

# Пояснительная записка

Сегодня навыки программирования нужны не только разработчикам сайтов, но и дизайнерам, артистам, учёным, аналитикам «больших данных». Спрос на специалистов, обладающих навыками программирования, в разы превосходит количество квалифицированных кадров.

Использование новых информационных технологий требует хорошей компьютерной подготовки. Для успешного обучения важен не столько набор знаний, сколько развитое мышление, умение получать знания, использовать имеющиеся навыки для решения различных учебных задач. Большие возможности при этом раскрываются при работе с компьютером.

Scratch – это творческая среда, разработанная специально для развития мышления, творческих и исследовательских способностей детей и подростков. Среда Scratch появилась в 2007 году под руководством профессора Митчелла Резника в исследовательской группе под названием Lifelong Kindergarten research group, которая существует при Массачусетском технологическом институте.

Работая над проектом в Scratch, дети учатся думать креативно (созидательно) и решать проблемы систематично – а это умения, которые являются критическими для достижения успеха в XXI веке.

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

Курс построен таким образом, чтобы мотивировать учащихся к занятиям программированием и найти ответы на вопросы, с которыми им приходится сталкиваться в повседневной жизни при работе с большим объемом информации; научиться общаться с компьютером, который ничего не сможет сделать, если человек не напишет для него соответствующую программу.

Курс нацелен на формирование навыков, которые пригодятся ребенку в будущем: умение анализировать информацию, придумывать новые решения задач, работать в команде, создавать и презентовать свой проект.

Актуальность данной программы заключается в том, что в ней существенно углубляется содержание образовательных школьных программ по информатике и даёт возможность детям освоить прикладные умения по информатике, необходимые в современной жизни.

Программа составлена в соответствии с Федеральным законом от 29 декабря 2012 года №273-ФЗ «Об образовании в Российской Федерации» (с изменениями и дополнениями); Концепцией развития дополнительного образования детей (утверждена распоряжением Правительства Российской Федерации от 4 сентября 2014 г. № 1726-р); Порядком организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам (утвержден Приказом Министерства образования и науки Российской Федерации от 29 августа 2013 г. № 1008; Письмом Министерства образования и науки Российской Федерации от 11.12.2006 г. № 06-1844 «О примерных требованиях к программам дополнительного образования детей»; Письмом Министерства образования и науки Российской Федерации от 18.11.2015 г. № 09-3242 «Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)».

**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.

**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 подростка, т.к. дает возможность увидеть практическое назначение алгоритмов и программ, что будет способствовать развитию интереса к профессиям, связанным с программированием.

**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 программирования.

**Основной вид деятельности: игра.** Также на занятиях практикуется учебная, познавательная и творческая деятельность индивидуально и в группах

**Цели программы:**

Развитие логического мышления, творческого и познавательного потенциала подростка, формирование у учащихся алгоритмического и структурного мышления, используя программирование в среде Scratch.

**Задачи:**

**Образовательные:**

* Овладение базовыми понятиями объектно-ориентированного программирования и применение их при создании проектов в визуальной среде программирования Scratch;
* Приобщение обучающихся к новым технологиям, способным помочь им в реализации собственного творческого потенциала;
* Развитие познавательной деятельности учащихся в области новых информационных технологий;
* Совершенствование навыков работы на компьютере и повышение интереса к программированию.

**Воспитательные:**

* Формирование культуру и навыки сетевого взаимодействия;
* Формировать эмоционально-положительное отношение к компьютерам и программированию.

**Развивающие:**

* Способствование развитию творческих способностей и эстетического вкуса подростков;
* Способствование развитию коммуникативных умений и навыков обучающихся.
* Способствование развитию логического мышления, памяти и умению анализировать;
* Создание условия для повышения самооценки обучающегося, реализации его как личности;
* Формирование потребности в саморазвитии;
* Способствование развитию познавательной самостоятельности.

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

* 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), формируемые 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;
* 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.

**Условия реализации программы:**

Срок реализации программы – 1 год (68 уч.ч.)

Программа рассчитана на учащихся – 8-12 лет и предполагает, что учащиеся 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.

Предполагаемый объем учебного времени – 2 часа в неделю.

Занятия проводятся 1 раз в неделю по 2 уч.часа. Программа рассчитана на 68 часов в год.

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

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.

**Формы контроля**

**Входящий контроль** проводится в начале учебного года с целью выявления имеющихся знаний и умений детей по программе на начало учебного года

**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.

**Оценка результатов обучения по дополнительной общеобразовательной программе «Увлекательное программирование на Scratch»**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **показатели** | **критерии** | **Степень выраженности оцениваемого качества** | **баллы** | **Формы**  **контроля** |
| Теоретическая подготовка | Соответствие теоретических знаний программным требованиям | Низкий уровень (ребенок овладел менее чем ½ объема знаний, предусмотренных программой)  Средний уровень (объем усвоенных знаний составляет более 1\2)  Высокий уровень (ребенок освоил практически весь объем знаний, предусмотренный программой) | 1-3  4-7  8-10 | Тест, собеседование |
| Практическая подготовка | Соответствие практических умений и навыков программе | Низкий уровень (практические умения и навыки неустойчивые, требуется постоянная помощь по их использованию)  Средний уровень (овладел практическими умениями и навыками, предусмотренными программой, применяет их под руководством педагога)  Высокий уровень (обучающийся овладел в полном объеме практическими умениями и навыками, практические работы выполняет самостоятельно, качественно, присутствует творческий подход к выполнению работы) | 1-3  4-7  8-10 | Практические работы, тест, защита проектов |

В **результате** освоения программы учащиеся **получат представление** о:

* 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 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.

Учащиеся будут **уметь**:

* 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 и импортировать их в программную среду Scratch;
* использовать графические примитивы векторного редактора для создания объектов;
* 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.

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.

**Учебно–тематический план образовательной программы:**

Учебная программа представлена следующими разделам:

1. Знакомство с программной средой Scratch
2. Компьютерная графика
3. Алгоритмы и исполнители
4. Проектная деятельность и моделирование процессов и систем

**Содержание программы:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ раздела** | **Название раздела** | **Теория** | **Практика** | **Всего** |
| 1 | Знакомство с программной средой Scratch | 2 | 4 | 6 |
| 2 | Компьютерная графика | 4 | 6 | 10 |
| 3 | Алгоритмы и исполнители | 20 | 22 | 42 |
| 4 | Проектная деятельность и моделирование процессов и систем | 3 | 7 | 10 |
|  | Итого | 29 | 39 | 68 |

**Раздел 1. Знакомство с программной средой Scratch**

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 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.

*Практическая деятельность:*

* 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 изображение 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 и импортирование их в программную среду Scratch.

Возможность создания геометрических фигур без внутренней заливки, но с текстовым блоком внутри. Стрелки, их направление.

*Аналитическая деятельность:*

* 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.

*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.

**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 исполнителям 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 одинотипных действий. Принцип суперкомпъютерных 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.

**Методическое обеспечение:**

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)

**Условия реализации рабочей программы**

**Материально-технические условия реализации рабочей программы:**

* Персональный компьютер или ноутбук с предустановленным программным обеспечением (Операционная система Windows/Linux, MS Office или аналог)
* Экран
* Колонки
* Средства телекоммуникации - локальная сеть, выход в Интернет
* Учебно-практическое и учебно-лабораторное оборудование
* Презентации и видео к урокам, дидактический материал. Видеохостинг Youtubе (видеоуроки «работа в среде Scratch»)
* Специализированная учебная мебель: компьютерные столы, стулья

**Список используемой литературы:**

1. **Денисова Л. В., Дженжер В. О.** Пропедевтика идей параллельного программирования в средней школе при помощи среды Scratch / Современные информационные технологии и ИТ-образование: III Межд. науч.-практ. конф., Москва, МГУ имени М. В. Ломоносова, 2008 г.: Сб. докладов: Учебно-методическое пособие / Под ред. В. А. Сухомлина. — М.: МАКС Пресс, 2008. — С 451–459.
2. **Патаракин Е.** **Д.** Учимся готовить в среде Scratch. **-** Санкт Петербург, 2008
3. **Патаракин Е.Д.** Освоение медиа-культуры через учебные игры с маленькими кирпичиками знаний. В книге Судьба России: вектор перемен, 2007
4. **Рындак В. Г., Дженжер В. О., Денисова Л. В.** Проектная деятельность школьника в среде программирования Scratch: учебно-методическое пособие / В. Г. Рындак, В. О. Дженжер, Л. В. Денисова. — Оренбург: Оренб. гос. ин-т. менеджмента, 2009. — 116 с.: ил.
5. **Рындак В. Г., Дженжер В. О., Денисова Л. В.** Концепция организации внеучебной проектной научно-познавательной деятельности школьника. // Образование и наука. Известия УрО РАО. — Екатеринбург: Изд-во УрО РАО, 2009. — № 7 (64). — С. 12–22.