

**МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ СРЕДНЯЯ ШКОЛА №33 Г.ЛИПЕЦКА
ИМЕНИ П.Н. ШУБИНА**

Рассмотрена
Педагогический совет
Протокол № 13 от 31.05.2023



Утверждаю
Директор МБОУ СШ №33 г. Липецка
Л.Л. Власова
Приказ № 218 от 31.05.2023

**Дополнительная общеразвивающая программа
технической направленности
«Программирование в Scratch»**

Возраст детей, на которых рассчитана программа- 9-10 лет
Срок реализации – 1 учебный год

Составители
Тюленёва Н.П., Чиркова Л.В.,
учителя информатики
МБОУ СШ №33 г.Липецка

2023-2024 учебный

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Сегодня наука и технология развиваются столь стремительно, что образование зачастую не успевает за ними. Одним из самых известных неформальных способов организации внеучебной образовательной деятельности является метод проектов. Самым подходящим инструментом для организации такой деятельности является среда Scratch, которая есть серьезное и современное направление компьютерного дизайна и анимации. Овладев даже минимальным набором операций, самый неискущённый пользователь может создавать законченные проекты. Scratch - это самая новая среда, которая позволяет детям создавать собственные анимированные и интерактивные истории, презентации, модели, игры и другие произведения. Работа в среде Scratch позволяет, с одной стороны, организовать среду для самореализации и самоутверждения учащихся, и, с другой стороны, сформировать у них тягу к творчеству и знаниям и дать подходящие средства её реализации. Быть успешным в такой среде становится проще.

Дополнительную общеразвивающую программу «Программирование в Scratch» можно рассматривать как инструмент для творчества, оставив программирование на втором плане. Обучающиеся могут сочинять истории, рисовать и оживлять на экране придуманных ими персонажей, учиться работать с графикой и звуком. Применений возможностям Scratch можно найти множество: в этой среде легко создавать анимированные открытки, мини-игры, мультфильмы. В результате выполнения простых команд может складываться сложная модель, в которой будут взаимодействовать множество объектов, наделенных различными свойствами. Начальный уровень программирования настолько прост и доступен, что Scratch рассматривается в качестве средства обучения не только старших, но и младших обучающихся.

Актуальность программы состоит в том, что мультимедийная среда Scratch позволяет сформировать у детей стойкий интерес к программированию, отвечает всем современным требованиям объектно-ориентированного программирования. Среда Scratch позволяет сформировать навыки программирования, раскрыть технологию программирования.

Новизна дополнительной общеразвивающей программы «Программирование в Scratch» 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 по программированию практически значимой для современного учащегося, т.к. дает возможность увидеть практическое назначение алгоритмов и программ, что будет способствовать развитию интереса к профессиям, связанным с программированием.

Направленность дополнительной общеразвивающей программы «Программирование в Scratch» – техническая. Направленность данной программы заключается в основах раннего программирования младших школьников.

Образовательная деятельность по дополнительной общеразвивающей программе «Программирование в Scratch» направлена на:

- формирование и развитие творческих способностей учащихся;
- удовлетворение индивидуальных потребностей учащихся в интеллектуальном, нравственном и интеллектуальном развитии,
- выявление, развитие и поддержку талантливых учащихся, а также лиц, проявивших выдающиеся способности;
- профессиональную ориентацию учащихся;
- создание и обеспечение необходимых условий для личностного развития, профессионального самоопределения учащихся;
- социализацию и адаптацию учащихся к жизни в обществе;
- формирование общей культуры учащихся;
- удовлетворение иных образовательных потребностей и интересов учащихся, не противоречащих законодательству Российской Федерации, осуществляемых за пределами федеральных государственных образовательных стандартов и федеральных государственных требований.

Педагогическая целесообразность данной программы состоит в том, что, изучая программирование в среде Scratch, у учащихся формируется не только логическое мышление, но и навыки работы с мультимедиа; создаются условия для активного, поискового учения, предоставляются широкие возможности для разнообразного программирования.

Цели и задачи обучения

Цель: создание условий для развития личности учащихся, способных к творческому самовыражению через овладение основами программирования.

Данная программа решает следующие основные задачи:

Обучающие:

1. Овладеть навыками составления алгоритмов.
2. Изучить функциональность работы основных алгоритмических конструкций.
3. Сформировать представление о профессии «программист».
4. Сформировать навыки разработки программ.
5. Познакомить с понятием проекта и алгоритмом его разработки.
6. Сформировать навыки разработки проектов: интерактивных историй, квестов, интерактивных игр, обучающих программ, мультфильмов, моделей и интерактивных презентаций.

Развивающие:

1. Способствовать развитию критического, системного, алгоритмического и творческого мышления.
2. Развивать внимание, память, наблюдательность, познавательный интерес.
3. Развивать умение работать с компьютерными программами и дополнительными источниками информации.
4. Развивать навыки планирования проекта, умение работать в группе.

Воспитательные:

1. Формировать положительное отношение к информатике и ИКТ.
2. Развивать самостоятельность и формировать умение работать в паре или в малой группе.
3. Формировать умение, демонстрировать результаты своей работы.

Принципы обучения, реализуемые программой:

- сознательности;
- наглядности;
- доступности;
- связи теории с практикой;
- творческой активности.

Важным условием развития творческого и познавательного интереса учащегося является индивидуальный подход к нему в процессе обучения.

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

Содержание практических занятий ориентировано не только на овладение учащимися 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.

Методы, используемые при реализации программы:

•**Вербальный метод** основан на богатстве, выразительности и многоплановости устной речи. Основными приемами и способами вербального обучения являются рассказ, объяснение, лекция, беседа, дискуссия, инструктирование, изложение, повествование, описание, рассуждение.

•**Иллюстративный метод** заключается в предъявлении обучающимся информации способом демонстрации разнообразного наглядного материала, в том числе с помощью технических средств.

•**Репродуктивный метод** - многократное воспроизведение (репродуцирование) действий, направлен на формирование навыков и умений. Этот метод предполагает как самостоятельную работу обучающихся, так и совместную работу с педагогом.

•**Метод проблемного изложения** - рассчитан на вовлечение ученика в познавательную деятельность в условиях словесного обучения, когда учитель сам ставит проблему, сам показывает пути ее решения, а учащиеся внимательно

следят за ходом мысли учителя, размышляют, переживают вместе с ним и тем самым включаются в атмосферу научно-доказательного по искового решения.

•**Частично-поисковые, или эвристические методы**, используются для подготовки учащихся к самостоятельному решению познавательных проблем, для обучения их выполнению отдельных шагов решения и этапов исследования.

•**Исследовательские методы** - способы организации поисковой, творческой деятельности учащихся по решению новых для них познавательных проблем.

•**Самостоятельная работа обучающихся с литературой** по теме является одним из способов самостоятельного приобретения, закрепления и углубления необходимых специальных знаний.

•**Алгоритмический метод** направлен на решение задач в строго определенной последовательности. Этот метод можно позволяет придать обучающимся уверенность в успехе и ориентирует их на идеальный конечный результат.

Возраст детей, участвующих в реализации программы «Программирование в Scratch» - учащиеся 3-х классов (9-10 лет).

Сроки реализации дополнительной общеразвивающей программы «Программирование в Scratch»: 1 учебный год.

В группу принимаются все желающие.

Формы занятий: лекция, объяснение, беседа, практическая работа. Все занятия направлены на развитие интереса учащихся к предмету, на расширение представлений об изучаемом материале. Занятия проводятся с группой учащихся достаточно однородной с точки зрения обучаемости.

Режим занятий: программа реализуется в общеобразовательном учреждении, количество занятий в неделю – 1; за учебный год – 29.

Учебный план общеразвивающей программы «Программирование в Scratch»

№	Название курса	Часов в неделю	Всего часов за год	Форма аттестации
1	Программирование в Scratch	1	29	Защита проекта

Календарный учебный график

Начало учебного года 29.09.2023

Окончание учебного года 25.05.2024

Продолжительность учебного года 29 учебные недели

Продолжительность учебных четвертей:

I четверть – 4 учебных недель;

II четверть – 8 учебных недель;

III четверть – 11 учебных недель;

IV четверть – 6 учебных недель.

Количество учебных дней в неделю - 5 дней.

Форма организации образовательного процесса: по четвертям.

Сроки и продолжительность каникул:

осенние:

- каникулы с 28 октября по 6 ноября 2023 года (10 дней), начало второй учебной четверти – 7 ноября 2023 года;

зимние:

- каникулы с 30 декабря 2023 года по 8 января 2024 года (10 дней), начало третьей учебной четверти – 9 января 2024 года;

весенние:

- каникулы с 23 марта по 31 марта 2024 года (9 дней), начало четвертой учебной четверти – 1 апреля 2024 года.

летние:

- каникулы с 24 мая по 31 августа 2024 года.

27 апреля 2024 года (суббота) обучение по расписанию понедельника.

Нерабочие праздничные дни (в связи с государственными праздниками):

23 февраля 2024 года, 8 марта 2024 года, 1, 9, 10 мая 2024 года.

Сроки проведения промежуточной аттестации:

13. 05. 2024 – 23. 05. 2024

Планируемые результаты освоения программы

Личностные:

- широкие познавательные интересы, инициатива и любознательность, мотивы познания и творчества; готовность и способность учащихся к саморазвитию и реализации творческого потенциала в духовной и предметно-продуктивной деятельности за счет развития их образного, алгоритмического и логического мышления;
- готовность к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики и ИКТ;
- интерес к информатике и ИКТ, стремление использовать полученные знания в процессе обучения другим предметам и в жизни;
- способность связать учебное содержание с собственным жизненным опытом и личными смыслами, понять значимость подготовки в области информатики и ИКТ в условиях развития информационного общества;
- готовность к самостоятельным поступкам и действиям, принятию ответственности за их результаты; готовность к осуществлению индивидуальной и коллективной информационной;
- владение умениями организации собственной учебной деятельности, включающими: целеполагание как постановку учебной задачи на основе соотнесения того, что уже известно, и того, что требуется установить;

- планирование – определение последовательности промежуточных целей с учетом конечного результата, разбиение задачи на подзадачи, разработка последовательности и структуры действий, необходимых для достижения цели при помощи фиксированного набора средств;
- прогнозирование – предвосхищение результата;
- контроль – интерпретация полученного результата, его соотнесение с имеющимися данными с целью установления соответствия или несоответствия (обнаружения ошибки);
- коррекция – внесение необходимых дополнений и корректив в план действий в случае обнаружения ошибки;
- оценка – осознание учащимся того, насколько качественно им решена учебно-познавательная задача;
- владение основными универсальными умениями информационного характера: постановка и формулирование проблемы;
- поиск и выделение необходимой информации, применение методов информационного поиска;
- структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных;
- умение использовать термины «информация», «сообщение», «данные», «алгоритм», «программа»; понимание различий между употреблением этих терминов в обыденной речи и в информатике;
- умение составлять линейные, разветвляющиеся и циклические алгоритмы управления исполнителями на языке программирования Скретч;
- умение использовать логические значения, операции и выражения с ними;
- овладение понятиями класс, объект, обработка событий;
- умение формально выполнять алгоритмы, описанные с использованием конструкций ветвления (условные операторы) и повторения (циклы), вспомогательных алгоритмов;
- умение создавать и выполнять программы для решения несложных алгоритмических задач в программе Скретч;
- способность к избирательному отношению к получаемой информации за счет умений ее анализа и критичного оценивания; ответственное отношение к информации с учетом правовых и этических аспектов ее распространения;
- способность и готовность к принятию ценностей здорового образа жизни за счет знания основных гигиенических, эргономических и технических условий безопасной эксплуатации средств ИКТ.

Метапредметные:

- владение умениями организации собственной учебной деятельности, включающими: целеполагание как постановку учебной задачи на основе соотнесения того, что уже известно, и того, что требуется установить;
- планирование – определение последовательности промежуточных целей с учетом конечного результата, разбиение задачи на подзадачи, разработка последовательности и структуры действий, необходимых для достижения цели при помощи фиксированного набора средств;
- прогнозирование – предвосхищение результата;
- контроль – интерпретация полученного результата, его соотнесение с имеющимися данными с целью установления соответствия или несоответствия (обнаружения ошибки);
- коррекция – внесение необходимых дополнений и корректив в план действий в случае обнаружения ошибки;
- оценка – осознание учащимся того, насколько качественно им решена учебно-познавательная задача;
- владение основными универсальными умениями информационного характера: постановка и формулирование проблемы;
- поиск и выделение необходимой информации, применение методов информационного поиска;
- структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных условий;
- самостоятельное создание алгоритмов деятельности при решении проблем творческого и поискового характера;
- владение основами продуктивного взаимодействия и сотрудничества со сверстниками и взрослыми: умение правильно, четко и однозначно сформулировать мысль в понятной собеседнику форме;
- умение осуществлять совместную информационную деятельность, в частности при выполнении проекта;
- использование коммуникационных технологий в учебной деятельности и повседневной жизни.

Формирование ИКТ-компетенций обучающихся

Обучающийся научится:

- целенаправленно искать и использовать информационные ресурсы, необходимые для решения учебных и практических задач с помощью средств ИКТ;
- выбирать, строить и использовать адекватную информационную модель для передачи своих мыслей средствами естественных и формальных языков в соответствии с условиями коммуникации;

- выделять информационный аспект задачи, оперировать данными, использовать модель решения задачи;

- использовать компьютерные технологии.

Обучающийся получит возможность научиться:

- использовать информацию с учетом этических и правовых норм;
- создавать информационные ресурсы разного типа и для разных аудиторий,

соблюдать информационную гигиену и правила информационной безопасности.

Формирование основ учебно-исследовательской и проектной деятельности

Обучающийся научится:

- проявлять самостоятельность, инициативность, ответственность;
- выбирать адекватные стоящей задаче средства, принимать решения.

Обучающийся получит возможность научиться:

- способности к разработке нескольких вариантов решений, к поиску нестандартных решений, поиску и осуществлению наиболее приемлемого решения.

Стратегия смыслового чтения и работы с текстом

Обучающийся научится:

- находить в тексте требуемую информацию (в соответствии с целями своей

деятельности);

- ориентироваться в содержании текста, понимать целостный смысл текста, структурировать текст;

- устанавливать взаимосвязь описанных в тексте событий, явлений, процессов;

- резюмировать главную идею текста.

Обучающийся получит возможность научиться:

- преобразовывать текст, «переводя» его в другую модальность, интерпретировать

- текст (художественный и нехудожественный – учебный, научно-популярный, информационный, текст non-fiction);

- критически оценивать содержание и форму текста.

Способы определения результативности:

- наблюдение;

- беседы индивидуальные и групповые;

- опрос;

- включения обучающихся в деятельность по освоению программы, выполнение заданий.

Для фиксации результатов контроля используется диагностическая карта мониторинга результатов обучения по дополнительной общеразвивающей программе (Приложение 1), заполняемая 2 раза (декабрь, апрель) за период обучения по программе. Экспертом в оценке уровня освоения программы обучающимися выступает педагог.

Обработка и интерпретация результатов:

Каждый показатель мониторинга оценивается от 1 до 3 баллов: 1 балл – ниже базового уровня, 2 балла – базовый уровень, 3 балла – выше базового уровня.

Критерии оценки уровня результативности:

- 1 - 6 баллов – программа освоена на низком уровне (освоение обучающимся менее 50% содержания дополнительной общеразвивающей программы);
- 7-12 баллов – программа освоена на базовом уровне (освоение учащимся от 50% до 70% содержания дополнительной общеразвивающей программы);
- 13-18 баллов – программа освоена на высоком уровне (освоение обучающимся более 70% содержания дополнительной общеразвивающей программы).

ОПИСАНИЕ УСЛОВИЙ РЕАЛИЗАЦИИ ПРОГРАММЫ

Организационно-педагогические условия:

- классный кабинет с мебелью;
- рабочее место педагога;
- интерактивная доска (мультимедийный проектор и экран);
- схемы, таблицы;
- канцелярские принадлежности.
- ноутбуки;
- доступ к сети Интернет;
- программное обеспечение: cratch.

Данную программу реализуют педагоги, имеющие высшее профессиональное образование или среднее профессиональное образование в области, соответствующей профилю кружка, секции, студии, клубного и иного детского объединения без предъявления требований к стажу работы, либо высшее профессиональное образование или среднее профессиональное образование и дополнительное профессиональное образование по направлению «Образование и педагогика» без предъявления требований к стажу работы.

Методическое обеспечение программы:

- мультимедийные презентации;
- пособия для групповой и индивидуальной работы;
- таблицы;
- аудио и видеозаписи;

- 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»).

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Герасимова Т. Б. Организация проектной деятельности в школе. // Преподавание истории в школе. 2007. № 5. С. 17–21.
2. Голиков Д.И. «Scratch для юных программистов», «БХВ-Петербург», Санкт-Петербург, 2017.
3. Краля Н. А. Метод учебных проектов как средство активизации учебной деятельности учащихся: Учебно-методическое пособие / Под ред. Ю. П. Дубенского. Омск: Изд-во ОмГУ, 2005. 59 с.
4. Матвеева Н. В. Информатика и ИКТ. 3 класс: методическое пособие / Н. В. Матвеева, Е. Н. Челак, Н. К. Конопатова, Л. П. Панкратова. М.: БИНОМ. Лаборатория знаний, 2009. 420 с.
5. Патаракин Е. Д. Учимся готовить в среде Скретч (Учебно-методическое пособие). М: Интуит.ру, 2008. 61 с.
6. Скретч [Электронный ресурс] // Материал с Wiki-ресурса Letopisi.Ru — «Время вернуться домой». URL: <http://letopisi.ru/index.php/Скретч>
7. Хохлова М. В. Проектно-преобразовательная деятельность младших школьников. // Педагогика. 2004. № 5. С. 51–56.

Интернет ресурсы:

1. Школа Scratch [Электронный ресурс] // Материал с Wiki-ресурса Letopisi.Ru — «Время вернуться домой». URL: http://letopisi.ru/index.php/Школа_Scratч
2. Scratch | Home | imagine, program, share [сайт]. URL: <http://scratch.mit.edu>
3. Scratch | Галерея | Gymnasium №3 [сайт]. URL: <http://scratch.mit.edu/galleries/view/54042>

Рабочая программа программы «Программирование в Scratch».

СОДЕРЖАНИЕ ПРОГРАММЫ

Темы 1. «Интерфейс программы Scratch» (1 час).

Теоретическая часть: Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch. История создания среды Scratch. Основные базовые алгоритмические конструкции (линейные алгоритмы, с условным оператором, циклического типа с предусловием и постусловием) и их исполнение в среде Scratch. Понятие

исполнителя, алгоритма и программы, их назначение, виды и использование. Виды управления исполнителем. Способы записи алгоритма. Основные характеристики исполнителя. Система команд исполнителя. Понятие проект, его структура и реализация в среде Scratch. Основные компоненты проекта Scratch: спрайты и скрипты. Принцип создания анимации и движения объектов. Листинг программы. Сцена. Текущие данные о спрайте. Стилль поворота. Закладки. Панель инструментов. Новый спрайт. Координаты мышки. Режим представления. Окно скриптов. Окно блоков. Блоки стека. Блоки заголовков. Блоки ссылок. Самодостаточные и открытые скрипты.

Практическая часть: Использование интерфейса программы Scratch и панели инструментов.

Тема 2. «Начало работы в среде Scratch» (2 часа).

Теоретическая часть: Сцена. Редактирование фона. Добавление фона из файла. Сцена. Ширина и высота сцены. Текущие координаты объекта. Редактирование текущего фона. Вставка нового фона из файла. Вставка стандартного фона из библиотечного модуля среды. Рисование фона в графическом редакторе. Создание нескольких фонов в одной сцене. Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов. Стандартный объект. Спрайты. Список спрайтов. Редактор рисования для создания новых спрайтов. Инструменты рисования (кисточка, линия, текст, эллипс) и редактирования объекта (ластик, заливка, поворот, выбор, печать, пипетка). Центрирование костюма. Масштабирование спрайта. Загрузка на сцену спрайтов из стандартной коллекции среды Scratch. Вставка спрайтов из файлов форматов JPG, BMP, PNG, GIF. Выбор случайного спрайта. Удаление спрайтов.

Практическая часть: Создание фона сцены на выбранную учащимся тему. Создание фона сцены и прорисовка основных спрайтов для Scratch-истории.

Тема 3. «Основные скрипты программы Scratch» (13 часов).

Теоретическая часть: Синий ящик – команды движения. Темно-зеленый ящик – команды рисования. Команды – идти; повернуться направо (налево); повернуть в направлении; повернуться к; изменить x (y) на; установить x (y) в; если край, оттолкнуться. Принципиальное различие действия команд идти и плыть в. Назначение сенсоров положение x, положение y и направлении. Команды – очистить, опустить перо, поднять перо, установить цвет пера, изменить цвет пера на, установить цвет пера, изменить тень пера, установить тень пера, изменить размер пера на, установить размер пера, печать. Фиолетовый ящик – внешний вид объекта. Оживление объекта с помощью добавления костюмов. Костюмы спрайта. Копирование и редактирование костюма спрайта с помощью редактора рисования. Переупорядочивание костюмов. Команды – перейти к костюму, следующий костюм, говорить...в течении...секунд, сказать, думать, думать...секунд, изменить... эффект на, установить эффект...в значение,

убрать графические эффекты, изменить размер на, установить размер, показаться, спрятаться, перейти в верхний слой, перейти назад на...1 слоев. Назначение сенсоров костюм и размер. Понятие раскадровки движения. Изменение костюма спрайта для имитации движения. Желтый ящик – контроль. Лиловый ящик – добавление звуков. Кнопка с зеленым флажком и ее назначение. Управление последовательностью выполнения скриптов. Понятие управляющих сообщений. Команды – передать, передать и ждать, когда я получу. Скрипты для создания условных конструкций программы – если, если...или. Скрипты для управления циклами – всегда, повторить, всегда, если, повторять до.. Команды – когда клавиша...нажата, когда щелкнут по, ждать...секунд, ждать до, остановить скрипт, остановить все. Загрузка звуков из стандартной коллекции и из файлов жесткого диска. Запись звука через микрофон. Принципиальная разница работы команд играть звук и играть звук до завершения. Команды – остановить все звуки, барабану играть...тактов, оставшиеся...тактов, ноту...играть...тактов, выбрать инструмент, изменить громкость, установить громкость, изменить темп на, установить темп. Назначение сенсоров громкость и темп. Использование в программах условных операторов. Базовая конструкция ветвление, назначение, виды (полная и неполная форма). Понятие условия. Изменение порядка выполнения скриптов в зависимости от условия. Разветвление листинга программы. Скрипты условных операторов. Использование неполной формы ветвления в системе Scratch. Функциональность работы циклов. Цикличность выполнения действий в зависимости от поставленных условий. Циклы с фиксированным числом повторений. Заголовок цикла. Тело цикла. Циклы с условным оператором. Заголовок цикла. Тело цикла. Предусловие и постусловие. Зацикливание. Зеленый ящик – операторы. Использование арифметических и логических блоков вместе с блоками управления. Числа. Строинги. Логические величины. Логические выражения. Арифметические операции. Логические операции. Операции сравнения. Команды для работы со строингами – слить, буква...в, длина строки. Команда выдать случайное от...до. Использование арифметических и логических блоков в листинге программы. Просмотр полученного результата. События. Оранжевый ящик – переменные. События в проектах Scratch. Понятие переменных и необходимость их использования в листинге программы. Глобальные и локальные переменные. Имя переменной и правила его формирования. Команды для переменных - поставить...в, изменить...на, показать переменную, спрятать переменную. Удаление переменных. Создание счетчиков с помощью переменных. Списки. Команды работы со списками – добавить...к, удалить...из, поставить...в...из, заменить элемент...в...на, элемент...из, длина списка. Голубой ящик – сенсоры. Ввод-вывод данных. Понятие сенсора. Правила применения и область действия команд касается, касается цвета и цвет.касается. Функционал команды спросить...и ждать. Сенсоры мышка по x, мышка по y, мышка нажата?, клавиша...нажата?,

расстояние до, перезапустить таймер. Сенсоры, значение которых можно выводить на экран – ответ, таймер, громкость, громко? ...значение сенсора и сенсор.... Необходимость ввода данных для их обработки в программе. Ввод данных с помощью команды спросить. Вывод конечного результата обработки с помощью команд говорить и сказать.

Практическая часть: Создание программ для передвижения спрайтов по сцене. Создание программ для рисования различных фигур. Создание программы для управления внешним видом объекта. Создание Scratch-историй с имитацией хождения и движения объектов. Создание программ с элементами управления объектом. Озвучивание Scratch-историй. Создание программ с изменением последовательного выполнения скриптов при наличии условий. Создание программ с использованием операций сравнения данных. Создание программ с использованием циклов с фиксированным числом повторений. Создание программ с использованием циклов с предусловием и постусловием. Создание программ с использованием арифметических данных и логических операций. Создание списков и необходимость их использования в проектах Scratch. Добавление в список данных. Удаление данных из списка. Удаление списка. Разработка сценария Scratch-историй с несколькими событиями. Создание проектов с использованием глобальных и локальных переменных. Создание программ-тестов по принципу сравнения данных из нескольких списков. Создание проектов с использованием значений сенсоров и команды спросить. Создание программ для обработки данных пользователя с выводом на экран конечного результата.

Тема 4. «Работа с несколькими объектами. Синхронизация их работы» (3 часа).

Теоретическая часть: Последовательность и параллельность выполнения скриптов. Последовательные и параллельные потоки в программах Scratch. Одновременная и попеременная работа нескольких исполнителей. Взаимодействие между спрайтами. Управление через обмен сообщениями.

Решение проблемы появления новых исполнителей только после того, как старые исполнители выполнили свои действия. Взаимодействие спрайтов с неподвижными объектами с помощью команд *касается* и *касается цвета*. Взаимодействие спрайтов с помощью команд *передать* и *когда я получу*. Использование сообщений для создания событий.

Практическая часть: Создание Scratch-историй с одновременной и попеременной работой нескольких исполнителей. Создание Scratch-историй с взаимодействием нескольких исполнителей и неподвижных объектов. Создание Scratch-историй с взаимодействием нескольких исполнителей.

Тема 5. «Использование программы Scratch для создания мини-игр» (10 часов).

Теоретическая часть: Виды компьютерных игр. Алгоритмическая разработка листинга программы. Компьютерные игры – вред или польза. Виды компьютерных игр. Этапы разработки игр программистами. Разработка базовых спрайтов для игры. Формирование базовых скриптов. Логика создания персонажей для игры. Перевод алгоритма, написанного на естественном языке, в коды Scratch. Синхронизация работы скриптов для разных спрайтов. Переход из одной сцены в другую. Создание интерфейса игры. Односторонний (без возможности вернуться назад) переход из одного пространства в другое. Понятие интерфейса. Элементы интерфейса. Основные принципы дизайна интерфейсов. Обратная связь. Необходимые элементы меню. Сообщество Scratch в Интернете. Просмотр и публикация проектов. Правила работы в сети. Интернет-сообщества. Сообщество Scratch. Регистрация на сайте. Использование заимствованных кодов и объектов. Авторские права. Публикация проектов Scratch.

Практическая часть: Алгоритмическая разработка проекта, запись на естественном языке событий и точек взаимодействия героев будущей игры. Разработка и создание основных спрайтов и их костюмов для будущей игры. Разработка скриптов для спрайтов и объектов. Доработка основного листинга программы с целью установления связей между спрайтами. Тестирование и отладка программы. Создать программу для перемещения объекта по игровой карте и разработать интерфейс для Scratch-проекта. Регистрация на сайте сообщества Scratch. Просмотр проектов сообщества и публикация собственных проектов. Разработка и защита творческого проекта. Разработка и создание программы с использованием подготовленных материалов. Тестирование и отладка проекта. Защита проекта.

ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№ п/п	Наименование темы	В том числе		Общее кол-во часов
		теория	практик а	всего
I. Интерфейс программы Scratch (1 ч)				
1	Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.	1	0	1
II. Начало работы в среде Scratch (2 ч)				
2	Сцена. Редактирование фона. Добавление фона из файла.	0,5	0,5	1
3	Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.	0,5	0,5	1

III. Основные скрипты программы Scratch (13ч)				
4	Синий ящик – команды движения. Темно-зеленый ящик – команды рисования.	0,5	0,5	1
5	Фиолетовый ящик – внешний вид объекта. Оживление объекта с помощью добавления костюмов.	0,5	0,5	1
6	Желтый ящик – контроль. Лиловый ящик – добавление звуков.	0,5	1,5	2
7	Использование в программах условных операторов.	0,5	0,5	1
8	Функциональность работы циклов. Цикличность выполнения действий в зависимости от поставленных условий.	0,5	1,5	2
9	Зеленый ящик – операторы. Использование арифметических и логических блоков вместе с блоками управления.	0,5	0,5	1
10	События. Оранжевый ящик – переменные.	0,5	0,5	1
11	Списки.	0,5	1,5	2
12	Голубой ящик – сенсоры. Ввод-вывод данных.	0,5	1,5	2
IV. Работа с несколькими объектами. Синхронизация их работы (3 ч)				
13	Последовательность и параллельность выполнения скриптов.	0,5	0,5	1
14	Взаимодействие между спрайтами. Управление через обмен сообщениями.	1	1	2
V. Использование программы Scratch для создания мини-игр (10 ч)				
15	Виды компьютерных игр. Алгоритмическая разработка листинга программы.	1	1	2
16	Разработка базовых спрайтов для игры. Формирование базовых скриптов.	0,5	0,5	1
17	Синхронизация работы скриптов для разных спрайтов.	1	1	2
18	Переход из одной сцены в другую. Создание интерфейса игры.	1	1	2
19	Сообщество Scratch в Интернете. Просмотр и публикация проектов.	0,5	0,5	1
20	Разработка и защита творческого проекта	0	2	2
Итого:		12	17	29

КАЛЕНДАРНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№ п/п	СОДЕРЖАНИЕ	Дата проведения занятий	Коррекция
Темы 1. «Интерфейс программы Scratch» (1 час).			
1.	Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.		
Тема 2. «Начало работы в среде Scratch» (2 часа).			
2.	Сцена. Редактирование фона. Добавление фона из файла.		
3.	Понятие спрайтов. Добавление новых спрайтов. Рисование новых объектов.		
Тема 3. «Основные скрипты программы Scratch» (13 часов).			
4.	Синий ящик – команды движения.		
5.	Темно-зеленый ящик – команды рисования.		
6.	Фиолетовый ящик – внешний вид объекта.		
7.	Оживление объекта с помощью добавления костюмов.		
8.	Желтый ящик – контроль.		
9.	Лиловый ящик – добавление звуков.		
10.	Использование в программах условных операторов.		
11.	Функциональность работы циклов.		
12.	Цикличность выполнения действий в зависимости от поставленных условий.		
13.	Зеленый ящик – операторы.		
14.	Использование арифметических и логических блоков вместе с блоками управления.		
15.	События. Оранжевый ящик – переменные.		
16.	Списки.		
Тема 4. «Работа с несколькими объектами. Синхронизация их работы» (3 часа).			
17.	Голубой ящик – сенсоры. Ввод-вывод данных.		
18.	Последовательность и параллельность выполнения скриптов.		
19.	Взаимодействие между спрайтами.		
Тема 5. «Использование программы Scratch для создания мини-игр» (10 часов).			
20.	Взаимодействие между спрайтами. Управление через обмен сообщениями.		
21.	Виды компьютерных игр. Алгоритмическая разработка листинга программы.		

22.	Разработка базовых спрайтов для игры. Формирование базовых скриптов.		
23.	Синхронизация работы скриптов для разных спрайтов.		
24.	Синхронизация работы скриптов для разных спрайтов.		
25.	Переход из одной сцены в другую. Создание интерфейса игры.		
26.	Переход из одной сцены в другую. Создание интерфейса игры.		
27.	Сообщество Scratch в Интернете. Просмотр и публикация проектов.		
28.	Разработка и защита творческого проекта		
29.	Разработка и защита творческого проекта		

Методические материалы

Основные требования к оформлению проекта по информатике. Формы продуктов проектной деятельности - Мультимедийный продукт

Структура проекта:

- Титульный лист.
- Оглавление.
- Введение. Формулируется суть проблемы, обосновывается выбор темы, определяется ее значимость и актуальность, указываются цель и задача проекта
 - Технологическая разработка проекта.
 - Разработка макета проекта;
 - Разработка структуры проекта;
 - Разработка типовой страницы (слайда) проекта;
 - Раскрыть теоретический блок по курсу информатики.
 - Заключение. Подвести общие итоги, сформулировать основные выводы, проанализировать степень успешности решения поставленных автором проекта целей и задач, определить перспективы дальнейшей разработки данной темы.
 - Библиографический список. Оформляется в соответствии с требованиями к письменной реферативной работе. Источниками данных для работы над проектом могут быть:
 - ресурсы Интернет, с помощью которого ученики также учатся правильно формировать запросы по поиску данных, обрабатывать полученную информацию;

- вспомогательная литература (статьи из журналов, газет, научно-техническая литература);
- лекционный и практический материал занятий.

Структура титульного листа

Название образовательной организации

Проект по предмету

Тема « »

Выполнил:

Учащийся ФИО

Руководитель:

Учитель ФИО

Местонахождение образовательной организации

Год написания

ПЛАН ЗАЩИТЫ ТВОРЧЕСКОГО ПРОЕКТА

1. Сообщение темы творческого проекта.
2. Демонстрация изготовленных изделий.
3. Цели и задачи творческого проекта.
4. Описание работы над изделием:
 - используемое ПО;
 - требования к параметрам ПК;
 - последовательность технологии создания.
5. Какие проблемы были при разработке проекта.
6. Что нового узнали при работе над выбранной темой.
7. Область применения.
8. Выводы.

Критерии оценки проекта

Критерии пояснительной записки оценки проекта :

- содержательная часть: актуальность темы, глубина и полнота раскрытия темы, адекватность передачи первоисточника;
 - целостность: соответствие проекта теме, логичность, связность, доказательность;
 - структурная упорядоченность пояснительной записки (наличие введения, основной части и заключения, их оптимальное соотношение), смысловая завершенность;

- оформление (соответствие требованиям, наличие плана, языковая правильность, наличие и эстетика иллюстративного материала, списка литературы, культура цитирования, сноски и т. д.);
- представление на процедуре защиты (как держится докладчик, насколько свободно ориентируется в тексте реферата, как реагирует на вопросы и т.п)

Критерии к практической части проекта.

- наличие обоснования выбора темы, ее актуальности;
- наличие сформулированных целей и задач работы;
- наличие краткой характеристики источников информации.
- реализация поставленных целей и задач
- наличие выводов по результатам анализа;
- выражение своего мнения по проблеме.