Управление по образованию Минского райисполкома

Государственное учреждение дополнительного образования

«Центр творчества детей и молодёжи Минского района»

УТВЕРЖДАЮ

Директор ГУДО «Центр творчества детей и молодёжи Минского района»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ О.А.Тимохина «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019

ПРОГРАММА

ОБЪЕДИНЕНИЯ ПО ИНТЕРЕСАМ

«Scratch программирование»

(базовый уровень изучения образовательной области «Информатика»)

Автор-составитель: ГАНЬКО

Николай Леонидович

Возраст учащихся: 10-13 лет

Срок реализации программы: 1 год

**По программе работают:**

КУРГАН Виктория Александровна

|  |
| --- |
| Рекомендовано  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2019г  Протокол №\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Сеница 2019

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

Индустрия создания компьютерных игр требует от разработчиков широкого спектра компетенций. Обучение школьников созданию компьютерных игр должно способствовать повышению интереса и мотивации учащихся к изучению школьных предметов информатики, математики, физики, литературы, истории и многих других. Первичным при этом является очевидное увлечение детей компьютерными играми. В процессе изучения технологии и инструментов разработки компьютерных игр у учащихся формируется восприятие компьютера и др. электронных устройств не только как источника развлечений, но и как рабочего инструмента, востребованного во многих современных профессиях. Кроме того, благодаря увлекательной форме использования знаний различных дисциплин более эффективно происходит осознание межпредметных связей.

Scratch (Скретч) — это среда визуального программирования с графическим интерфейсом, которая была создана медиалабораторией Массачусетского технологического института (the Media Laboratory at the Massachusetts Institute of Technology — MIT), чтобы сделать программирование простым, понятным и интересным именно для детей. Продукт и среда открыты, бесплатны и доступны на сайте scratch.mit.edu. Как утверждают разработчики, Scratch помогает детям учиться думать творчески и критически, работать вместе — это базовые навыки для жизни в ХХI в.

Scratch - не только среда для обучения программированию, в первую очередь Scratch - это инструмент для развития у учащихся таких навыков XXI века, как:

**информационная грамотность**. Создавая проекты, дети работают с разными видами информации: текст, графика, анимация, звук;

**коммуникативные навыки**. Scratch позволяет учащимся работать над проектами совместно;

**критическое и системное мышление**. Работая в Scratch, дети учатся критически мыслить и рассуждать: в проектах надо согласовывать поведение героев, их взаимодействие;

**креативность и любознательность**.

**ЦЕЛЬ ПРОГРАММЫ:** развитие логики и алгоритмического мышления, творческих способностей, формирование базовых компетенций в области разработки компьютерных игр.

**ЗАДАЧИ ПРОГРАММЫ**:

* развитие у учащихся логики, алгоритмического, образного и аналитического мышления, творческих способностей;
* формирование знаний и умений по созданию анимации, компьютерных игр в среде визуального программирования Scratch;
* формирование навыков сетевой коммуникации;
* формирование навыков алгоритмизации и программирования на языке Scratch;
* формирование навыков работы в команде;
* формирование навыков систематизации информации, самообучения и самоконтроля;
* отработка умений и навыков презентации проектов.

Основными **формами** и **методами** проведения учебных занятий являются: практические работы на компьютере, метод проектов, демонстрации с использованием видеоматериалов и наглядных средств (*по принципу «делай, как я»: учитель показывает, как использовать те или иные инструменты, а учащиеся повторяют за ним*), проблемные беседы, дискуссии, защита индивидуальных и коллективных проектов. В рамках курса возможны встречи с профессиональными разработчиками компьютерных игр, а также другими ИТ-специалистами (по инициативе учителя и при поддержке ИТ-компаний и компаний-разработчиков компьютерных игр).

Доступ к упоминаемым в учебной программе концепциям игр, графическим библиотекам и соответствующим методическим материалам, которые предоставляют геймдизайнеры, обеспечивается посредством сети Интернет на сайте <http://www.scratch.by>. Кроме того, учитель может использовать любые другие примеры игр и собственные идеи проектов (в частности, образовательных), которые могут быть реализованы в среде Scratch.

Контроль достижений учащихся осуществляется через наблюдение за активностью учащихся на учебных занятиях, беседы с учащимися при выполнении практических работ и защите проектной работы в данном объединении по интересам, а также участие в республиканских и международных конкурсах по программированию и разработке игр в среде Scratch.

ОРГАНИЗАЦИОННЫЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

Программа рассчитана на учащихся – *10-13 лет*

Общее количество часов в год – *72 часа*

Периодичность проведения занятий – *1 раз в неделю*

Продолжительность одного занятия – *2 академических*  *часа*

Занятия проводятся в компьютерном классе.

**Рекомендации по оборудованию и программному обеспечению.**

Желательно обеспечить каждого учащегося персональным компьютером с установленными:

* операционной системой Windows, Linux или Mac OS;
* Adobe AIR и офлайн-редактором Scratch (Scratch Offline Editor) –подробнее см. <https://scratch.mit.edu/scratch2download/>.

Требуется подключение к сети Интернет (минимально - компьютер учителя) и желательно наличие проектора.

*Примечание: если в компьютерном классе не все компьютеры подключены к сети Интернет, то работа может быть частично организована на базе компьютера учителя с использованием проектора либо в режиме онлайн, либо с использованием офлайн-редактора Scratch и подготовленных учителем скриншотов и рекомендованных в программе примеров игр, загруженных заранее из Сети, или аналогичных проектов, подготовленных учителем.*

***Учебно-тематический план***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п\п | Наименование темы | Количество часов | | |
| всего | теория | практика |
|  | Вводное занятие | 2 | 2 | - |
| 1 | Анимация как вступительная и обучающая составляющая игры | 4 | 2 | 2 |
| 2 | Составляющие компьютерной игры | 6 | 2 | 4 |
| 3 | Сообщество Scratch. Создание и редактирование персонажей. Покадровая анимация | 6 | 2 | 4 |
| 4 | Переменные и данные. Типы данных. Операторы | 4 | 2 | 2 |
| 5 | Сцена и локации игры. Многоуровневые игры.  Фон как составляющая интерфейса. Введение в игровую физику. Анимация локации. Перспектива и 3D | 6 | 2 | 4 |
| 6 | Координаты и координатная плоскость. Определение алгоритма. Алгоритмические конструкции: ветвления и циклы.  Логические (булевы) выражения | 6 | 2 | 4 |
| 7 | Диалоги и интерфейс. Команды в Scratch на английском языке и их аналоги-операторы языков программирования. Образовательные проекты и игры. Коллективная разработка игр. Авторское право | 6 | 2 | 4 |
| 8 | Защищенные переменные и считывающие блоки scratch. Сложные вычисления и формулы. Использование случайных чисел в компьютерных играх. Строковые типы данных | 6 | 2 | 4 |
| 9 | Инструмент «перо». Углы, градусы, вращения и повороты. Модель и моделирование. Виртуальная реальность и игровая физика | 6 | 2 | 4 |
| 10 | Систематизация и классификация. Виды игр. Игровые жанры. Польза и вред компьютерных игр. Геймификация. Элементы игры. Команда разработчиков игры. Списки и массивы | 6 | 2 | 4 |
| 11 | Подпрограммы и функции в программировании | 6 | 2 | 4 |
| 12 | Работа со звуком и музыкой в Scratch.  Искусственный интеллект | 6 | 2 | 4 |
|  | Итоговое занятие | 2 | - | 2 |
| **Всего** | | **72** | **26** | **46** |

СОДЕРЖАНИЕ ОБРАЗОВАТЕЛЬНОЙ ОБЛАСТИ

**Тема 1. Анимация как вступительная и обучающая составляющая игры**

**Правила работы и безопасного поведения в компьютерном классе.**

**Анимация — основа любой игры. Алгоритмы, программы и скрипты.** **Анимация. Сценарий.** Анализ различных определений термина анимация. Анализ примеров анимации.

Определение понятий алгоритма, программы и скрипта. Сходство и отличия указанных терминов. Способы описания сценария анимации.

Задача: описать сценарий и создать анимацию на любую тему. Создание анимации для вступительной или обучающей составляющей игры. Примеры могут быть привязаны к одной или нескольким играм, предлагаемым геймдизайнерами (экшн-платформеры типа Mario Bros, Doodle Jump, экшн-топдауны: Танчики, или Гонки, и другие, в которых персонажи должны перемещаться по сцене). Например, берется заготовка сцены и на ней размещается спрайт персонажа. Реализация различных способов анимации движения персонажа.

**Тема 2. Составляющие компьютерной игры**

Интерактивность. Компьютерная игра. Составляющие компьютерной игры. Понятие компьютерной программы. Персонажи и спрайты. События и сенсоры. Взаимодействие персонажей игры.

Задача: на основе рассмотренного на предыдущем занятии сценария анимации и ее реализации (это сквозной проект для работы в течение нескольких занятий) добавляются элементы интерактивности в виде управления движением персонажем с помощью мыши, а затем – модификации с управлением с помощью клавиатуры. Добавление обработки касания спрайтов («звёзды», «бонусы», «ключи» и т.п.) и «собирание» их на сцене.

Коллективное обсуждение изученных понятий с рассмотрением примеров. Отличие анимации от игры. Анализ реализованных элементов интерактивности, описание того, что еще хотелось бы реализовать или перечисление и описание элементов интерактивности для проектируемой новой игры.

**Тема 3. Сообщество Scratch. Создание и редактирование персонажей. Покадровая анимация**

Изучение ресурсов сайта scratch.mit.edu. Изучение ресурсов студии «Юный разработчик игр (Беларусь)»: <https://scratch.mit.edu/studios/1463078/> и других студий. Примеры анимации и игр. Разделы разработок зарегистрированных участников портала Scratch. Различные виды проектов Scratch. Выбор на сайте проекта, который хотелось бы изучить и реализовать самостоятельно. Описание сценария этого проекта.

Альтернативная задача: Изучение примеров игр и концепций, предложенных геймдизайнерами. Определение проекта, который хотелось бы реализовать самостоятельно. Описание сценария этого проекта.

**Персонажи и спрайты.** Изучение библиотек спрайтов, предложенных геймдизайнерами. Импорт-экспорт спрайтов.

**Анимация персонажей.** Покадровая анимация. Программная анимация. Пошаговое выполнение скрипта.

Создание спрайтов и костюмов. Добавление в «сквозной» проект анимации на основе смены костюмов.

**Тема 4. Переменные и данные. Типы данных. Операторы**

**Переменные в программировании.** Типы данных. Ввод-вывод данных. Способы отображения значения переменной на сцене (двойной клик по блоку вывода).

Подсчет очков в игре. Добавление подсчета очков в «сквозной» проект.

Реализации сложных вычислений с использованием большего количества команд группы ОПЕРАТОРЫ, в том числе «Остаток от деления () на ()»; «Округлить ()»; «(квадратный корень …) от ()».

Разработка сценария, создание переменных и спрайтов для игровых или образовательных проектов с вычислениями и подсчетами. Например, игры типа «Клик-менеджер»: «Наряжалка», «Хидден». Задачи по математике: решение уравнений и др. вычисления. Проект «Калькулятор».

**Тема 5. Сцена и локации игры. Многоуровневые игры. Фон как составляющая интерфейса. Введение в игровую физику. Анимация локации. Перспектива и 3D**

**Сцена и фоны сцены.** **Фон как составляющая интерфейса проекта.**

Анализ способов реализации смены уровней игры и использования специальных фонов в интерфейсе проекта. Например, «О проекте», «Правила игры», начало (заставка) и конец игры и пр.

Добавление смены фонов-уровней и элементов интерфейса в любой начатый ранее сквозной проект, например: экшн-платформеры типа Mario Bros, Doodle Jump, экшн-топдауны: Танчики, или Гонки, или другие подобные, а также смена фонов (уровней) в играх типа «Клик-менеджер»: «Наряжалка», «Хидден» и т.п.

Понятие перспективы и способы ее реализации в играх Scratch (анимация фона, вспомогательных «фоновых» спрайтов, например, деревьев, домов и т.п.). Имитация реализации 3D-игр в Scratch.

Игровая физика и способы реализации движения персонажа. Стартовый проект Scratch «Pong Starter remix»: <https://scratch.mit.edu/projects/89474936/>. Управление скоростью движения спрайта, понятие ускорения. Способы реализации прыжков и отскоков спрайта.

Описание сценария более сложного «поведения» персонажа игры для добавления в сквозной проект необходимых элементов более логичного управления движением персонажа (например, автоматическое изменение местоположения спрайта по достижении им края сцены в играх типа экшн-платформеры Mario Bros, Doodle Jump, экшн-топдауны: Танчики, или Гонки, или других подобных).

Подготовка фонов сцены с учетом описанных в сценарии возможностей. Например, добавление «триггеров», «дверей» и пр. для обработки касания и автоматизации перемещения в необходимые координаты или на новый уровень (сменой фона).

Описание сценариев и реализация появления (в том числе с перемещением по сцене) кликабельных спрайтов (одежда, найденные на сцене объекты и пр.).

**Тема 6. Координаты и координатная плоскость. Определение алгоритма. Алгоритмические конструкции: ветвления и циклы.**

**Логические (булевы) выражения**

**Координаты и координатная плоскость.**

Сравнение команд движения «Идти () шагов» и команд работы с координатами. Различные способы управления движением персонажа с использованием этих команд. Сравнение двух способов управления движением спрайта с помощью клавиатуры:   
1) с использованием команд СОБЫТИЯ: «Когда клавиша () нажата»;   
2) с использованием команд УПРАВЛЕНИЕ: «Если <> То …» и СЕНСОРЫ: «Клавиша () нажата?». Реализация наиболее подходящих для «сквозного» проекта способов перемещения персонажей.

Реализация появления кликабельных спрайтов (одежда, найденные на сцене объекты и пр.) в играх типа «Клик-менеджер»: «Наряжалка», «Хидден» и т.п. в необходимых координатах на сцене, в том числе с перемещением по сцене.

**Определение алгоритма.** **Логические (булевы) выражения**.

Добавление «умного» управления в игры типа «Клик-менеджер» («Наряжалка», «Хидден») и др.: условия типа «не надевать одну одежду на другую» и т.п. проверка различных условий.

**Циклы и их виды в Scratch.** Использование циклов различного вида для оптимизации графических и пр. эффектов создаваемого проекта.

Понятие версионности в создании игр и компьютерных программ.

Пометки в виде комментариев о возможных вариантах развития и модификации игры.

Примеры проектов сайта scratch.mit.edu с использованием координат и циклов. Например, «Fireworks 100% Pen»: https://scratch.mit.edu/projects/100903791.

**Тема 7. Диалоги и интерфейс. Команды в Scratch на английском языке и их аналоги-операторы языков программирования. Образовательные проекты и игры.** **Коллективная разработка игр. Авторское право**

**Понятие интерфейса.** Закрепление определений понятия интерфейса.

Использование спрайтов и фонов для реализации элементов интерфейса. Проектирование и реализация интерфейса игрового или образовательного проекта с использованием спрайтов-кнопок (со всплывающими подсказками) и добавление визуальных и пр. эффектов наведения мыши и нажатия на кнопки.

Закрепление определения термина диалог. Диалоги персонажей. Проектирование и реализация диалогов персонажей любого начатого ранее или нового проекта.

**Образовательные проекты и игры. Коллективная разработка игр.** Проектирование и разработка новой образовательной игры в командах по 2 – 3 человека

**Авторское право.** Ремиксы игр на сайте Scratch и их обсуждение. Документирование создаваемых проектов с помощью комментариев в коде.

Т**ема 8. Защищенные переменные и считывающие блоки** **scratch. Сложные вычисления и формулы. Использование случайных чисел в компьютерных играх. Строковые типы данных**

**Защищенные переменные, считывающие блоки и случайные числа.** Разнообразное использование команд группы «ОПЕРАТОРЫ» с целью изучения команд для математических расчетов и вычислений с использованием функций. Разработка образовательных проектов по математике.

Dynamic Maurer Rose remix»: <https://scratch.mit.edu/projects/104738367/> - первичное ознакомление, для изучения использованных в проекте формул, с отсылкой на тему 9 – Инструмент «Перо».

Игры жанра «Пазл»: случайное перемешивание объектов. Игра типа «Охотник»: появления зайцев в случайных местах и т.п.

**Строковые типы данных.** Разработка языковых игр и др. проектов с использованием строкового типа.

**Тема 9. Инструмент «перо». Углы, градусы, вращения и повороты. Модель и моделирование. Виртуальная реальность и игровая физика**

**Инструмент «Перо» и рисование. Направление движения и повороты.** Ознакомление с образовательным проектом «Геометрические фигуры»: <https://scratch.mit.edu/projects/90085838/> и реализация аналогичных проектов. Разработка графического редактора с функциями построения геометрических фигур и других образовательных игр. Проекты типа «Разукрашка», «Рисовалка» и т.п.

**Графики функций и траектории.** Использование математических формул и рисование графиков функций. Рисование графиков функций с помощью инструмента «Перо». Способы задания траекторий движения персонажей. Проектирование и реализация движения персонажей по заданной траектории. Добавление эффектов «вылета» на сцену различных объектов по траектории.

**Модель и моделирование. Виртуальная реальность и игровая физика.** Реализация реалистичных физических «способностей» персонажей «сквозного» проекта. Динамическое управление скоростью движения, силой броска и т.п.

**Тема 10. Систематизация и классификация. Виды игр. Игровые жанры. Польза и вред компьютерных игр. Геймификация. Элементы игры. Команда разработчиков игры. Списки и массивы**

**Систематизация и классификация. Виды игр.**

Коллективное обсуждение реализуемых проектов: какие игровые жанры избраны разработчиками? Геймификация игровых и пр. проектов.

Команда разработчиков игры. Некоторые ИТ-профессии. Коллективное обсуждение команд (реальных или придуманных) и определение «профессий» каждого участника.

Возможна встреча с профессиональными разработчиками компьютерных программ и игр или экскурсия в ИТ-компанию.

**Списки и массивы. Изучение примеров игр со списками.** Оптимизация языковых игр и интерфейсов других проектов с помощью списков.

**Тема 11. Подпрограммы и функции в программировании**

**Подпрограммы и функции.** Изучение примеров проектов с использованием группы команд ДРУГИЕ БЛОКИ: «Другие блоки (магазин-отделы copy)»: <https://scratch.mit.edu/projects/102905443/>; образовательный проект «Геометрические фигуры»: <https://scratch.mit.edu/projects/90085838/>; стартовый проект Scratch «Рисование спиралей (remix)»: <https://scratch.mit.edu/projects/89475559/>; «Dynamic Maurer Rose remix»: <https://scratch.mit.edu/projects/104738367/>.

Понятие рекурсии на примерах: «Фрактал - кривая Коха»: <https://scratch.mit.edu/projects/107536230/>; «Fraktal remix»: <https://scratch.mit.edu/projects/106367149/>; «Кривая дракона remix»: https://scratch.mit.edu/projects/106367127/ и др.

Создание аналогичных проектов. Оптимизация ранее реализованных проектов путем использования других блоков («подпрограмм») Scratch.

**Технологический цикл создания компьютерной игры.** Творческий проект «Создание собственной компьютерной анимации, игры или образовательного проекта». Определение цели и постановка задач проекта. Описание сценария и алгоритмов анимации, игры или образовательного проекта. Реализация проекта. Тестирование и документирование.

**Тема 12. Работа со звуком и музыкой в Scratch.**

**Искусственный интеллект**

**Работа со звуком и музыкой.** Добавление звука в ранее созданные проекты. Реализация более сложных игр со звуком.

**Музыкальные игры.** Разработка любых музыкальных игр. Например, музыкальный синтезатор, обучение музыке и пр.

**Искусственный интеллект (ИИ):** создание проектов с элементами ИИ по примерам: «Синтезатор речи (Text-to-Speech Rus) remix»: <https://scratch.mit.edu/projects/74562822/>, «Робот поэт онлайн remix»: <https://scratch.mit.edu/projects/74563000/> или других.

**Обобщение и завершение курса.** Презентация проекта «Создание собственной компьютерной анимации или игры».

**ОЖИДАЕМЫЙ РЕЗУЛЬТАТ**

В результате изучения занятий

**учащиеся будут знать:**

* основы алгоритмизации и программирования;
* инструменты среды визуального программирования Scratch;
* основы создания анимации;
* основы технологии разработки компьютерных игр и образовательных проектов;

**учащиеся будут уметь:**

* алгоритмически и логически мыслить;
* разрабатывать сценарии и алгоритмы анимации, компьютерных игр и образовательных проектов;
* создавать анимацию и простые интерактивные игры, в которых реализуются движения и взаимодействия различных объектов, звук и пр. с помощью инструментов среды визуального программирования Scratch;
* самообучаться.

Знания, умения и навыки, полученные учащимися при изучении данных занятий, будут **способствовать**:

углублению знаний, связанных с содержанием учебных предметов (информатика, математика, физика, литература и др.) и формированию межпредметных связей;

повышению познавательной активности, формированию познавательного интереса, развитию интеллектуального и творческого потенциала;

развитию коммуникативных и информационных компетенций учащихся.

**МЕТОДИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ**

В программе реализуется многоуровневый дифференцированный подход, при котором освоение курса может начинаться с нуля в 6-м или 7-м классе либо базироваться на знаниях и умениях, приобретенных учащимися ранее.

В учебной программе предлагается практико-ориентированный метод обучения по принципу «от практики – к теории», который заключается в том, что инструменты среды программирования Scratch, а также основы алгоритмизации, программирования и технологии разработки игр, на занятиях изучаются на практических примерах реализации отдельных функциональных возможностей игры или программы, без детальных теоретических объяснений. Таким образом, у учащихся формируются практические навыки и доступные для понимания представления о требуемом результате. На последующих занятиях или в течение второго года обучения, в качестве закрепления, изученные инструменты, технологические и пр. моменты определяются и закрепляются теоретически, с необходимыми пояснениями учителя. При этом формируется абстрактное и логическое мышление учащихся.

В зависимости от года обучения, уровня подготовки, интересов и способностей учащихся, учителю предлагается возможность использовать одну или **несколько стратегий проектно-ориентированного обучения**:

а) рассмотрение простых примеров рекомендованных в программе учебно-демонстрационных игр и реализация в течение одного или нескольких занятий одной такой игры или ее модификации (с учетом желания и творческих идей учащихся);

б) использование в качестве сквозных проектов концепций игр основных жанров, которые предлагают компании-разработчики игр (далее – геймдизайнеры) и их поэтапная реализация на протяжении нескольких занятий (в течение почти всего курса) с использованием графических библиотек и пр. материалов, переданных этими компаниями;

в) для тех учащихся, которые наиболее мотивированы и не первый год изучают программирование в среде Scratch – реализация собственных идей и проектов с использованием изучаемых на занятиях возможностей и приемов, которые таким образом повторяются и более глубоко и осознанно закрепляются, усваиваются. У учащихся при этом развивается творческое мышление и способности к самообучению, самоорганизации и самоконтролю;

г) изучение языка программирования и среды Scratch с последующей реализацией образовательных игровых и прикладных проектов по любым предметам школьной программы. Таким образом, у учащихся формируется понимание актуальности сферы информационных технологий и ряда ИТ‑профессий в социально востребованном аспекте, а также мотивация к получению результата, который не просто интересен, но и полезен, в котором заинтересованы многие потребители (как учителя, так и учащиеся более младшего возраста). Кроме того, обеспечивается формирование межпредметных связей в системе учебных дисциплин средней школы. Учителю рекомендуется рассматривать с учащимися учебники любых школьных предметов, консультироваться с учителями-предметниками и формулировать задачи по созданию образовательных проектов и игр, которые в дальнейшем могут быть внедрены в преподавании соответствующих предметов другими учителями.

**Литература и ресурсы сети Интернет**

1. Программирование для детей / К. Вордерман, Дж. Вудкок, Ш. Макаманус [и др.]; пер. с англ. С. Ломакина. — М.: Манн, Иванов и Фербер, 2015. — 224 с.
2. Официальный сайт проекта Scratch. — <https://scratch.mit.edu/>
3. Помощь Scratch. — <https://scratch.mit.edu/help/>
4. Студия «Юный разработчик игр (Беларусь)». — <https://scratch.mit.edu/studios/1463078/>
5. Загрузка офлайн-редактора Scratch. — <https://scratch.mit.edu/scratch2download/>
6. Патаракин Е. Учимся готовить в Scratch. — <http://www.uroki-scratch.narod.ru/DswMedia/patarakin.pdf>
7. Скретч. Материал из Letopisi.Ru «Время вернуться домой». — <http://letopisi.org/index.php/Скретч>
8. SCRATCH (есть справочник команд с примерами). — <http://scratch.ucoz.net/>
9. Творческая мастерская Scratch (описание уроков с примерами). — <http://www.nachalka.com/book/export/html/1398>
10. Школа Scratch. 2010 г. — <http://socobraz.ru/index.php/Школа_Scratch>
11. Программирование в среде Scratch. 2011 г. — <http://scratch-elektiv.ucoz.ru/>
12. [Scratch в Оренбурге](https://sites.google.com/site/orenscratch/) (есть примеры уроков и проектов). — <https://sites.google.com/site/orenscratch/home>
13. Полякова Л. А. (учитель информатики МОУ СОШ № 84 г. Челябинска). Изучаем Scratch (справочник команд, уроки). — <http://www.uroki-scratch.narod.ru/p1aa1.html>
14. Русскоязычный форум на сайте Scratch. — <https://scratch.mit.edu/discuss/27/>
15. Russian Scratch School (российская школа Scratch, куратор — Е. Патаракин). — <https://scratch.mit.edu/studios/73443/>
16. Обучающие скретч-программы для развития детей. Студия «Обучалки». — <https://scratch.mit.edu/studios/81359/>
17. Scratch stories & scripts for russian schools. Студия «Russian Stories». — <https://scratch.mit.edu/studios/7898/>
18. Русское сообщество скретчеров. Студия. — <https://scratch.mit.edu/studios/488294/projects/>
19. Supercode.Ru 2008 Contest Winners. Студия. — https://scratch.mit.edu/studios/17369/

СОГЛАСОВАНО

Начальник управления по образованию

Минского райоисполкома

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В.Остапенко

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2019 г.