**МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**«ШКОЛА №126 С УГЛУБЛЕННЫМ ИЗУЧЕНИЕМ АНГЛИЙСКОГОЯЗЫКА»**

|  |  |  |
| --- | --- | --- |
| РассмотреноНа ШМОПротокол №\_\_\_\_\_от «\_\_\_\_»\_августа\_\_\_\_Председатель ШМО\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Принятона педагогическом советеПротокол №\_\_\_\_\_\_\_\_\_\_\_от «\_\_\_\_»\_августа\_\_\_\_\_\_ | Утвержденоприказом директораПриказ №\_\_\_\_\_\_\_\_\_\_\_\_\_от «\_\_\_\_»\_августа\_\_\_\_\_Директор школы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

РАБОЧАЯ ПРОГРАММА

внеурочной деятельности

«Увлекательное программирование на Scratch»

ИНФОРМАТИКА

7 КЛАСС

Срок реализации программы – 1 год

Составила: Пронина Н.А.

учитель информатики

Нижний Новгород

2017-2018 учебный год

***ПОЯСНИТЕЛЬНАЯ ЗАПИСКА***

Программа составлена в соответствии с Федеральным законом «Об образовании в Российской Федерации» от 29.12.2012 № 273.

Рабочая программа «Увлекательное программирование на Scratch» разрабатывалась на основе следующих материалов и документов:

* модуль «Пропедевтика программирования со Scratch», Сорокина Т.Е.;
* «Творческие задания в среде Scratch: рабочая тетрадь для 5-6 классов», Ю.В. Пашковская;
* «Пропедевтика идей параллельного программирования в средней школе при помощи среды Scratch», В.Г. Рындак, В.О. Джинжер, Л.В. Денисова;
* «Ранее обучение программированию в среде Scratch», В.Г. Рындак, В.О. Джинжер, Л.В. Денисова;
* «Проектная деятельность школьника в среде программирования Scratch»/Учебно-методическое пособие/ В.Г. Рындак, В.О. Джинжер, Л.В. Денисова.

**Цель:** сформировать у учащихся базовые представлений о языках программирования, алгоритме, исполнителе, способах записи алгоритма; организовать проектную научно-познавательную деятельность творческого характера; сформировать у ученика 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;
* способствовать развитию коммуникативных умений и навыков обучающихся;
* формировать умение демонстрировать результаты своей работы.

***ОБЩАЯ ХАРАКТЕРИСТИКА КУРСА***

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 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.

По поводу целей проекта Митчелл Резник сказал: «Это следующий шаг в генерации контента (материалов) пользователем. Нашей целью было расширить диапазон того, что дети могут создавать, совместно использовать и изучать. 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.

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 имеются графические блоки, которые совмещаются 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. Есть возможность сделать изменения в стеках, даже когда программа запущена, что позволяет больше экспериментировать с новыми идеями снова и снова.

**Манипуляции данными.** Со Scratch можно 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: можно 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**.

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 Scratсh, 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 Downloаd.

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 рабочую программу «Увлекательное программирование на Scratch» 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.

**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 учащихся – свободный, по желанию детей и их родителей.

**Организация деятельности учащихся:**

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.

**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).

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.

**Условия реализации программы.**

Срок реализации программы – 1 год.

Программа рассчитана на учащихся V-VII классов 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 – 1 раз в неделю по 40 минут. Программа рассчитана на 35 часов в год.

***ЛИЧНОСТНЫЕ, МЕТАПРЕДМЕТНЫЕ И ПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ СОДЕРЖАНИЯ КУРСА***

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 навыков и умений безопасного и целесообразного поведения при работе с компьютерными программами и в Интернете;
* формирование умения соблюдать нормы информационной этики и права.

***СОДЕРЖАНИЕ КУРСА***

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

Свободное программное обеспечение. Авторы программной среды Scratch. Параметры для скачивания и установки программной среды на домашний компьютер.

Основные элементы пользовательского интерфейса программной среды Scratch. Внешний вид рабочего окна. Блочная структура систематизации информации. Функциональные блоки. Блоки команд, состояний, программ, запуска, действий и исполнителей. Установка русского языка для Scratch.

Создание и сохранение документа. Понятия спрайта, сцены, скрипта. Очистка экрана.

Основной персонаж как исполнитель программ. Система команд исполнителя (СКИ). Блочная структура программы. Непосредственное управление исполнителем.

Библиотека персонажей. Сцена и разнообразие сцен, исходя из библиотеки данных. Систематизация данных библиотек персонажей и сцен. Иерархия в организации хранения костюмов персонажа и фонов для сцен. Импорт костюма, импорт фона.

**Аналитическая деятельность:**

* выделять аппаратное и программное обеспечение компьютера;
* определять технические устройства для ввода и вывода информации;
* понимать иерархическую организацию библиотеки данных программной среды;
* выделять путь к элементам библиотеки;
* выделять фрагменты изображения для 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:**

* выбирать и запускать программную среду Scratch;
* работать с основными элементами пользовательского интерфейса программной среды;
* изменять размер и перемещать окно программы, выбирать необходимый режим окна;
* вводить имя файла с помощью клавиатуры;
* выбирать необходимый файл из нужной папки библиотеки программы;
* создавать, копировать, переименовывать, перемещать, копировать и удалять файлы;
* соблюдать требования техники безопасности при работе в компьютерном классе.

**Раздел 2. Компьютерная графика**

Компьютерная графика. Векторные и растровые графические редакторы. Встроенный растровый графический редактор. Основные инструменты графического редактора — кисточка, ластик, заливка (цветом или градиентом), рисование линий, прямоугольников, квадратов, эллипсов и окружностей, выбор фрагмента 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.

**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. Создание блок-схем в графическом редакторе.

**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.

***ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ***

|  |  |  |
| --- | --- | --- |
| **№****п/п** | **Тема (раздел)** | **Кол-во****часов** |
| **1** | **Знакомство с программной средой Scratch**  | **3** |
| **2** | **Компьютерная графика** | **12** |
| **3** | **Алгоритмы и исполнители** | **15** |
| **3.1** | Линейные алгоритмы | 3 |
| **3.2** | Циклические алгоритмы | 3 |
| **3.3** | Параллелизм в программной среде | 3 |
| **3.4** | Ветвление в алгоритмах | 3 |
| **3.5** | Последовательное выполнение фрагментов программы разными исполнителями | 3 |
| **4** | **Проектная деятельность и моделирование процессов и систем** | **5** |
|  | **Всего:** | **35** |

***ПОУРОЧНОЕ ПЛАНИРОВАНИЕ***

|  |  |  |
| --- | --- | --- |
| **Номер урока** | **Тема урока** | Сроки |
| 1 | Знакомство со средой Скретч. Понятие спрайта и объекта. Создание и редактирование спрайтов и фонов для сцены. |   |
| 2 | Знакомство со средой Скретч (продолжение). Пользуемся помощью Интернета. Поиск, импорт и редакция спрайтов и фонов из Интернета. |   |
| 3 | Управление спрайтами: команды **идти, повернуться на угол, опустить перо, поднять перо, очистить.** |   |
| 4 | Рисование в Скретч. Координатная плоскость. Точка отсчёта, оси координат, единица измерения расстояния, абсцисса и ордината. |   |
| 5 | Навигация в среде Скретч. Определение координат спрайта. Команда **идти в точку с заданными координатами**. |   |
| 6 | Создание проекта «Кругосветное путешествие Магеллана». Команда **плыть в точку с заданными координатами**.  |   |
| 7 | Создание проекта «Кругосветное путешествие Магеллана» (продолжение). Режим презентации. |   |
| 8 | Понятие цикла. Команда **повторить**. Рисование узоров и орнаментов. |   |
| 9 | Конструкция **всегда**. Создание проектов «Берегись автомобиля!» и «Гонки по вертикали». Команда **если край, оттолкнуться.**  |   |
| 10 | Ориентация по компасу. Управление курсом движения. Команда **повернуть в направление**. Проект «Полёт самолёта». |   |
| 11 | Спрайты меняют костюмы. Анимация. Создание проектов «Осьминог», «Девочка, прыгающая на скакалке» и «Бегущий человек».  |   |
| 12 | Создание мультипликационного сюжета «Кот и птичка». Управление несколькими объектами.  |   |
| 13 | Создание мультипликационного сюжета «Кот и птичка» (продолжение). |   |
| 14 | Соблюдение условий. Сенсоры. Блок **если.** Управляемый стрелками спрайт. |   |
| 15 | Создание коллекции игр: «Лабиринт», «Кружащийся котёнок». |   |
| 16 | Составные условия. Проекты «Хождение по коридору», «Слепой кот», «Тренажёр памяти». |   |
| 17 | Датчик случайных чисел. Проекты «Разноцветный экран», «Хаотичное движение», «Кошки-мышки», «Вырастим цветник». |   |
| 18 | Циклы с условием. Проект «Будильник».  |   |
| 19 | Последовательное и одновременное выполнение команд. |  |
| 20 | Запуск спрайтов с помощью мыши и клавиатуры. Проекты «Переодевалки» и «Дюймовочка». |   |
| 21 | Самоуправление спрайтов. Обмен сигналами. Блоки **передать сообщение** и **Когда я получу сообщение**. Проекты «Лампа» и «Диалог». |   |
| 22 | Доработка проектов «Магеллан», «Лабиринт».  |   |
| 23  | Датчики. Проекты «Котёнок-обжора» |   |
| 24 | Переменные. Их создание. Использование счётчиков. Анимация. Разворачиваем Пчелу в направление движения. |   |
| 25 | Ввод переменных. Проект «Цветы». Доработка проекта «Лабиринт» - запоминание имени лучшего игрока.  |   |
| 26 | Ввод переменных с помощью рычажка. Проекты «Цветы» (вариант-2), «Правильные многоугольники». |   |
| 27-28 | Список как упорядоченный набор однотипной информации. Создание списков. Добавление и удаление элементов. Проекты «Гадание», «Назойливый собеседник». |   |
| 29 | Поиграем со словами. Строковые константы и переменные. Операции со строками. |   |
| 30 | Организация интерактивного диалога с пользователем |   |
| 31 | Создание игры «Угадай слово». |   |
| 32 | Создание тестов – с выбором ответа и без. |   |
| 33-34 | Свободное проектирование. Создание проектов по собственному замыслу. Регистрация в Скретч-сообществе.  |   |
| 35 | Презентация творческих проектов. |   |

***ОПИСАНИЕ УЧЕБНО-МЕТОДИЧЕСКОГО И МАТЕРИАЛЬНО-ТЕХНИЧЕСКОГО ОБЕСПЕЧЕНИЯ***

**УЧЕБНО-МЕТОДИЧЕСКИЕ СРЕДСТВА:**

Дидактические материалы (опорные конспекты, проекты примеры, раздаточный материал для практических работ)

Методические разработки (презентации, видеоуроки, flash-ролики)

Сетевые ресурсы Scratch

Видеохостинг Youtub (видеоуроки «работа в среде Scratch»)

Учебно-тематический план

**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 7 или Windows 8

Open Office, Microsoft Office

Компьютерные программы: Scrath

**СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ:**

**Литература для педагога:**

1. Модуль «Пропедевтика программирования со Scratch», Сорокина Т.Е;
2. Рындак В. Г., Дженжер В. О., Денисова Л. В. Проектная деятельность школьника в среде программирования Scratch. — Оренбург: Оренб. гос. ин-т. менеджмента, 2009.
3. «Пропедевтика идей параллельного программирования в средней школе при помощи среды Scratch», В.Г. Рындак, В.О. Джинжер, Л.В. Денисова;
4. «Ранее обучение программирование в среде Scratch», В.Г. Рындак, В.О. Джинжер, Л.В. Денисова;
5. Сборник «Информатика. Математика. Программы внеурочной деятельности для начальной и основной школы: 3 – 6 классы» / М.С. Цветкова, О.Б.Богомолова. – М.: БИНОМ. Лаборатория знаний, 2013. – 128 с.: ил.

**Литература для учащегося:**

1. Творческие задания в среде Scratch: рабочая тетрадь для 5-6 классов/ Ю.В. Пашковская. – М.: БИНОМ. Лаборатория знаний, 2014. – 200 с.: ил.

**Интернет ресурсы:**

<http://scratch.mit.edu> – официальный сайт Scratch

<http://letopisi.ru/index.php> /Скретч - Скретч в Летописи.ру

<http://setilab.ru/scratch/category/commun> - Учитесь со Scratch

***ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ИЗУЧЕНИЯ***

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.

***Получит 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.

***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;
* 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 и импортировать их в программную среду Scratch;
* использовать графические примитивы 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.

***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.

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.