

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Государственное бюджетное общеобразовательное учреждение
«Начальная школа №1 с.п. Экажево»

РАССМОТРЕНО
На заседании
педагогического совета
Протокол № 1 от
26.08 2024 г.

СОГЛАСОВАНО
Методист
регионального
модельного центра РИ
Булгаева Л.Х. Булгаева

УТВЕРЖДЕНО
Директора
ГБОУ «НОШ №1
с.п. Экажево»
Талиева Г.М. Талиева
«27.08» 2024 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
Технической направленности
«Юный программист»

Вид программы: Модифицированная
одноуровневая
стартового уровня

тип программы: сложно- структурированная

срок реализации: 1 год (108 ч.)

возраст обучающихся: 9-11 лет

форма обучения: очная

Автор педагог дополнительного образования Гелисханов М.М.

с.п. Экажево 2024 г.

1. Пояснительная записка.
2. Планируемые результаты освоения курса внеурочной деятельности.
3. Содержание учебного предмета, курса.
4. Тематическое планирование.

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА.

Дополнительная общеобразовательная программа «Юный программист» (далее программа) имеет техническую направленность, она предназначена для получения учащимися дополнительных знаний в области программирования в игровой, увлекательной форме, используя языки программирования: Scratch.

Нормативные правовые акты и государственные программные документы в соответствие с которыми разработана дополнительная общеобразовательная программа:

1. Федеральный закон Российской Федерации от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации»;
2. Федеральный закон от 24 июля 1998 г. № 124-ФЗ «Об основных гарантиях прав ребенка в Российской Федерации»
3. Приказ Министерства просвещения РФ от 3 сентября 2019 г. № 467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей» (в редакции от 21 апреля 2023 г.)
4. Концепция развития дополнительного образования детей до 2030 г., утверждённая распоряжением Правительства РФ от 31 марта №678-р 9в редакции от 21 апреля 2023 г.)
5. Приказ Министерства просвещения РФ от 27 июля 2022 г. №629 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
6. Федеральный закон от 31 июля 2020 г. № 304-ФЗ «О внесении изменений в ФЗ «Об образовании в РФ» по вопросам воспитания обучающихся»
7. Приказ Минтруда и соцзащиты РФ от 22 сентября 2021 г. № 652н «Профессиональный стандарт «Педагог дополнительного образования детей и взрослых»»
8. Постановление Главного государственного санитарного врача Российской Федерации от 28 сентября июля 2020 года № 28 г. Москва «Об утверждении СанПиН 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания, обучения,отдыха и оздоровления детей и молодежи «

Актуальность программы.

Большинство школьных программ по информатике в основном охватывают пользовательский уровень обучения работы на компьютере. Школьников учат работать с текстовыми документами, элементарной графикой, создавать презентации и так далее. Темам «алгоритмы» и «программирование» отводится небольшое количество часов, изучаются они поздно, в старших классах. Отсутствуют такие необходимые для развития логического мышления предметы, как алгоритмика, логика, которые были, когда учились более старшие поколения. Это замедляет формирование алгоритмического мышления и не способствует развитию интереса учащихся к программированию. Парадокс ещё заключается в том, что наибольший бал в заданиях Единого Государственного Экзамена и во всех значимых олимпиадах, приносят как раз задания на программирование.

программирования и IT - технологий, что очень актуально в современном образовательном процессе, также выявить способных к программированию детей и оказать помощь в профессиональном самоопределении.

Новизна программы.

Новизна представленной программы заключается, **во-первых**, в том, что она является первой ступенью непрерывного курса обучения детей программированию, дизайну и сайтостроению со ступени начального до старшего звена в «Школе программирования и дизайна».

В основе организации целостного педагогического процесса - известные принципы преемственности, последовательности и систематичности, но, в данном случае они предполагают такую организацию педагогического процесса, при которой существует взаимосвязь между образовательными программами, при которой последующий курс является более высокой ступенькой ранее проводимой работы, закрепляет и развивает достигнутое, поднимает обучающегося на более высокий уровень развития. Видение перспективы своего творческого развития способствует росту познавательного интереса.

«Школа программирования и дизайна» - это серия дополнительных общеобразовательных программ для детей 8 - 17 лет, основные направления которых - программирование, дизайн и сайтостроение.

Обучение построено по ступенчатому принципу: отучившись по программе «Юный программист» ребенок может продолжить обучение на более старших ступенях: «Компьютерная графика и дизайн» и «Web - дизайн и программирование». В зависимости от подготовки ребенка возможен переход сразу на какую-либо ступень.

Во-вторых, в сочетании известных педагогических форм, методов, приёмов организации образовательного процесса с инновационными (мультимедийные учебные пособия, флеш-анимации, видеоролики, вебкейсы и другие интернет - технологии), что стимулирует развитие интереса обучающихся. Созданные проекты размещаются на Международном портале Scratch, одаренные учащиеся работают над индивидуальными проектами, с которыми участвуют в конкурсах по программированию.

Отличительные особенности программы.

Отличительными особенностями программы «Юный программист» являются:

1. Проектный подход.

Использование метода проектов позволяет обеспечить условия для развития у учащихся навыков самостоятельной постановки задач и выбора оптимального варианта их решения, самостоятельного достижения цели, анализа полученных результатов с точки зрения решения поставленной задачи.

2. Межпредметность.

В программе прослеживается тесная взаимосвязь с математикой, физикой, географией, русским языком, музыкой и другими предметами школьного цикла. Знания, полученные на других предметах, логичным образом могут быть использованы при разработке проектов.

3. Пропедевтика.

Отличительной особенностью программы является ранний возраст

обучающихся программированию. Через разработку проектов учащиеся получают знания, обозначенные в программах более старших классов. Так, например, осваиваются основные алгоритмические конструкции (информатика), действия с десятичными дробями, отрицательными числами, понятие координатной плоскости (математика).

4. Коммуникация.

Огромным достоинством программы является возможность обучению навыкам работы в парах, группе, командах, создание коллективных проектов, чего практически невозможно достичь при изучении традиционных языков Бейсик и Паскаль. Обязательное условие — публичная презентация и защита проектов. Возможность увидеть результаты своего труда в Интернет также стимулирует интерес детей к обучению.

Обучение программированию по программе «Юный программист» осуществляется в свободно распространяемой среде Scratch. Для начала обучения детей программированию выбрана среда Scratch, так как данный язык очень прост для ребенка. Программы в Scratch не пишут, а собирают из готовых блоков, то есть снимается проблема работы с текстом программы, не нужно заучивать наизусть названия команд и уметь писать их без ошибок. Основной задачей становится разработка алгоритма и его воплощение в виде компьютерной игры, мультфильма, видеоролика. Освоив язык программирования Scratch ребятам будет легче изучать другие языки.

Педагогическая целесообразность программы заключается в том, что она является целостной и непрерывной в течение всего процесса обучения. Позволяет формировать у обучающихся не только логическое мышление, но и раскрывать в себе творческие возможности, развивать навыки работы с мультимедиа и самореализоваться в современном мире. Не смотря на очевидные преимущества обучения основам программирования в среде Scratch, в настоящее время они еще мало освоены, очень мало методических разработок и учебников по их освоению.

Цель программы:

создать комфортные условия для формирования и развития у обучающихся алгоритмического мышления в процессе изучения основ программирования в среде Scratch.

Задачи программы:

Образовательные:

- обучение основным базовым алгоритмическим конструкциям;
- освоение основных этапов решения задачи;
- обучение навыкам разработки, тестирования и отладки несложных программ;
- формирование навыков решения олимпиадных задач;
- знакомство с понятием проекта, его структуры, дизайна и разработки.

Развивающие:

- развивать познавательный интерес обучающихся;
- развивать творческое воображение, математическое и образное мышление учащихся;
- развивать умение работы с компьютерными программами;

- развивать творческое воображение, математическое и образное мышление учащихся;
- развивать умение работы с компьютерными программами;
- развивать навыки планирования проекта, умение работать в группе.

Воспитательные:

- воспитывать интерес к занятиям информатикой;
- воспитывать культуру общения между учащимися;
- воспитывать культуру безопасного труда при работе за компьютером;
- воспитывать культуру работы в глобальной сети.
- стимулирование интереса и склонности к выбору будущей профессии в сфере информатики и математики, а также в смежных областях.

Характеристика обучающихся.

Дополнительная общеобразовательная программа «Юный программист» предназначена для обучения детей в возрасте 8-13 лет. Учащиеся набираются по желанию. Наполняемость групп: число учащихся в группах - 12 человек.

Формы и режим занятий.

Основными формами организации образовательного процесса являются:

- лекции, в том числе с использованием наглядных средств обучения;
- беседы;
- викторины;
- дискуссии;
- практические занятия на компьютере;
- конкурсы;
- игры;
- самостоятельная работа учащихся;
- соревнования;
- защита проектов.

Программа реализуется в течение всего календарного года, включая каникулярное время.

Число занятий - 2 раза в неделю - 2 часа:

Работа с использованием компьютера, с перерывами на теорию, лекции, обсуждение проектов, эвристические беседы, дискуссии, а также перерывами.

2. ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ ПО ИТОГАМ ОБУЧЕНИЯ.

По окончании изучения теории учащиеся 1 года обучения должны знать:

- ✓ правила безопасной работы;
- ✓ что такое исполнитель, среда исполнителя, команды исполнителя;
- ✓ что такое алгоритм, скрипт, программа;
- ✓ типы алгоритмов (линейный, разветвляющийся, циклический);
- ✓ интерфейс программы Scratch;
- ✓ логические операции;
- ✓ координаты на плоскости;
- ✓ понятие проект, его структура и реализация в среде Scratch.

По окончании освоения практической части учащиеся 1 года обучения должны уметь:

- ✓ составлять алгоритмы любого типа;
- ✓ оформлять алгоритмы в изучаемой среде программирования;
- ✓ тестировать и отлаживать созданную программу;
- ✓ создавать личные проекты в среде Scratch;
- ✓ использовать сеть для обмена материалами работы.

По окончании изучения теории учащиеся 2 года обучения должны знать:

- ✓ общий подход к решению задач по программированию, разнообразие способов решения задач;
- ✓ основные виды алгоритмов;
- ✓ основы алгоритмического языка КуМир;
- ✓ графические возможности КуМир;
- ✓ понятие «хороший стиль программирования».

По окончании освоения практической части учащиеся 2 года обучения должны уметь:

- ✓ писать простейшие программы на алгоритмическом языке.
- ✓ создавать программы, рисующие несложные картинки.
- ✓ конструировать многомодульные программы.
- ✓ создавать читаемые программы (т.е. использовать «хороший стиль программирования»).

Формы промежуточной и итоговой аттестации.

Промежуточная и итоговая аттестация проводится в следующих формах подведения итогов:

- ✓ опрос;
- ✓ зачёт;
- ✓ тестирование;
- ✓ самостоятельная работа;
- ✓ открытое занятие для родителей;
- ✓ участие в тематических конкурсах, играх, соревнованиях.

Таблицы (КИМ), в которых фиксируются результаты аттестации по полугодиям в приложении.

Способы определения результативности.

С целью диагностики успешности освоения программы, достижений обучающихся и коррекции образовательного процесса применяются следующие методы:

- педагогическое наблюдение за обучающимися при выполнении заданий, в ходе работы над проектом;
- педагогический анализ выполнения творческих работ, результатов тестирования, анкетирования, участия в тематических играх, конкурсах, соревнованиях.
- педагогический мониторинг (диагностика развития личностных качеств обучающихся, диагностика освоения ЗУНов).

**СОДЕРЖАНИЕ ДОПОЛНИТЕЛЬНОЙ
ОБЩЕОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ ДЛЯ
ДЕТЕЙ
(1 ГОД ОБУЧЕНИЯ).**

Тема 1: Введение в мир программирования (2 часа).

Теория - 1 час

Знакомство. Цель, задачи программы. План работы на учебный год. Режим занятий. Первичный инструктаж по ОТ, ПБ, ГО и ЧС. Первичный инструктаж по темам: «Правила поведения в ГБОУ «НОШ №1 с.п.Экажево», «Охрана жизни и здоровья учащихся на учебных занятиях». Демонстрация примеров проектов, сделанных в среде Scratch.

Практика – 1 час

Анкетирование с целью выявления интересов и ожиданий. Входная диагностика. Собеседование.

Тема 2: Думай как компьютер (2 часа).

Теория - 1 час

Понятие исполнителя, алгоритма. Основные характеристики исполнителя. Способы управления исполнителем. Просмотр видеоролика «Алгоритм». Способы записи алгоритма.

Практика – 1 час

Тренировочные и творческие задания по алгоритмам, описанным различными способами. Игра «Перевозчик». Самостоятельная работа.

Тема 3: Что такое компьютерная программа (1 час).

Теория - 1 час

Компьютер как универсальный исполнитель. Понятие программы, назначение, и использование. Как стать программистом. Просмотр видеоролика «Программа, программист, программирование». Виды программ, отличие программы от скрипта.

Тема 4: Знакомство со средой Scratch (2 часа).

Теория - 1 час

Что такое Scratch. Установка и запуск Scratch. Основные элементы интерфейса программы Scratch. Меню и инструменты. Основные характеристики исполнителя, их особенности в среде исполнителя Scratch. Цветные блоки. Назначение блоков. Скрипты, порядок выполнения скриптов.

Практика – 1 час

Установка Scratch. на свой компьютер. Создание, сохранение и открытие проектов в среде Scratch. Настройка среды. Знакомство с проектами сообщества Scratch.

Тема 5: Спрайты. Библиотеки костюмов и сцен (2 часа).

Теория - 1 час

Спрайты. Информация спрайта. Что могут спрайты. Операции над спрайтами. Костюмы. Выбор костюмов. Создание декораций - фонов. Изменение внешнего вида спрайта и фона. Импорт изображений. Экспорт спрайтов и их использование в проектах.

Практика – 1 час

Творческие задания для детей - создание исполнителей и фонов в проекте: «Круглые рисунки». Создание костюмов. Пользуемся помощью интернета.

Тема 6: Графический редактор Scratch (3 часа).

Теория - 1 час

Растровая и векторная графика. Особенности работы с растровым и векторным режимами графического редактора Scratch. Общие инструменты рисования. Обозначение центра изображения. Установка прозрачных цветов. Инструменты растрового и векторного режимов рисования. Слои изображения. Группировка фигур.

Практика – 2 часа

Рисуем в Scratch в растровом режиме: домик, кувшинку, кораблик, рыбку, робота. Рисуем в векторном режиме: машинку, монстра. Сохраняем спрайты для их использования в проектах.

Тема 7: Учимся управлять спрайтами (2 часа).

Теория - 1 час

Основные команды движения и события. Коричневые блоки - команды группы «События» - «Когда щёлкнут по флажку» и «Когда клавиша на клавиатуре нажата». Синие блоки - команды группы «Движение». Знакомство с отрицательными числами. Движение задом наперед.

Практика – 1 час

Создание алгоритма первого проекта в среде Scratch. Анимация «Научим кота бегать», запуская скрипт разными событиями: клик мышкой, нажатие разных клавиш. Задание: «Автомобиль с пятью скоростями». Выполнение дополнительных заданий.

Тема 8: Создание спецэффектов - команды «Внешность» (3 часа)

Теория - 1 час

Фиолетовые блоки - «Внешность» - блоки управления внешностью для спрайтов и сцены. Что такое анимация. Анимирование костюма. Эффекты изображений: учим спрайты исчезать и появляться - телепортация с помощью эффектов, изменять цвет и т. д. Размер и видимость. Диалоги - спрайты умеют думать и говорить. Слои.

Практика – 2 часа

Задания: «Осьминог», «Танцующая балерина», «Бегущий человек», «Оживляем динозавра», «Программируем страшилки» - спрайты неожиданно появляются и исчезают, как привидения.

Тема 9: Навигация в среде Scratch (3 часа).

Теория - 1 час

Навигация. Координатная плоскость. Что такое система координат. Знакомимся с координатой X. Знакомимся с координатой Y. Перемещение

объектов по горизонтали. Перемещение объектов по вертикали. Что такое угол поворота. Крутой поворот. Вращение спрайтов. Блоки перемещения по координатной плоскости.

Практика – 2 часа.

Ориентируемся по координатам. Задания: расшифруй пословицы, определи координаты точек, нарисуй рисунок по координатам. Рисование по координатам в среде Scratch. Задания с использованием команд движения: «Движение царпки по периметру сцены», «Движение по диагонали на нажатие клавиш».

Тема 10: Команды группы «перо» (3 часа).

Теория - 1 час

Темно-зеленые блоки - группа «Перо». Спрайты умеют рисовать. Перьевая черепаха. Исследуем печать.

Практика – 2 часа.

Игры с пером. Рисуем разные фигуры, буквы, лесенки, штрихпунктирные линии, спирали, цветные пятна, узоры из фигур. Задания: «Конфетти», «След от полёта», «Паук плетёт паутину».

Тема 11: Команды управления. Простые циклы (3 часа).

Теория - 1 час

Желтые блоки - команды группы «Управление». Паузы. Простые циклы.

Цикл «Повторить». Запись цикла со счетчиком в виде блок – схемы.

Понятия: «Заголовок цикла», «Тело цикла». Циклы и эффект цвета. Циклы и эффект призрака. Вращение. Бесконечный цикл. Автоматическая печать.

Практика – 2 часа.

Учим спрайты совершать повторяющиеся действия. Создаём «Вечный двигатель». Усовершенствуем ранее созданные программы в Scratch.

Создаём новые: «Гонки», «Полет самолета», «Разноцветный экран».

Тема 12: Вложенные циклы (3 часа).

Теория - 1 час

Вложенные циклы. Работа с алгоритмами. Советы и секреты.

Практика – 2 часа.

Задания: «Вращающиеся квадраты», «Танцующий динозавр», «Снежинка», «Цветок», «Разные орнаменты», «Создаем рисунки из многоугольников».

Тема 13: Команды группы - «Звук»(3 часа).

Теория - 1 час

Добавление звуков к спрайтам. Проигрывание звука. Запись звука. Как проигрывать аудиофайлы. Переворачиваем звуки. Игра на барабанах и другие звуки. Создание своей музыки. Полный звукоряд. Устанавливаем темп и такт. Контроль громкости звука. Ноты. Как рисуют ноты.

Практика – 2 часа.

Задания: «Организуем ансамбль», «Танцы на сцене», «Фейерверки». Создаем песни: «Маленькой ёлочке», «Чижик-пыжик», «Гуси» и другие.

Тема 14: Звуковые события. Распознавание движения (2 часа).

Теория - 1 час

Коричневые блоки - команды группы «События» - «Когда громкость больше параметра», «Когда движение видео больше параметра», «Когда фон меняется».

Практика – 1 час.

Задания: «Учим кота реагировать на звук, на изменение фона и на распознавание движения».

Тема 15: Обмен сообщениями (3 часа).

Теория - 1 час

Как осуществить связь между спрайтами в виде беседы. Коричневые блоки - команды группы «События» - «Когда я получу сообщение», «Передать сообщение», «Передать сообщение и ждать». Создаем процедуры (подпрограммы) при помощи передачи сообщений.

Практика – 2 часа.

Задания: «Рисование квадратов случайных цветов в ответ на передачу сообщения», «Бесконечное движение между фонами», «Лампа», «Презентация». Создаем проекты: «Дюймовочка», или «Путешествие в страну безопасного интернета» или на свою тему.

Тема 16: Блоки Scratch: обзор и типы данных. (2 часа)

Теория - 1 час.

Командные блоки. Блоки-функции (репортеры). Блоки-триггеры.

Контрольные блоки. Что означает форма блока. Разновидности данных в Scratch. Числовые данные. Строковые данные.

Практика – 1 час.

Проект «Кот и птичка в клетке». Создание героев и декораций проекта.

Программирование проекта. Отладка программы.

Тема 17: Математические операторы (3 часа).

Теория - 1 час.

Светло-зелёные блоки - группа «Операторы». Арифметические операторы.

Случайные числа. Математические функции.

Практика – 2 часа.

Решаем задачи, используя операторы. Задания: «Простая арифметика со Скретчем», «Делаем кота большим и маленьким», «Рисующий волшебник», «Барабан», «Расширяющаяся спираль».

Тема 18: Проект «Генератор кубиков»(2 часа).

Практика – 2 часа.

Проект со случайными числами «Генератор кубиков». Рисование спрайтов и фона проекта. Программирование проекта.

Тема 19: Считывание и распознавание (3 часа).

Теория - 1 час.

Голубые блоки - команды группы «Сенсоры». Сенсоры восприятия внутренних и внешних событий: управление с клавиатуры, столкновения спрайтов, касания, сенсор общения с человеком, для управления временем в проектах и т. д.

Практика – 2 часа.

Использование блоков группы «Сенсоры». Создаём игры: «Кошки-мышки», «Игра с пончиком».

Тема 20: Истина или ложь (3 часа).

Теория - 1 час

Операторы сравнения. Логические операторы: блоки «и», «или», «не».

Объединение вопросов. Соблюдение условий. Использование логических операторов для проверки областей числовых значений.

Практика – 2 часа.

Самостоятельная работа: понятия «Истина» и «Ложь». Проекты с использованием блоков «и», «или», «не»: «Сравнение десятичных дробей», «Угадай мои координаты».

Тема 21: Принятие решений и ветвление (3 часа).

Теория - 1 час

Использование блока «если». Структура блока «если». Использование переменных как флагов. Использование блока «если/иначе». Структура блока «если/иначе». Вложенные блоки «если» и «если/иначе».

Практика – 2 часа.

Работа с алгоритмами с ветвлением. Проекты: «Осторожно лужи», «Управляемый робот», «Тренажер памяти».

Тема 22: Проект «Простой Paint»(2 часа).

Практика – 2 часа.

Рисуем интерфейс программы для рисования: фон, цветные карандаши и ластик. Создаём скрипты, проверяем работу программы.

Тема 23: Игра «Лабиринт»(2 часа).

Практика – 2 часа.

Создание игры - «Лабиринт» - три варианта.

Тема 24: Как не зациклится (3 часа).

Теория - 1 час

Сложные условия. Блок «повторять, пока не выполнится условие».. Создание блока «всегда если». Блок «ожидание». Стоп-команды: блок «стоп».

Практика – 2 часа.

Создание проектов: «Часы», «Стрельба по воробьям».

Тема 25: Команды группы - «Данные». Переменные (3 часа).

Теория - 1 час

Оранжевые блоки - группа команд «Данные». Понятие - переменная.

Создание, переименование и удаление переменной. Использование переменных. Присвоение и изменение значений. Сравнение переменных.

Константа. Датчики.

Практика – 2 часа.

Создание проектов: «С переменной время», «С переменной счёт», «С переменной прыжок», «Идеальный вес», «Оптические иллюзии».

Тема 26: Строки и списки (3 часа).

Теория - 1 час

Работа со словами. Строковые операторы. Подсчет специальных символов в строке. Создание списков. Команды управления списками. Доступ к элементам списка. Динамические списки.

Практика – 2 часа.

Игры со списками: «Угадай слово», «Замена букв».

Тема 27: Создание блоков (3 часа).

Теория - 1 час

Создание своего блока. Подпрограмма (процедура). Блоки с параметрами.

Переменные в подпрограммах. Вложенные процедуры.

Практика – 2 часа.

Создание игр и тестов: «Звёздное небо», «Психологический тест».

Тема 28: Создание мультфильмов в среде Scratch (3 часа).

Практика – 3 часа.

Мультфильмы по сказкам «Колобок», «Репка».

Тема 29: Создание комиксов в среде Scratch (3 часа).

Практика – 3 часа.

Создание комиксов «Охота за приведениями», «Мой город».

Тема 30: Создание интерактивных поздравительных открыток в среде Scratch (3 часа).

Практика – 3 часа.

Создание открыток «С днём рождения», «С Новым годом».

Тема 31: Создание презентаций в среде Scratch (3 часа).

Практика – 3 часа.

Презентации: «Мои увлечения», «Моя семья».

Тема 32: Создание компьютерных игр в среде Scratch (2 часа).

Практика – 2 часа.

Проект «Убегая от преследователя», «Пора в космос».

Тема 33: Создание интерактивных игр (3 часа).

Практика – 3 часа.

Игра «Бешенные обезьяны», «Интерактивное общение».

Тема 34: Создание интерактивных тестов (3 часа).

Практика – 3 часа.

3. Учебно-тематический план (1 год обучения).

№	Наименование раздела	Количество часов		
		Теория	Практика	Всего часов
1. Введение в мир программирования				
1	Введение в мир программирования	1	1	2
	Думай, как компьютер	1	1	2
	Что такое компьютерная программа	1		1
	Всего:	3	2	5
2. Основные приемы программирования и создания проектов в среде Scratch				
4	Знакомство со средой Scratch	1	1	2
5	Спрайты. Библиотеки костюмом и сцен	1	1	2
6	Графический редактор Scratch	1	2	3
7	Учимся управлять спрайтами	1	1	2
8	Создание срецэффектов-команды «Внешность».	1	2	3
9	Навигация в среде Scratch	1	2	3
10	Команды группы «Перо».	1	2	3
11	Команды управления. Простые циклы	1	2	3
12	Вложенные циклы.	1	2	3
13	Команды группы - «Звук».	1	2	3
14	Звуковые события. Распознавание	1	1	2

	движения.			
15	Обмен сообщениями.	1	2	3
16	Блоки Scratch: обзор и типы данных.	1	1	2
17	Математические операторы.	1	2	3
18	Проект «Генератор кубиков»		2	2
19	Считывание и распознавание.	1	2	3
20	Истина или ложь.	1	2	3
21	Принятие решений и ветвление.	1	2	3
22	Проект «Простой Paint».	1	2	3
23	Игра «.лабиринт»		2	2
24	Как не заикнется.	1	2	3
25	Команды группы - «Данные». Переменные.	1	2	3
26	Строки и списки	1	2	3
27	Создание блоков	1	2	3
	Всего	22	43	65
3. Разработка групповых проектов в среде Scratch				
28	Создание мультфильмов в среде Scratch		3	3
29	Создание комиксов в среде Scratch		3	3
30	Создание интерактивных поздравительных открыток в среде Scratch		3	3
31	Создание презентаций в среде Scratch		3	3
32	Создание		2	2

	компьютерных игр в среде Scratch			
33	Создание интерактивных игр		3	3
34	Создание интерактивных тестов		3	3
35	Создание музыкального клипа в среде Scratch		3	3
36	Создание учащимися разных проектов		3	3
	Итого		26	26
4. Выполнение творческой работы-создание личного проекта в среде Scratch				
	Разработка личного проекта	1		1
	Дизайн проекта		3	3
	Программируем проект		3	3
	Тестирование и отладка проекта		2	2
	Защита проекта		3	3
	Всего	1	11	12
	Итого	26	82	108

Методическое обеспечение программы

В программе применяются следующие методы, позволяющие дать детям первоначальные основы программирования:

- □ Словесные методы позволяют быстро передавать детям информацию, ставить перед ними учебную задачу, указывать пути ее решения. Это лекции, беседы, дискуссии. Словесные методы и приемы сочетаются с наглядными, игровыми, практическими методами, делая последние более результативными.
- □ Наглядные методы: Наглядность оживляет процесс обучения, способствует побуждению у детей интереса к занятию. В качестве наглядности используются презентации, мультимедийные учебные пособия, флеш-анимации, видеоролики, веб-кейсы и другие интернет - технологии, демонстрационный раздаточный материал, карточки.
- □ Практические методы: Нельзя научить детей, только показывая и рассказывая, не предлагая самим детям каких - либо действий. Ребенок овладевает опытом только тогда, когда сам участвует в практической деятельности, практической работы учащихся за компьютером. При решении задач используется семинарское занятие
- □ Игровые методы: позволяют осуществлять учебные задачи в атмосфере легкости и заинтересованности, активности детей. Используются игровые задания, создание игр на компьютере, метод интерактивной игры, дидактические игры, и упражнения, выполнив которые ребенок легко может усвоить правила поведения, технику безопасности, гимнастику для глаз.

Формы обучения по программе:

- □ Фронтальная - подача учебного материала всему коллективу детей, на этих занятиях важен «эффект эмоционального воздействия и сопереживания», что приводит к повышению умственной активности, побуждает ребенка к самовыражению (интегрированные и итоговые занятия, интеллектуальные игры).
- □ Индивидуальная - используется при возникновении затруднения, не уменьшая активности детей и содействуя выработке навыков самостоятельной работы. В индивидуальных занятиях нуждаются дети с явно выраженными способностями к той или иной деятельности, дети с доминирующим познавательным интересом.
- □ Подгрупповая - предоставляется возможность самостоятельно построить свою деятельность на основе принципа индивидуализации и сознательности и активности, ощутить помощь со стороны друг друга, учесть возможности каждого на конкретном этапе деятельности. Всё это способствует более быстрому выполнению задания.

4. Список используемых источников.

для педагога:

1. Патаракин Е. Д. Руководство для пользователя среды Scratch. Версия 0.2, 2007г.
2. Патаракин Е. Д. Учимся готовить в среде Скретч (Учебно-методическое пособие). М: Интуит.ру, 2008 г.
3. Рындак В. Г., Дженжер В. О., Денисова Л. В. Проектная деятельность школьника в среде программирования Scratch: учебно-методическое пособие — Оренбург, 2009 г.
4. Пашковская Ю. В. Творческие задания в среде Scratch: рабочая тетрадь для 5 - 6 классов/ Ю. В. Пашковская.- М.:БИНОМ. Лаборатория знаний, 2014. - 200 с. :ил.
5. Л. А. Анеликова, О. Б. Гусева Программирование на алгоритмическом языке КуМир. Под редакцией Кушниренко А. Г. - М.: СОЛОН-ПРЕСС, 2013. - 48 с.: ил. - (Элективный курс Профильное обучение).
6. Еремин Е. А. Газета « Информатика». Среда Scratch - первое знакомство. - М.: Первое сентября, 2008 - № 20 (573) - стр. 16-28.
7. У. Сэнд, К. Сэнд Hello World! Занимательное программирование. — СПб.: Питер, 2016. — 400 с.: ил. — (Серия «Вы и ваш ребенок»).
8. Свейгарт, Эл. Автоматизация рутинных задач с помощью Python: практическое руководство для начинающих.: Пер. с англ.—М.: ООО —ИД. Вильямс”, 2016.—592с.: ил.—Парад. тит. англ.
9. Мэтис Эрик Изучаем Python. Программирование игр, визуализация данных, веб-приложения. — СПб.: Питер, 2017. — 496 с.: ил. — (Серия «Библиотека программиста»).

для ученика:

1. Торгашева Ю. В. Первая книга юного программиста. Учимся писать программы на Scratch. — СПб.: Питер, 2016. — 128 с.: ил. — (Серия «Вы и ваш ребенок»).
2. Программирование для детей/К. Вордерман, Дж. Вудкок, Ш. Макаманус и др. ; пер. с англ. С. Ломакина. - М. :Манн, Иванов и Фербер, 2015. - 224 с. :ил.
3. Пэйн, Брайсон. Python для детей и родителей / БрайсонПэйн ; [пер. с англ. МА. Райтмана]. — Москва : Издательство «Э», 2017. — 352 с. : ил. — (Программирование для детей).