Муниципальное Бюджетное Образовательное Учреждение

Полазненская средняя общеобразовательная школа №1

Утверждаю:

Директор МБОУ ПСОШ №1

\_\_\_\_\_\_\_\_\_\_О.М.Брызгалова

« »\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

ТЕХНИЧЕСКОЙ НАПРАВЛЕННОСТИ

«**Программирование** в среде **scratch**»

Составитель: Серина Ксенья Игоревна,

педагог дополнительного образования,

учитель информатики

рп. Полазна, 2022 г.

# ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Содержание курса «Программирование в среде Scratch» для учащихся 3 и 4 классов рассчитано на обучение в объеме 34 учебных часа (1 час в неделю).

Сегодня наука и технология развиваются столь стремительно, что образо- вание зачастую не успевает за ними. Одним из самых известных неформаль- ных способов организации внеучебной образовательной деятельности являет- ся метод проектов. Самым подходящим инструментом для организации такой деятельности является среда Scratch, которая есть серьезное и современное направление компьютерного дизайна и анимации. Овладев даже минималь- ным набором операций, самый неискушённый пользователь может создавать законченные проекты. Scratch - это самая новая среда, которая позволяет де- тям создавать собственные анимированные и интерактивные истории, презен- тации, 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 зна- ниям и дать подходящие средства её реализации. Быть успешным в такой сре- де становится проще.

Scratch можно рассматривать как инструмент для творчества, оставив

программирование на втором плане. Школьники могут сочинять истории, ри- совать и оживлять на экране придуманных ими персонажей, учиться работать с графикой и звуком. Применений возможностям Scratch можно найти множе- ство: 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 школьников.

Когда ученики создают проекты в Scratch, они осваивают множество навыков 21 века, которые будут необходимы для успеха:

* творческое мышление,
* ясное общение,
* системный анализ,
* беглое использование технологий,
* эффективное взаимодействие,
* проектирование,
* умение обучатся и самообучаться,
* самостоятельное принятие решений.

Изучение Scratch может серьезно помочь школьникам освоить азы алго- ритмизации и программирования, а полученные знания пригодятся для даль-

2

нейшего и более серьезного изучения программирования. Работа в среде Scratch ведется так же как средство подготовки учащихся к всевозможным конкурсам и выставкам по данной тематике, которые в настоящее время наби- рают большие обороты.

Изучение данного курса направлено на достижение *спедующих цепей:* формирование общеучебных умений и навыков на основе средств и методов информатики и ИКТ, в том числе овладение умениями рабо- тать с различными видами информации, самостоятельно планировать и осуществлять индивидуальную и коллективную информационную деятельность, представлять и оценивать ее результаты; пропедевтическое изучение понятий основного курса школьной ин- форматики;

воспитание ответственного и избирательного отношения к информа- ции; развитие познавательных, интеллектуальных и творческих спо- собностей учащихся,

развитие способности к решению творческих задач, развитие самостоятельности.

Ключевым в программе является понятие «проектная научно- познавательная деятельность школьника», как совместная (с другими субъек- тами), так и самостоятельная деятельность с использованием методов научно- го исследования, ведущим мотивом которой является познавательный инте- рес. Проектная научно-познавательная деятельность не является самоцелью, но рассматривается как среда, в которой наиболее естественным образом рас- крывается личностный потенциал школьника.

В этой связи ***цепями проектной научно-познавательной деятепьно- cmu шкопьника*** являются:

развитие интеллектуальных, познавательных и творческих способно- стей школьника;

развитие метапредметных умений (личностных, познавательных, коммуникативных, регулятивных);

развитие способов мыслительной деятельности;

формирование целостной картины мира и системного мышления на основе межпредметных связей.

Следует иметь в виду, что возрастные особенности младшего школьника не позволяют в полной мере реализовать проведение полноценных научных исследований. В то же время раннее включение в организованную специаль- ным образом проектную деятельность творческого характера позволяет сфор- мировать у школьника познавательный интерес и исследовательские навыки, которые в старшем возрасте пригодятся им для выполнения научно- познавательных проектов.

***Основная задача курса*** способствовать формированию у школьников информационной и функционалъной компетентности, развитие алгоритмиче- ского мышления. Назначение курса — помочь детям узнать основные возмож- ности компьютера и научиться им пользоваться в повседневной жизни. А так же:



дать учащимся представление о современном подходе к изучению ре- ального мира, о широком использовании алгоритмов и вычислитель- ной техники в научных исследованиях;

сформировать у учащихся умения владеть компьютером как сред- ством решения практических задач;

подготовка учеников к активной полноценной жизни и работе в усло- виях технологически развитого общества;

создание условий для внедрения новых информационных технологий в учебно-воспитательный процесс школы;

реализовать в наиболее полной мере возрастающий интерес учащихся к углубленному изучению программирования через совершенствова- ние их алгоритмического и логического мышления;

формирование знаний о роли информационных процессов в живой природе, технике, обществе;

формирование знаний о значении информатики и вычислительной техники в развитии общества и в изменении характера труда челове- ка;

формирование знаний об основных принципах работы компьютера, способах передачи информации;

формирование знаний об основных этапах информационной техноло- гии решения задач в широком смысле;

формирование умений моделирования и применения его в разных предметных областях;

формирование умений и навыков самостоятельного использования компьютера в качестве средства для решения практических задач.

Реализация этих задач будет способствовать дальнейшему формирова- нию взгляда школьников на мир, раскрытию роли информатики в формирова- нии естественнонаучной картины мира, развитию мышления, в том числе формированию алгоритмического стиля мышления, подготовке учеников к жизни в информационном обществе.

# Требования к подготовке учащихся Учащиеся должны:

***знать:***

отдельные способы планирования деятельности;

составление плана предстоящего проекта в виде рисунка, схемы; составление плана предстоящего проекта в виде таблицы объектов, их свойств и взаимодействий;

разбиение задачи на подзадачи;

o требованиях к организации компьютерного рабочего места, соблю- дать требования безопасности и гигиены в работе со средствами ИКТ; устройства компьютера;

примеры информационных носителей.

*уметь:*

4

приводить простые жизненные примеры передачи, хранения и обра- ботки информации в деятельности человека, в живой природе, обще- стве, технике;

различать программное и аппаратное обеспечение компьютера; запускать программы из меню Пуск;

изменять размеры и перемещать окна, реагировать на диалоговые ок- на;

вводить информацию в компьютер с помощью клавиатуры и мыши; применять текстовый редактор для набора, редактирования и форма- тирования простейших текстов;

иметь представление о способах кодирования информации;

применять простейший графический редактор для создания и редак- тирования рисунков;

выполнять вычисления с помощью приложения Калькулятор; составлять план проекта, включая:

овыбор темы;

оанализ предметной области; оразбиение задачи на подзадачи;

проанализировать результат и сделать выводы; найти и исправить ошибки;

подготовить небольшой отчет о работе; публично выступить с докладом;

наметить дальнейшие пути развития проекта.

# Планируемые результаты

***Личностные*** *резупьтаты*

К личностным результатам освоения информационных и коммуникационных технологий как инструмента в учёбе и повседневной жизни можно отнести:

* критическое отношение к информации и избирательность её восприя- тия;
* уважение к информации о частной жизни и информационным резуль- татам других людей;
* осмысление мотивов своих действий при выполнении заданий с жиз- ненными ситуациями;
* начало профессионального самоопределения, ознакомление с миром профессий, связанных с информационными и коммуникационными технологиями.

***Метапредметными резупьтатами*** изучения курса «Программирование в среде Scratch» являются формирование следующих универсальных учебных действий:

*Регцпятивные* ***УУД:***

* планирование последовательности шагов алгоритма для достижения цели;
* поиск ошибок в плане действий и внесение в него изменений.

## Познаватепьные УУД:

* моделирование преобразование объекта из чувственной формы в модель, где выделены существенные характеристики объекта (про- странственно-графическая или знаково-символическая);
* анализ объектов с целью выделения признаков (существенных, несу- щественных);
* синтез составление целого из частей, в том числе самостоятельное достраивание с восполнением недостающих компонентов;
* выбор оснований и критериев для сравнения, сериации, классифика- ции объектов;
* подведение под понятие;
* установление причинно-следственных связей;
* построение логической цепи рассуждений.

## Коммцникативные УУД:

* аргументирование своей точки зрения на выбор оснований и критери- ев при выделении признаков, сравнении и классификации объектов;
* выслушивание собеседника и ведение диалога;
* признавание возможности существования различных точек зрения и права каждого иметь свою.

***Предметными резупьтатами*** изучения курса «Программирование в среде Scratch» в 5-м классе являются формирование следующих **знаний и умений:**

|  |  |
| --- | --- |
| **УЧАЩИЕСЯ** | |
| должны зітхть | **ДОЛЖНЫ УМЕТЬ** |
| Что такое Scratch и его назначение.  Основные базовые алгоритмиче- ские конструкции. Исполнитель и его система команд. Самодостаточ- ные и открытые скрипты. Принцип создания анимации и движения объектов. | Размещать объекты на сцене. Поворачи- вать их и масштабировать. |
| Сцена. Текущие координаты объек- та. Спрайт. Интерфейс программы Scratch. | Вставлять стандартный фон из библио-  течного модуля среды. Рисовать фон в графическом редакторе. Добавлять фон из файла. Создавать спрайты с помощью графического редактора среды Scratch. Загружать на сцену спрайты из стан- дартной коллекции Scratch. Вставлять спрайты из файлов. Центрировать ко- стюм. Масштабировать спрайт. Удалять спрайты. |
| Команды из ящиков движения, | Создавать программы для движения |
| внешности, звука, рисования, кон- | спрайтов по сцене, для рисования раз- |
| троля, сенсоров, операторов и пе- | личных фигур, имитации естественного |

б

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ременных.  Scratch | События | в | проектах | движения героев в различных направле-  ниях. Озвучивать как полностью проект, так и отдельные события внутри проек- та. Создавать программы - с изменением последовательного выполнения скрип- тов при наличии условий, с использова- нием циклов с фиксированным числом повторений, с предусловием и посту- словием. Использовать в программах операции сравнения данных, арифмети- ческие и логические действия над дан- ными, сравнение данных из нескольких списков, глобальные и локальные пере- менные. Обрабатывать данные с выво- дом на экран конечного результата |
| Принцип взаимодействия спрайтов через обмен сообщениями. После- довательные и параллельные пото- ки в программах Scratch | | | | Создавать Scratch-истории с взаимодей-  ствием нескольких исполнителей и не- подвижных объектов, а так же с одно- временной и попеременной работой не- скольких исполнителей. |
| Виды компьютерных игр. Этапы создания компьютерных игр. Ин- терфейс игры. Адрес сообщества Scratch в Интернете. Авторские права. | | | | Поэтапно создавать компьютерную иг-  ру. Создавать программу для перемеще- ния объекта по игровой карте в одном направлении и в пространстве из не- скольких связанных между собой ком- нат. Разрабатывать интерфейс для Scratch проекта. Регистрироваться на сайте сообщества Scratch. Просматри- вать проекты сообщества и публиковать  собственные проекты. |

Тематическое планирование

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | НАИМЕНОВАНИЕ ТЕМЫ | TEOPH | I I PAKT. | BCEГO |
| І. **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 фай- ла. | 0,5 | 0,5 | 1 |
|  | Понятие спрайтов. Добавление новых спрайтов. Рисова- ние новых объектов. | 0,5 | 0,5 |  |
| **III. Основные скрипты программы Scratch (18 ч)** | | | | |
| 4 | Синий ящик команды движения. Темно-зеленый ящик команды рисования. | 0,5 | 1,5 | 2 |
|  | Фиолетовый ящик внешний вид объекта. Оживление  объекта с помощью добавления костюмов. | 0, | 3,5 | 2 |
| 6 | Желтый ящик — контроль. Лиловый ящик добавление  зВукоВ. | 0, | 3,5 | 2 |
| 7 | Использование в программах условных операторов. | 0,5 | 1,5 | 2 |
| 8 | Функциональность работы циклов. Цикличность выпол-  нения действий в зависимости от поставленных условий. | 0,5 | 1,5 | 2 |
| 9 | Зеленый ящик операторы. Использование арифметиче-  ских и логических блоков вместе с блоками управления. | 0,5 | 1,5 | 2 |
| 10 | События. Оранжевыйящик переменные. | 0,5 | 1,5 | 2 |
| 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 (4 ч)** | | | | |
| 13 | Последовательность и параллельность выполнения  СК]ЭИПТОВ. | 0,5 | 1,5 | 2 |
| 14 | Взаимодействие между спрайтами. Управление через  обмен сообщениями. |  |  | 2 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| V. **Использование программы Scratch для создания мини-игр (7 ч)** | | | | |
| 15 | Виды компьютерных игр. Алгоритмическая разработка  листинга программы. |  |  | 2 |
| 16 | Разработка базовых спрайтов для игры. Формирование  базовых скриптов. | 0, | 3,5 | 2 |
| 17 | Синхронизация работы скриптов для разных спрайтов. | 0 | 1 | 1 |
| 18 | Переход из одной сцены в другую. Создание интерфейса  ИгрЫ. | 0  ’ | 0 5 | 1 |
| 19 | Сообщество Scratch в Интернете. Просмотр и публика-  ция проектов. | 0,5 | 0,5 |  |
|  | **VI. Разработка творческого проекта (2 ч)** |  |  |  |
| 20 | Разработка и защита творческого проекта | 0 | 6 | 6 |
| **Итого:** | | **11** | **27** | **38** |

Содержание учебного предмета

# І. Интерфейс программы Scratch (1 ч).

1. **Введение. Что такое Scratch. Основные алгоритмические конструкции. Знакомство с интерфейсом программы Scratch.** Теория. 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. Поня- тие исполнителя, алгоритма и программы, их назначение, виды и использова- ние. 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 объек- тов. Листинг программы. Сцена. Текущие данные о спрайте. Стиль поворота. Закладки. Панель инструментов, Новый спрайт. Координаты мышки. Режим представления. Окно скриптов. Окно блоков. Блоки стека. Блоки заголовков. Блоки ссылок. Самодостаточные и открытые скрипты (1 час).

# Начало работы в среде Scratch (2 ч).

1. **Сцена. Редактирование фона. Добавление фона из файла.**

Теория. Сцена. Широта и высота сцены. Текущие координаты объекта. Редак- тирование текущего фона. Вставка нового фона из файла. Вставка стандартно- го фона из библиотечного модуля среды. Рисование фона в графическом ре- дакторе. Создание нескольких фонов в одной сцене (0,5 часа).

**Практика.** Создание фона сцены на выбранную учащимся тему (0,5 часа).

# Понятие спрайтов. Добавление новых спрайтов. Рисование новых объ-

ектов.

Теория. 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. Загруз- ка на сцену спрайтов из стандартной коллекции среды Scratch. Вставка спрай- тов из файлов форматов JPG, BMP, PNG, GIF. Выбор случайного спрайта. Удаление спрайтов (0,5 часа).

**Практика.** Создание фона сцены и прорисовка основных спрайтов для Scratch-истории. (0,5 часа).

1. **Основные скрипты программы Scratch (18 ч).**
2. **Синий ящик** — **команды движения. Темно-зеленый ящик** — **команды рисования.**

Теория. Команды — *идти; повернуться направо (налево); повернуть в направ- лении; повернуться к,‘ изменить х (у) на,‘ установить х (у) в,‘ если край, от- толкнуться.* Принципиальное различие действия команд *идти в* и *плыть в.* Назначение сенсоров *положение х, положение у* и *направлении.* Команды *очистить, опустить nepo, поднять nepo, установить цвет nepa, изменить цвет nepa на, установить цвет nepa, изменить тень nepa, установить тень nepa, изменить размер nepa на, установить размер nepa, печать* (0,5 часа).

**Практика.** Создание программ для передвижения спрайтов по сцене. Созда- ние программ для рисования различных фигур (1,5 часа).

1. **Фиолетовый ящик** — **внешний вид объекта. Оживление объекта с по- мощью добавления костюмов.**

Теория. Костюмы спрайта. Копирование и редактирование костюма спрайта с помощью редактора рисования. Переупорядочивание костюмов. Команды *nepeйти к костюму, следующий костюм, говорить... в течении... секунд, ска- зать, думать, думать... секунд, изменить . .эффект на, установить эф- фект ... в значение, убрать графические эффекты, изменить размер на, уст а- новить размер, показаться, спрятаться, nepeйти в верхний слой, nepeйти назад на... 1 слоев.* Назначение сенсоров *костюм* и *размер.* Понятие раскад- ровки движения. Изменение костюма спрайта для имитации движения (0,5 ча- са).

**Практика.** Создание программы для управления внешним видом объекта. Со- здание Scratch-историй с имитацией хождения и движения объектов (1,5 часа).

1. **Желтый ящик** — **контроль. Лиловый ящик** — **добавление звуков.**

Теория. Кнопка с зеленым флажком и ее назначение. Управление последова- тельностью выполнения скриптов. Понятие управляющих сообщений. Коман- ды — *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. .* КомандьІ — *когда клавиша... нажата, когда щелкнym no, ждать... секунд, ждать до, остано-*

*вить cкpиnт, остановить все.* Загрузка звуков из стандартной коллекции и из файлов жесткого диска. Запись звука через микрофон. Принципиальная раз- ница работы команд *играть звук* и *играть звук до завершения.* Команды *остановить все звуки, барабан у играть... тактов, оставшиеся... тактов, но- ту... играть... тактов, выбрать инструмент, изменить громкость, устано- вить громкость, изменить темп на, установить темп.* Назначение сенсоров *громкость* и *темп* (0,5 часа).

**Практика.** Создание программ с элементами управления объектом. Озвучи- вание Scratch-историй (1,5 часа).

# Использование в программах условных операторов.

Теория. Базовая конструкция ветвление, назначение, виды (полная и неполная форма). Понятие условия. Изменение порядка выполнения скриптов в зависи- мости от условия. Разветвление листинга программы. Скрипты условных опе- раторов. Использование неполной формы ветвления в системе Scratch (0,5 ча- са).

**Практика.** Создание программ с изменением последовательного выполнения скриптов при наличии условий (1,5 часа).

# Функциональность работы циклов. Цикличность выполнения дей- ствий в зависимости от поставленных условий.

Теория. Циклы с фиксированным числом повторений. Заголовок цикла. Тело цикла. Циклы с условным оператором. Заголовок цикла. Тело цикла. Пред- условие и постусловие. Зацикливание (0,5 часа).

**Практика.** Создание программ с использованием циклов с фиксированным числом повторений. Создание программ с использованием циклов с предусло- вием и постусловием (1,5 часа).

# Зеленый ящик — операторы. Использование арифметических и логиче- ских блоков вместе с блоками управления.

Теория. Числа. Стринги. Логические величины. Логические выражения. Арифметические операции. Логические операции. Операции сравнения. Ко- манды для работы со стрингами — *слить, буква... в, длинна строки.* Команда *выдать случайное от...* Эо. Использование арифметических и логических бло- ков в листинге программы. Просмотр полученного результата (0,5 часа).

**Практика.** Создание программ с использованием операций сравнения дан- ных. Создание программ с использованием арифметических данных и логиче- ских операций (1,5 часа).

# События. Оранжевый ящик — переменные.

Теория. События в проектах Scratch. Понятие переменных и необходимость их использования в листинге программы. Глобальные и локальные перемен- ные. Имя переменной и правила его формирования. КомандьІ для переменных

*- поставить... в, изменить... на, показать переменн ую, спрятать переменн ую.*

Удаление переменных. Создание счетчиков с помощью переменных (0,5 часа).

**Практика.** Разработка сценария Scratch-историй с несколькими событиями. Создание проектов с использование глобальных и локальных переменных (1,5 часа).

1. **Списки.**

Теория. Создание списков и необходимость их использования в проектах Scratch. Добавление в список данных. Удаление данных из списка. Удаление списка. Команды работы со списками — *добавить... к, удалить... из, поста- вить... в... из, заменить элемент ... в... на, элемент ... из, длина cnиcкa* (0,5 часа). **Практика.** Создание программ-тестов по принципу сравнения данных из не- скольких списков (1,5 часа).

1. **Голубой ящик** — **сенсоры. Ввод-вывод данных.**

Теория. Понятие сенсора. Правила применения и область действия команд *касается, касается цвета* и *цвет.касается.* Функционал команды *спро- сить... и ждать.* Сенсоры *мышка no х, мышка no у, мышка нажата?, клави- ша... нажата?, расстояние до, nepeзanycтить 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 *cnpocить.* Вывод конечного результата обработки с помощью команд *говорить* и *сказать* (0,5 часа).

**Практика.** Создание проектов с использованием значений сенсоров и коман- ды *cnpocить.* Создание программ для обработки данных пользователя с выво- дом на экран конечного результата (1,5 часа).

1. **Работа с несколькими объектами. Синхронизация их работы (4 ч).**
2. **Последовательность и параллельность выполнения скриптов.**

Теория. Последовательные и параллельные потоки в программах Scratch. Од- новременная и попеременная работа нескольких исполнителей (0,5 часа).

**Практика.** Создание Scratch-историй с одновременной и попеременной рабо- той нескольких исполнителей (1,5 часа).

1. **Взаимодействие между спрайтами. Управление через обмен сообщени-**

ями.

Теория. Решение проблемы появления новых исполнителей только после то- го, как старые исполнители выполнили свои действия. Взаимодействие спрай- тов с неподвижными объектами с помощью команд *касается* и *касается цве- та.* Взаимодействие спрайтов с помощью команд *передать* и *когда я получу.* Использование сообщений для создания событий (1 час).

**Практика.** Создание Scratch-историй с взаимодействием нескольких испол- нителей и неподвижных объектов. Создание Scratch-историй с взаимодействи- ем нескольких исполнителей (1 час).

1. **Использование программы Scratch для создания мини-игр (7 ч).**

# Виды компьютерных игр. Алгоритмическая разработка листинга про-

Теория. Компьютерные игры вред или польза. Виды компьютерных игр. Этапы разработки игр программистами (1 час).

**Практика.** Алгоритмическая разработка проекта, запись на естественном языке событий и точек взаимодействия героев будущей игры (1 час).

# Разработка базовых спрайтов для игры. Формирование базовых скриптов.

Теория. Логика создания персонажей для игры. Перевод алгоритма, написан- ного на естественном языке, в коды Scratch (0,5 часа).

**Практика.** Разработка и создание основных спрайтов и их костюмов для бу- дущей игры. Разработка скриптов для спрайтов и объектов (1,5 часа).

# Синхронизация работы скриптов для разных спрайтов.

**Практика.** Доработка основного листинга программы с целью установления связей между спрайтами. Тестирование и отладка программы (1 час).

# Переход из одной сцены в другую. Создание интерфейса игры.

Теория. Односторонний (без возможности вернуться назад) переход из одного пространства в другое. Понятие интерфейса. Элементы интерфейса. Основные принципы дизайна интерфейсов. Обратная связь. Необходимые элементы ме- ню (0,5 час).

**Практика.** Создать программу для перемещения объекта по игровой карте и разработать интерфейс для Scratch-проекта (0,5 часа).

1. **Сообщество Scratch в Интернете. Просмотр и публикация проектов.** Теория. Правила работы в сети. Интернет-сообщества. Сообщество Scratch. Регистрация на сайте. Использование заимствованных кодов и объектов. Ав- торские права. Публикация проектов Scratch. (0,5 часа).

**Практика.** Регистрация на сайте сообщества Scratch. Просмотр проектов со- общества и публикация собственных проектов (0,5 часа).

# Разработка творческого проекта (6 ч)

1. **Разработка и защита творческого проекта.** Разработка и создание про- граммы с использованием подготовленных материалов. Тестирование и от- ладка проекта. Защита проекта (6 часа).

# Материально-технические условия реализации программы Аппаратное и техническое обеспечение:

Рабочее место обучающегося:

ноутбук: производительность процессора (по тесту PassMark — CPU BenchMark <http://www.cpubenchmark.net/>): не менее 2000 единиц; объём опера- тивной памяти: не менее 4 Гб; объём накопителя SSD/eMMC: не менее 128 Гб (или соответствующий по характеристикам персональный компьютер с мони- тором, клавиатурой и колонками);

мЫшЬ.

Рабочее место наставника:

ноутбук: процессор Intel Core i5-4590/AMD FX 8350 — аналогичная или более новая модель, графический процессор NVФIA GeForce GTX 970, AMD Radeon R9 290 — аналогичная или более новая модель, объём оперативной памяти: не менее 4 Гб, видеовыход HDMI 1.4, DisplayPort 1.2 или более новая модель (или соответствующий по характеристикам персональный компьютер с монитором, клавиатурой и колонками);

презентационное оборудование с возможностью подключения к компьютеру

— 1 комплект;

маркерная доска, соответствующий набор письменных принадлежностей — 1 шт.; единая сеть Wi-Fi.