конструкции управления в Scratch. Алгоритмы и исполнители (16 часов)** | | | | | | | |
|  |  |  | | Линейный алгоритм. Рисование линий исполнителем **Scratch**. Исполнитель **Scratch** рисует квадраты и прямоугольники линейно | |  |  |
|  |  |  | | Понятие цикла. Многократное повторение команд как организация цикла. **Команда** ***Повторить*** Циклические алгоритмы. Упрощение программы путём сокращения количества команд при переходе от линейных алгоритмов к циклическим. Исполнитель **Scratch** рисует квадраты, линии | | Индивиду-альные |  |
|  |  |  | | Конечный цикл. Исполнитель **Scratch** рисует несколько линий и фигур. Рисование узоров и орнаментов | | Группо-вые |  |
|  |  |  | | Циклический алгоритм. Цикл в цикле. Вложенные и внешние циклы. Копирование фрагментов программы. Повторение пунктирной линии с поворотом | |  |  |
|  |  |  | | Бесконечный цикл. **Конструкция** ***Всегда***. Создание проектов «Автомобиль на дороге» и «Гонки по трассе». **Команда** ***Если край*, *Оттолкнуться*** | | Подгруп-повые | Творчес-кая работа |
|  |  |  | | Ориентация по компасу. Управление курсом движения. **Команда** ***Повернуть в направление***. Проект «Полёт мяча» | | Подгруп-повые | Творчес-кая работа |
|  |  |  | | Бесконечный цикл. Одна программа для исполнителя **Scratch**, но разные костюмы. Анимация исполнителя **Scratch** на основе готовых костюмов.  Создание сюжета «Осьминожка», «Прыжки на скакалке» и «Порхание мотылька» | | Группо-вые | Творчес-кая работа |
|  |  |  | | Несколько исполнителей. Параллельное выполнение действий исполнителями. Одинаковые программы для нескольких исполнителей. Создание мультипликационного сюжета «Аквариум». Создание мультипликационного сюжета «Кот и птичка» | | Группо-вые | Творчес-кая работа |
|  |  |  | | Два исполнителя со своими программами. Мини-проект «Часы» | | Индиви-дуальные | Творчес-кая работа |
|  |  |  | | Алгоритмы с ветвлением. Условие ЕСЛИ. Соблюдение условий. Сенсоры. Блок **если.**Управляемый стрелками спрайт | | Индивидуальные |  |
|  |  |  | | Создание коллекции игр: «Лабиринт», «Кружащийся котёнок» | | Индиви-дуальные | Творчес-кая работа |
|  |  |  | | Составные условия. Проекты «Хождение по коридору», «Слепой кот», «Тренажёр памяти». | | Группо-вые | Творчес-кая работа |
|  |  |  | | Датчик случайных чисел. Проекты «Разноцветный экран», «Хаотичное движение», «Кошки-мышки», «Вырастим цветник» | | Группо-вые | Творчес-кая работа |
|  |  |  | | Циклы с условием. Проект «Будильник» | | Подгруп-повые | Творчес-кая работа |
|  |  |  | | Запуск спрайтов с помощью мыши и клавиатуры. Проекты «Переодевалки» и «Дюймовочка» | | Подгруп-повые | Творчес-кая работа |
|  |  |  | | Самоуправление спрайтов. Обмен сигналами. **Блоки** ***Передать сообщение***и***Когда я получу сообщение***. Проекты «Лампочка» и «Диалог» | | Подгруп-повые | Творчес-кая работа |
| **Раздел 4. Проектная деятельность (2 часа)** | | | | | | | |
|  |  | |  | | Создание проектов по собственному замыслу | Индивиду-альные | Творчес-кая работа |
|  |  | |  | | Создание проектов по собственному замыслу | Подгруп-повые  Индивиду-альные | Защита проектов |

# X. Обязательные формы контроля и критерии оценивания

В процессе обучения обучающиеся не получают прямых оценок своей деятельности.

В качестве способов проверки результатов в процессе обучения применяются тестирование по изучаемым темам, конкурсы между обучающимися на скорость и качество решения поставленной задачи. При оценке учитывается правильность, оптимальность, скорость решения задачи и уровень самостоятельности при решении задачи. Контроль усвоения осуществляется педагогом на каждом занятии для коррекции педагогической деятельности. Успехи, достигнутые учениками, демонстрируются во время презентации проектов и оцениваются соучениками и педагогом (руководителем кружка). После проведения презентации или испытания предполагается рефлексия, где каждый ребёнок высказывает своё мнение о том, что у него лучше всего получилось и над чем стоит поработать в дальнейшем. Во время проведения презентации проектов необходим анализ недочётов, при этом подчёркиваются позитивные стороны каждой ситуации.

**Механизм оценивания результативности освоения программы**

**Параметры оценивания знаний, умений и навыков обучающихся**

| **Измеряемые параметры** | **Критерии оценки** | | | | |
| --- | --- | --- | --- | --- | --- |
| **Минимальный уровень знаний и умений** | **Допустимый уровень знаний и умений** | **Приемлемый уровень знаний и умений** | **Оптимальный уровень знаний и умений** | |
| **1. Знания в области техники безопасности** | | | | | |
| Знание правил внутреннего распорядка и техники безопасности при работе с вычислительной техникой | Ученик слабо формулирует правила ТБ при работе с компьютерной техникой | Ученик неуверенно формулирует правила ТБ при работе с компьютерной техникой | Ученик уверенно формулирует правила ТБ при работе с компьютерной техникой, но не всегда знает, как их применить | Ученик отлично знает правила ТБ при работе с компьютерной техникой и самостоятельно их применяет | |
| **2. Теоретические знания в области компьютерной техники** | | | | | |
| Знания элементов среды **Scratch** | Ученик имеет недостаточное общее представление об основных элементах среды **Scratch** | Ученик недостаточно уверенно разбирается в основных элементах среды **Scratch** с помощью педагога | Ученик разбирается в основных элементах среды **Scratch** с помощью педагога | Ученик самостоятельно разбирается в основных элементах среды **Scratch** | |
| Знание основных операторов и правил создания программ в среде **Scratch** | Ученик слабо ориентируется в основных операторах и правилах в создания программ в среде **Scratch** | Ученик недостаточно уверенно ориентируется в основных операторах и правилах создания программ в среде **Scratch** | Ученик ориентируется в основных операторах и правилах создания программ в среде **Scratch** с помощью педагога | Ученик самостоятельно ориентируется в основных операторах и правилах создания программ в среде **Scratch** | |
| Знание основ проектной деятельности | Ученик слабо ориентируется в основах проектной деятельности | Ученик недостаточно уверенно ориентируется  в основах проектной деятельности с помощью педагога. | Ученик ориентируется в основах проектной деятельности с помощью педагога. | Ученик хорошо и самостоятельно ориентируется в основах проектной деятельности | |
| **3. Практические навыки в области компьютерной техники** | | | | | |
| Практические навыки работы в среде **Scratch** | Ученик с трудом выполняет задания по работе в среде **Scratch** | Ученик неуверенно выполняет задания по работе в среде **Scratch** с помощью педагога | Ученик неуверенно самостоятельно выполняет задания по работе в среде **Scratch** | | Ученик уверенно самостоятельно выполняет задания по работе в среде **Scratch** |
| Формирование и сопровождение проектов | Ученик с трудом и неуверенно создает проект, либо не создает его совсем | Ученик неуверенно, с помощью педагога создает проект | Ученик достаточно самостоятельно, но с помощью педагога создает проект | | Ученик уверенно и самостоятельно создает проекты |
| **4. Личностные качества ребенка** | | | | | |
| Коммуникабельность | Ученик замкнут, плохо идет на контакт с товарищами и педагогом, боится обратиться за помощью | Ученик обращается за помощью только в том случае, если совсем заходит в тупик | Ученик легко общается с людьми, но не всегда обращается за помощью при затруднениях в работе | | Ученик всегда обращается за помощью при затруднениях и сам готов помочь товарищам, легко общается с людьми |
| Трудолюбие | Работу выполняет небрежно, не хочет исправлять ошибки | Работу выполняет не всегда аккуратно, неохотно исправляет ошибки | Работу выполняет охотно, но ошибки исправляет после вмешательства педагога | | Работу выполняет охотно и тщательно, стремится самостоятельно исправлять ошибки |
| Креативность | Не склонен проявлять фантазию и творческий подход при работе | Неохотно проявляет фантазию и творческий подход при работе | Неохотно проявляет фантазию, но использует творческий подход при работе | | Всегда проявляет фантазию и творческий подход при работе |

# XI. Методическое, учебно-методическое и материально-техническое обеспечение образовательного процесса

Для реализации программы используются следующие **методы обучения**:

По **источнику полученных знаний**: словесные, наглядные, практические.

По **способу организации** познавательной деятельности:

* развивающего обучения (проблемный, проектный, творческий, частично-поисковый, исследовательский, программированный);
* дифференцированного обучения (уровневые, индивидуальные задания);
* игровые (конкурсы, игры-конструкторы, турниры с использованием мультимедиа, дидактические).

**Средства:**

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

**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 800х600.

**Программное обеспечение:**

* операционная система: Windows XP/7/8/10, Linux;
* графический редактор: Paint.net;
* браузер Internet Explorer, Opera, Chrome, Mozilla;
* компьютерные программы: среда визуального программирования Scrath.

# СПИСОК РЕКОМЕНДОВАННОЙ ЛИТЕРАТУРЫ

**Литература для педагога:**

1. Патаракин Е.Д. Учимся готовить в среде Скретч: учебно-методическое пособие) — И.: Интуит.ру, 2007. – 61 с.: ил.
2. Босова Л.Л., Методика применения интерактивных сред для обучения младших школьников программированию / Л.Л. Босова, Т.Е. Сорокина //Информатика и образование. – №7 (256). – 2014.
3. В.Г. Рындак, В.О. Джинжер, Л.В. Денисова, «Пропедевтика идей параллельного программирования в средней школе при помощи среды Scratch».
4. В.Г. Рындак, В.О. Джинжер, Л.В. Денисова, «Ранее обучение программированию в среде Scratch»
5. В.Г. Рындак, В.О. Дженжер, Л.В. Денисова, «Учебно-методическое пособие. Проектная деятельность школьника в среде программирования Scratch» / - Оренбург - 2009
6. Сорокина Т.Е. Пропедевтика программирования в курсе информатики 5–6 классов на базе среды SCRATCH: мат. конф./ XII открытая Всерос. конф. (15–16 мая 2014 г.). – Казань: Казанский (Приволжский) федеральный университет, 2014. – 369 c. – С. 313–315.

**Литература для обучающегося:**

1. Творческие задания в среде Scratch: рабочая тетрадь для 5-6 классов/ Ю.В. Пашковская. – М.: БИНОМ. Лаборатория знаний, 2014. – 200 с.: ил.
2. Цветкова М.С., Масленикова О.Н. «Практические задания с использованием информационных технологий для 5-6 классов: Практикум» - М.: БИНОМ. Лаборатория знаний, 2007.
3. Пашковская Ю.В. «Творческие задания в среде программирования Scratch. 5-6 классы. Рабочая тетрадь» - М.: БИНОМ. Лаборатория знаний, 2013.

**Интернет ресурсы:**

1. http://scratch.mit.edu – официальный сайт Scratch.
2. http://letopisi.ru/index.php /Скретч - Скретч в Летописи.ру.
3. <http://setilab.ru/scratch/category/commun> - Учитесь со Scratch.
4. https://youtu.be/vd20J2r5wUQ – видеоурок по SCRATCH [электронный ресурс].
5. https://youtu.be/jSs9axeyBHs – видеоурок по SCRATCH [электронный ресурс].
6. https://youtu.be/JjMDHJtFvFM – видеоурок по SCRATCH [электронный ресурс].
7. https://youtu.be/x-jPH3gzifw – видеоурок по SCRATCH [электронный ресурс]. [http://scratch.sostradanie.org](http://scratch.sostradanie.org/) – Изучаем Scratch.
8. <http://odjiri.narod.ru/tutorial.html> – учебник по Scratch.
9. [http://younglinux.info](http://younglinux.info/) - Цикл из 10 уроков “Введение в Scratch”.
10. <http://anngeorg.ru/info/scratch> – Знакомимся с программой Scratch.