

Согласованно  
Руководитель «Точки  
роста»  
Протокол № 1  
от « 11 » 08 2020-2021г

*Крыгина Ю.С.*

Утверждено директор  
школы

Приказ № 29/4

от « 11 » 08 2020-2021г

/Крыгина Ю.С./



## Рабочая программа

по информатике

для 5 класса



Класс:5

Учитель: Козина Н.О.

2020-2021 уч. год



## Оглавление

<b>1. Пояснительная записка .....</b>	<b>3</b>
<b>2. Общая характеристика учебного предмета .....</b>	<b>3</b>
<b>3. Место курса в учебном плане. ....</b>	<b>4</b>
<b>4. Личностные, метапредметные и предметные результаты освоения учебного предмета. ....</b>	<b>4</b>
<b>5. Содержание курса .....</b>	<b>6</b>
<b>6. Поурочное планирование .....</b>	<b>7</b>
<b>7. Планируемые результаты обучения.....</b>	<b>21</b>

## **1. Пояснительная записка**

Программа данного курса посвящена обучению школьников началам программирования на примере графического языка Scratch, а также умению работать с данными в текстовых документах. Занятия курса направлены на развитие мышления, логики, творческого потенциала учеников. Программа ориентирована на использование получаемых знаний для разработки реальных проектов. Курс содержит большое количество творческих заданий (именуемых Кейсами).

### **Цель и задачи обучения**

Целью изучения предмета «Информатика» является получение теоретических и практических знаний, умений и навыков в области современной информатики; формирование целостного мировоззрения, соответствующего современному уровню развития науки и общественной практики, учитывающего социальное, культурное, языковое, духовное многообразие современного мира.

Для достижения поставленной цели необходимо решение следующих задач: создание условий для развития интеллектуальных и творческих способностей учащихся, необходимых для успешной социализации и самореализации личности;

формирование информационной и алгоритмической культуры;

развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составить и записать алгоритм;

формирование умений формализации и структурирования информации, умения выбирать способ представления данных в соответствии с поставленной задачей;

овладение важнейшими общеучебными умениями и универсальными учебными действиями (формулировать цели деятельности, планировать ее, находить и обрабатывать необходимую информацию из различных источников, включая Интернет и др.).

## **2. Общая характеристика учебного предмета**

Программа по предмету «Информатика» предназначена для изучения курса информатики учащимися основной школы.

Она включает в себя два блока:

- Программирование в Scratch
- Работа с текстовым процессором LibreOffice.org Writer

Важная задача изучения этих содержательных линий в курсе – добиться систематических знаний, необходимых для самостоятельного решения задач, в том числе и тех, которые в самом курсе не рассматривались. На протяжении первой части курса учащиеся изучают базовые основы программирования на примере графического языка Scratch.

**Технологии, используемые в образовательном процессе:**

- Технологии традиционного обучения для освоения минимума содержания образования в соответствии с требованиями стандартов;

технологии, построенные на основе объяснительно-иллюстративного способа обучения. В основе – информирование, просвещение обучающихся и организация их репродуктивных действий с целью выработки у школьников общеучебных умений и навыков.

- Технологии компьютерных практикумов.
- Игровые технологии.
- Тестовые технологии.
- Технологии реализации межпредметных связей в образовательном процессе.
- Технологии дифференцированного обучения для освоения учебного материала обучающимися, различающимися по уровню обучаемости, повышения познавательного интереса.
- Технология проблемного обучения с целью развития творческих способностей обучающихся, их интеллектуального потенциала, познавательных возможностей. Обучение ориентировано на самостоятельный поиск результата, самостоятельное добывание знаний, творческое, интеллектуально-познавательное усвоение учениками заданного предметного материала.
- Личностно-ориентированные технологии обучения, способ организации обучения, в процессе которого обеспечивается всемерный учет возможностей и способностей обучаемых и создаются необходимые условия для развития их индивидуальных способностей.
- Информационно-коммуникационные технологии.
- Технология коллективных методов обучения (работа в парах постоянного и сменного состава)

**Формы организации образовательного процесса:** фронтальные, групповые, индивидуальные, индивидуально-групповые, практикумы; урок-консультация, урок-практическая работа, уроки с групповыми формами работы, уроки-конкурсы.

### **3. Место курса в учебном плане.**

Данная программа предусматривает на реализацию программы «Точка роста» по информатике в 5 классе 51 часов. Рабочая программа рассчитана на 34 учебные недели, 1,5 часа в неделю, общее количество часов — 51. Рабочая программа может реализовываться с использованием электронного обучения (ЭО) и дистанционных образовательных технологий (ДОТ).

### **4. Личностные, метапредметные и предметные результаты освоения учебного предмета.**

**Личностными результатами, формируемыми при изучении предмета информатика, являются:**

- формирование ответственного отношения к учению, готовности и способности обучающихся к саморазвитию и самообразованию на основе мотивации к обучению и познанию, осознанному выбору и

построению дальнейшей индивидуальной траектории образования на базе ориентировки в мире профессий и профессиональных предпочтений, с учётом устойчивых познавательных интересов;

- формирование целостного мировоззрения, соответствующего современному уровню развития науки и общественной практики, учитывающего социальное, культурное, языковое, духовное многообразие современного мира;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, общественно полезной, учебно-исследовательской, творческой и других видов деятельности.

### **Метапредметные результаты изучения предмета «Информатика»:**

- умение самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учёбе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности;
- умение самостоятельно планировать пути достижения целей, в том числе альтернативные, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;
- умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата, определять способы действий в рамках предложенных условий и требований, корректировать свои действия в соответствии с изменяющейся ситуацией;
- умение оценивать правильность выполнения учебной задачи, собственные возможности её решения;

владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;

- умение определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логическое рассуждение, умозаключение (индуктивное, дедуктивное и по аналогии) и делать выводы;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;
- умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками; работать индивидуально и в группе: находить общее решение и разрешать конфликты на основе согласования позиций и учёта интересов; формулировать, аргументировать и отстаивать своё мнение;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий.

## **Предметные результаты изучения предмета «Информатика»:**

- формирование представления об основных изучаемых понятиях курса;
- формирование информационной и алгоритмической культуры; формирование представления о компьютере как универсальном устройстве обработки информации; развитие основных навыков и умений использования компьютерных устройств;
- развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составить и записать алгоритм для решения конкретной задачи;
- формирование умений формализации и структурирования информации, умения выбирать способ представления данных в соответствии с поставленной задачей, с использованием соответствующих программных средств обработки данных;
- формирование представления о том, что значит “программировать” на примере языка Scratch, формирование умения составлять сценарии проектов среды Scratch;
- знакомство с основными алгоритмическими структурами — линейной, условной и циклической;
- формирование умения тестировать и оптимизировать алгоритмы исполнителей;
- формирование умения создавать и редактировать документы в текстовом процессоре;
- формирование умения размещать документы в облачном хранилище, организовывать коллективную работу с документами, настраивать права доступа к документам;
- формирование умения формализации и структурирования информации;
- использование готовых прикладных компьютерных программ по выбранной специализации;
- развитие умений применять изученные понятия, результаты, методы для решения задач практического характера и задач из смежных дисциплин с использованием при необходимости справочных материалов, компьютера;
- формирование навыков и умений безопасного и целесообразного поведения при работе с компьютерными программами и в Интернете, умения соблюдать нормы информационной этики и права.

## **5. Содержание курса**

### **Программирование в Scratch (30 часа)**

Знакомство со средой программирования Scratch. Создание и сохранение документа. Понятия спрайта, сцены, скрипта. Библиотека персонажей. Исполнитель Scratch. Основные инструменты встроенного графического редактора программной среды SCRATCH. Линейный алгоритм. Создание блок-схемы. Рисование линий исполнителем Scratch. Конечный и бесконечный циклы. Цикл в цикле. Анимация исполнителя Scratch на основе готовых костюмов. Дублирование исполнителей. Алгоритмы с ветвлением. Цикл с условием. Перемещение исполнителей между слоями. Программирование клавиш. Управление событиями. Координатная плоскость.

Создание списков. Использование подпрограмм. Отладка программ с ошибками.

### **Работа с текстовым процессором LibreOffice.org Writer (21 часа)**

Загрузка и установка LibreOffice. Интерфейс редактора. Стандартные действия. Форматирование документа: шрифты, стили, размер шрифта. Работа с цветом. Сложное форматирование. Использование списков. Колонтитулы. Изображения в текстовых документах. Графика в текстовых документах. Таблицы в документах. Работа с Google-docs.

## **6. Поурочное планирование**

### **Модуль 1. Программирование в Scratch**

#### **Урок 1. Знакомство со средой программирования Scratch**

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

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

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

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

#### **Кейс 1. Научи кота бегать и мяукать.**

##### **Исполнитель Scratch, цвет и размер пера.**

Понятие ИСПОЛНИТЕЛЯ.

Команды управления пером: «Опустить перо», «Поднять перо», «Очистить», «Установить цвет пера», «Установить размер пера».

#### **Кейс 2. Рисуем разноцветные лужи и облака для прогулки кота.**

##### **Урок 2. Основные инструменты встроенного графического редактора программной среды SCRATCH.**

Инструменты растрового графического редактора — кисточка, ластик, заливка, квадрат, круг, линия.

Копирование, поворот, горизонтальное отражение, вертикальное отражение во встроенном редакторе программной среды Scratch.

Командный блок внешность (фиолетовый) — команды начальной установки эффекта цвет «Установить эффект цвет в значение 0 и команда начальной установки размера «Установить размер 100%».

Команды: «Изменить цвет эффект на \_», «Изменить размер на \_»

Командный блок управления пером (зелёный) — команда «Печать» для копирования графического изображения исполнителя в нужном месте экрана. Эффекты, которые могут быть применены к графическим изображениям действующего исполнителя.

В графическом редакторе можно создавать сложные графические изображения, почти не прикладывая собственных усилий, а лишь правильно применяя встроенные возможности программной среды.

### **Кейс 3. Свободное рисование**

Придумай, чем можно дополнить проект с прошлого урока (Нарисовать дерево? Воздушный шар? Дом?) и реализуй это.

### **Урок № 3. Алгоритм. Линейный алгоритм. Создание блок-схемы.**

#### **Основные графические примитивы векторного редактора**

#### **LibreOffice.Draw.**

Создание собственных изображений в других программах (например, LibreOfficeDraw) и импортирование их в программную среду Scratch.

Знакомство с основными графическими примитивами векторного редактора LibreOfficeDraw. Возможность создания геометрических фигур без внутренней заливки, но с текстовым блоком внутри. Стрелки, их направление. Алгоритм, блок-схема как способ записи.

### **Кейс 4. Запиши мой алгоритм!**

Ребята разбиваются на пары, в паре формулируют друг другу какую-то повседневную задачу, для решения которой нужен линейный алгоритм, придумывают этот алгоритм и рисуют блок-схему для задачи, заданной им напарником.

#### **Линейный алгоритм. Рисование линий исполнителем Scratch.**

Решение поставленной задачи в виде последовательного выполнения команд.

Создание блок-схемы линейного алгоритма средствами редактора векторной графики.

Последовательное выполнение команд.

Изменение параметров пера.

### **Кейс 5. Что бывает полосатое?**

Придумайте линейный алгоритм для создания полосатого объекта (заранее придумайте, что это будет – шарф, зебра, забор?) Создайте блок-схему этого алгоритма и реализуйте алгоритм в среде Scratch.

### **Урок № 4. Линейный алгоритм. Исполнитель Scratch рисует квадраты и прямоугольники линейно.**

Создание алгоритма для рисования исполнителем квадрата путем последовательного выполнения команд.

Рисование линейного алгоритма, состоящего из двух колонок блоков команд.

Использование векторного редактора офисного пакета LibreOffice в качестве инструмента для создания блок-схем.

Выбор нужного значения из предлагаемого списка вариантов.

Отладка программы для получения верного результата.

Команда «повернуть в направление».

Пошаговое выполнение программы для её отладки.

Центр костюма исполнителя Scratch.

### **Урок №5. Конечный цикл. Scratch рисует квадраты, линии.**

Сохранение готовых программ для дальнейшего использования.

Рисование блок-схемы циклического алгоритма.

Использование векторного редактора офисного пакета LibreOffice в качестве инструмента для создания блок-схем;



Использование команд поворота на прямой угол (90°) по часовой и против часовой стрелки;

Использование циклического алгоритма для рисования исполнителем квадрата;

Оптимизация линейного алгоритма за счёт использования циклической конструкции в программной среде Scratch;

Команда открыть... из пункта меню File;

Команда сохранить как... из пункта меню File;

Циклический алгоритм;

Блок-схема циклического алгоритма;

Команды: «повернуться на 90° по часовой стрелке»,

«повернуться на 90° против часовой стрелки», «повторить \_».

Упражнение. Сравнение двух алгоритмов, рисующих квадрат.

Идти 100 шагов Повторить 4

Повернуться на 90° Идти 100 шагов

Идти 100 шагов Повернуться на 90°

Повернуться на 90°

Идти 100 шагов

Повернуться на 90°

Идти 100 шагов

Повернуться на 90°

Рекомендуется объяснить, что команды, обеспечивающие повторяющееся выполнение одной или нескольких команд называются циклическими, показать, что использование команды цикла всегда сокращает количество команд в программе, а значит делает её более красивой. В более короткой программе легче обнаружить ошибку или неточность.

## **Урок №6. Конечный цикл. Scratch рисует несколько линий и фигур.**

### **Копирование фрагментов программы.**

Использование операции цикла для решения учебных задач.

Применение поворота на прямой угол (90°) при создании геометрических фигур и перемещении исполнителя.

Оптимизация линейного алгоритма за счёт использования циклической конструкции в программной среде Scratch.

Тело цикла.

Конечный и бесконечный циклы.

Блок-схема бесконечного цикла.

Имя спрайта и костюма.

Изменение костюма исполнителя.

Копирование фрагмента программы.

Команды: «Следующий костюм», «Перейти к костюму».

Рекомендуется объяснить, что циклические конструкции могут применяться не только для рисования фигур. Например, в цикле можно менять внешность исполнителя.

### **Кейс 6. Создай картинку из квадратов.**

Предложите ребятам нарисовать свою картинку из квадратов, может быть, это будет лестница, может быть, ковер, может быть, что-то еще.

## **Урок №7. Циклический алгоритм. Цикл в цикле.**

Использование операции цикла в цикле для решения учебных задач.

Использование поворота на прямой угол ( $90^\circ$ ) по часовой и против часовой стрелки.

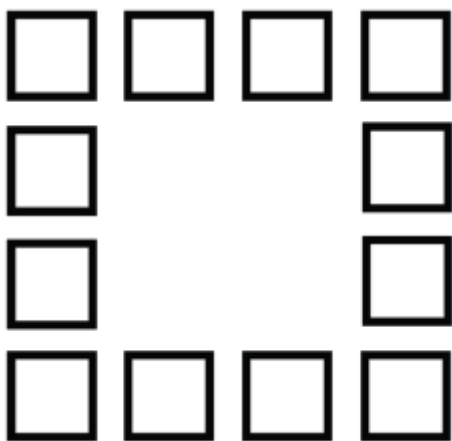
Оптимизация алгоритма за счёт использования конструкции «цикл в цикле».

Блок-схема конструкции цикл в цикле.

Например, расписание уроков составляется на неделю, в месяце в среднем четыре недели. Если расписание не меняется в течение учебного года, то получаем цикл из девяти месяцев по четыре недели с одинаковым расписанием.

Упражнение. Использование циклов «повторить \_»

Нарисуйте квадраты, как показано на рисунке, используя конструкцию цикл в цикле.



## **Урок №8. Цикл в цикле. Повторение пунктирной линии с поворотом.**

### **Блок-схема цикла.**

Использование конструкции «цикл в цикле».

Создание и реализация алгоритма рисования квадрата несплошными линиями;  
Оптимизация алгоритма за счёт использования конструкции «цикл в цикле» в программной среде Scratch;

Использование операции копирования внешности исполнителя путём копирования костюма.

Несплошные линии.

Алгоритм рисования несплошных линий с использованием циклических конструкций.

Рисование квадрата несплошными линиями, используя конструкцию «цикл в цикле».

При перемещении исполнителя можно рисовать линии отличные от сплошной. Эти возможности определяются использованием команд из блока ПЕРО.

### **Кейс 7. Мой необычный дом.**

Предложите ребятам нарисовать дом и что-то рядом с ним, используя линейные и циклические алгоритмы, а также смену костюмов Исполнителя. Может быть, над домом будут звезды разного размера, а около дома будет пунктирная разноцветная дорога, или деревья разного размера.

## **Урок №9. Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов.**

Бесконечный цикл.

Анимация исполнителя с помощью смены костюмов.

Эффект «призрак».

Изменение размера исполнителя.

Интерактивное взаимодействие с исполнителем с помощью клавиатуры.

Команды «спросить» и «думать».

**Сцена как исполнитель. Создаем модель таймера.**

Анимация сцены (фона).

Синхронная анимация сцены и исполнителя.

**Урок №10. Одинаковые программы для нескольких исполнителей.**

Анимация с помощью вращения.

Дублирование исполнителей.

При копировании исполнителей копируются и их скрипты.

Использование одинаковых программ, но разных костюмов у исполнителей.

**Урок №11. Параллельное выполнение действий несколькими исполнителями**

Дублирование исполнителей.

Синхронное выполнение скриптов.

При параллельном выполнении скриптов результат работы всех исполнителей получается гораздо быстрее, чем в тех задачах, когда работал один исполнитель.

**Урок №12. Разбиение программы на части для параллельного выполнения исполнителями. Таймер.**

Планирование действий разных исполнителей во времени для решения общей задачи.

Сенсор «таймер»

Сброс «таймера»

Команда «ждать до ...»

**Кейс 8. Мини-проект «Смена времени суток».**

Создайте скрипт, в котором будет четыре сцены: «утро», «день», «вечер», «ночь». Используйте «таймер» для своевременного изменения сцен, своевременного запуска скриптов разных исполнителей.

**Урок №13. Два исполнителя со своими программами. Мини-проект «Часы с кукушкой».**

Координаты в Scratch.

Команда «идти в x: \_ y: \_» (переместиться в точку с координатами (x,y))

Команда «плыть \_ секунд в точку x: \_ y: \_» (плавно двигаться в точку с координатами (x,y))

Команды «спрятаться», «показаться».

**Кейс 9. Мини-проект «Часы с кукушкой».**

Создайте сцену в виде часов, и три исполнителя: часовую стрелку, минутную стрелку и кукушку. Стрелки должны вращаться на циферблате. При наступлении каждого часа (т.е. в \_ часов : 00 минут) должна появляться кукушка.

**Урок №14. Алгоритмы с ветвлением. Условие ЕСЛИ.**

Еще из сказок мы знаем, что на камне было написано: «Налево пойдешь – коня потеряешь, направо пойдешь – себя не найдешь, прямо пойдешь – в сказку попадешь»

Блок-схема алгоритма с ветвлением.

Полная форма ветвления. Команда «если \_ или»

Неполная форма ветвления. Команда «если \_»

Сенсор «касается» (касания края, или другого спрайта, или указателя мыши)

### **Урок №15. Цикл с условием. Мини-проект «Шарики в лабиринте»**

Сенсор «касается цвета»

Программируем отскок шариков от стен

### **Кейс 10. Мини-проект «Шарики в лабиринте»**

Создайте сцену в виде бильярдного стола, на котором три шарика будут двигаться с разными скоростями и отталкиваться от бортиков.

### **Урок №16. Цикл с условием. Исполнитель определяет цвет.**

Сенсор «касается цвета»

Программируем поведение исполнителя в зависимости от цвета фона.

### **Урок № 17. Оператор случайных чисел.**

Команда «Выдать случайное число от \_ до \_». Случайные перемещения исполнителя в координатной плоскости.

Случайное количество шагов.

Случайные координаты.

Поворот на случайный угол.

### **Урок №18. Перемещение исполнителей между слоями.**

Команда «Перейти в верхний слой»

Команда «Перейти назад на \_ слоев»

Эффект «Призрак»

### **Урок №19. Действия исполнителей в разных слоях.**

### **Кейс 10. Мини-проект «Дорога».**

Создайте сцену, на которой изображена дорога с двумя полосами, по обочинам дороги расположите кусты и деревья. По дороге должны двигаться автомобили. Правильно расположите автомобили, деревья и кусты по слоям.

### **Урок №20 Взаимодействие исполнителей.**

Исполнители касаются друг друга.

Команда «Касается \_»

Поведение исполнителей при столкновении.

### **Урок №21. Последовательное выполнение команд исполнителями.**

Команда «Передать»

Команда «Когда я получу»

Связи между программами разных исполнителей.

### **Урок №22. Программирование клавиш. Мини-проект «Лабиринт»**

Взаимодействие пользователя с программой.

Клавиши управления перемещением исполнителя

Команда «Изменить значение x на \_»

Команда «Изменить значение y на \_»

### **Кейс 10. Игра «Лабиринт»**

Создайте сцену в виде лабиринта. С помощью клавиш со стрелками управляйте движением маленькой мышки. Мышка должна добраться до выхода, не задевая стенок лабиринта. В случае касания стенок мышка возвращается в исходную точку.

### **Урок №23. Управление событиями.**

Передача сообщений между исполнителями и фоном.

Запуск программ после получения сообщения.

### **Урок №24. Координатная плоскость. Геометрические фигуры.**



Рисование геометрических фигур на координатной плоскости.

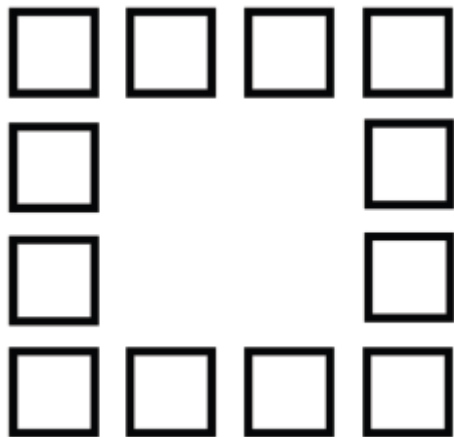
Последовательное выполнение команд одним исполнителем.

Параллельное выполнение команд несколькими исполнителями.

### **Кейс 11. Исследование времени выполнения программ.**

Создайте одного исполнителя, рисующего квадраты, как показано на рисунке.

Затем создайте четырех исполнителей, которые будут параллельно рисовать по три квадрата. Сравните время работы в этих двух случаях.



### **Урок №25. Координатная плоскость. Переменные.**

Блок «Переменные»

Рисование геометрических фигур на координатной плоскости с использованием переменных.

Построение перпендикуляров к координатным осям.

### **Урок №26. Создание списков.**

Название списка.

Элементы списка.

Длина списка.

Команда «Создать список».

Выбор элемента списка.

### **Кейс 11. Мини-проект «Викторина».**

Создайте проект «Викторина» в котором будет задано 10 вопросов.

Правильный ответ надо выбрать из списка. В случае правильного ответа очки увеличиваются на единицу. В конце выставляется оценка: «отлично» за 9 или 10, «хорошо» за 7 или 8, «удовлетворительно» за 5 или 6 правильных ответов.

### **Урок №27. Использование подпрограмм.**

Как сделать программу структурированной и более понятной.

Команда «Передать \_ и ждать»

Команда «Играть звук \_»

Команда «Ноту \_ играть \_ тактов»

### **Урок №28. Сообщество Scratch.**

Scratch 2.0.

Регистрация на сайте.

Личный кабинет.

Публикация проектов.

### **Урок №29. Отладка программ с ошибками.**

<http://scratch.mit.edu/projects/10437040>

<http://scratch.mit.edu/projects/10437249>

<http://scratch.mit.edu/projects/10437366>

<http://scratch.mit.edu/projects/10437476>

### **Уроки №30. Кейс 12. Итоговый проект.**

Учащиеся самостоятельно или в парах выполняют индивидуальный проект, согласованный с учителем. Как итог, можно провести конкурс проектов.

#### **Обучающийся научится:**

- составлять сценарии проектов среды Scratch;
- составлять алгоритмы, определять последовательность выполнения команд;
- создавать и редактировать рисунки в графическом редакторе;
- использовать обширную библиотеку готовых сцен и исполнителей;
- изменять размер, костюм, прозрачность исполнителя;
- создавать линейные алгоритмы для исполнителя;
- создавать циклические алгоритмы;
- создавать ветвящиеся алгоритмы;
- управлять одновременной работой нескольких исполнителей;
- передавать сообщения между исполнителями;
- внедрять звуковые эффекты в алгоритмы исполнителей;
- создавать алгоритмы, которые будут выполняться одновременно (параллельно) несколькими исполнителями;
- тестировать и оптимизировать алгоритмы исполнителей.

### **Модуль 2. Работа с текстовым процессором LibreOffice.org Writer.**

#### **Урок №31. Знакомство с офисным пакетом LibreOffice.**

LibreOffice — офисный пакет, совместимый с 32/64-битными системами. Поддерживает большинство популярных операционных систем, включая GNU/Linux, Microsoft Windows и Mac OS X.

LibreOffice бесплатен и имеет открытый исходный код, следовательно, его можно бесплатно скачивать и использовать.

Загрузка и установка LibreOffice.

Интерфейс редактора.

Меню, панели инструментов, строка состояния.

Режимы работы с документом.

Создание нового документа, ввод текста.

Сохранение, открытие и закрытие документа.

Редактирование текста: копирование, вставка, перемещение, удаление фрагментов текста.

Форматирование документа: шрифты, стили, размер шрифта. Работа с цветом.

Отмена действий.

#### **Сложное форматирование.**

Межстрочный интервал, выравнивание, отступ, поля.

Вставка специальных символов.

Отображение непечатаемых символов. Буквица. Водяные знаки.

#### **Кейс 1. Создание таинственного документа.**

Представьте себе, что вы создаете таинственный документ, в котором описано место нахождения сокровища. Придумайте короткий текст,

запишите его, отформатируйте его так, чтобы он красиво выглядел на листе. Добавьте водяные знаки, буквицу и какие-нибудь специальные символы.

### **Урок №32. Использование списков.**

Нумерованный список.

Маркированный список.

Многоуровневый список.

Параметры списка.

### **Кейс 2. О спорт, ты мир!**

Создайте многоуровневый список, в котором перечислите несколько, наиболее интересных вам видов спорта, у каждого вида спорта (на следующем уровне) перечислите нескольких наиболее известных атлетов, а у каждого атлета (на следующем уровне) – несколько наиболее известных его достижений. Отформатируйте ваш список так, чтобы он выглядел красиво. Для поиска информации воспользуйтесь средой интернет.

### **Колонтитулы.**

Верхний и нижний колонтитулы.

Форматирование колонтитулов.

Прозрачность.

Нумерация страниц.

### **Кейс 3. Мини-Европа.**

Начнем работу над проектом Мини-Европа. Предыстория, которую рассказывает учитель: в некоторых европейских городах, например, в Брюсселе, есть парки, в которых расположены интересные европейские объекты в миниатюре (показать видео или презентацию). Мы создадим мини-книжку, в которой будут представлены те европейские достопримечательности, которые вам бы хотелось увидеть. Выберите 4-6 стран и найдите в интернете наиболее интересные достопримечательности (например, Франция - Эйфелева башня и т.п.), а также небольшой рассказ про них (можно воспользоваться Википедией). Расположите информацию про каждую страну на отдельной странице, отформатируйте текст, добавьте нумерацию и колонтитулы (в колонтитулы можно вынести название страны или города).

### **Урок №33. Изображения в текстовых документах.**

Вставка рисунка в документ.

Форматирование рисунка.

Размер и положение рисунка.

Обтекание текстом.

Вставка фигур: линии, стрелки, многоугольники и т.п.

### **Кейс 3. Продолжение работы над проектом Мини-Европа.**

Найдите в интернете фотографии выбранных вами достопримечательностей и разместите их в тексте.

### **Графика в текстовых документах.**

Текстовые эффекты.

Надписи.

Схемы.

Редактор формул.

### **Кейс 3. Завершение работы над проектом Мини-Европа.**

Дополните ваш проект текстовыми эффектами. Покажите вашу работу классу.

#### **Урок №34. Практическая работа.**

**Кейс 4. Создание буклета программы школьного вечера.**

**Кейс 5. Создание пригласительных билетов на школьный вечер.**

#### **Урок №35. Создание таблиц.**

Вставка таблицы в документ.

Добавление строк и столбцов в таблицу.

Ширина столбцов и высота строк таблицы.

Внешние и внутренние границы таблицы.

#### **Урок №36. Работа с таблицами.**

Название таблицы.

Объединение ячеек. Разбиение ячеек.

Направление текста в ячейке.

Изменение фона ячейки, таблицы.

Изменение границ таблицы.

**Кейс 6. Создание таблицы графика дежурств в классе.**

#### **Урок №45. Печать документа.**

Параметры страницы.

Нумерация страниц.

Выбор страниц для печати.

Предварительный просмотр.

Печать документа.

**Практическая работа. Создание сложных таблиц.**

**Кейс 7. Создайте таблицу Расписание уроков. Придумайте, как сделать ее необычной, какого цвета будут ячейки, как будут располагаться столбцы?**

Самый простой вариант

День		Предмет	День		Предмет
Понедельник	1		Четверг	1	
	2			2	
	3			3	
	4			4	
	5			5	
	6			6	
Вторник	1		Пятница	1	
	2			2	
	3			3	
	4			4	
	5			5	
	6			6	
Среда	1		Суббота	1	
	2			2	
	3			3	
	4			4	
	5			5	
	6			6	



## **Кейс 8. Создай свою визитку.**

### **Урок №37. Знакомство с Google-docs.**

Бесплатный сервис.

Облачное хранилище.

Обмен файлами.

Кроссплатформенность.

Коллективная работа.

### **Начало работы с Google-docs.**

Учетная запись Gmail.

Google – диск.

## **Кейс 9 (большой, на всю тему изучения Google-документов). Создание папки нашего класса. Начало.**

Каждый ребенок создает свой Google-документ, в котором будет собирать небольшой рассказ о себе, своих увлечениях. Учитель создает общую папку, а также Google-таблицу, в которой содержатся ссылки на странички ребят.

### **Урок №38. Работа с текстом в Google-docs.**

Ввод и редактирование текста.

Панель инструментов.

## **Кейс 9. Где я ошибся?**

Работа в парах. Ученик создает гугл-документ, который открывает на доступ своему напарнику. В созданном документе он набирает (или берет из интернета небольшой текст), в котором нарочно делает ошибки. Напарник должен обнаружить все ошибки и их прокомментировать с помощью инструмента Комментарий.

### **Работа с текстом в Google-docs.**

Форматирование текста: шрифт, стиль, цвет, размер, выравнивание.

Отступ. Межстрочный интервал.

Экранная клавиатура.

**Кейс 9. Продолжение.** Начнем наполнять файлы с информацией об учениках. Каждый ученик в своем файле пишет краткий рассказ про себя (меня зовут ..., я учусь ..., я увлекаюсь..., этим летом я был..., я хочу стать ... и т.п.) и красиво форматирует текст.

### **Урок №39. Работа с изображениями.**

Вставка рисунка.

Размер рисунка, поворот, положение, прозрачность, обтекание текстом.

**Кейс 9. Продолжение.** Продолжаем наполнять файлы с информацией об учениках. Ребята дополняют файлы с рассказом о себе изображениями (часть, возможно, потребуется принести из дома, например, свою фотографию, фотографию домашнего животного, или фото из путешествий). К некоторым изображениям стоит добавить интересные эффекты.

### **Урок №40. Работа со списками.**

Маркированный список.

Нумерованный список.

Шаблоны списков.

Сброс нумерации.

**Кейс 9. Продолжение.** Дополните свой файл списком стран, в которых вы бы хотели побывать, к каждой стране добавьте вложенным списком

несколько городов, к каждому городу – несколько достопримечательностей этого города (тоже вложенным списком).

#### **Урок №41. Работа с таблицами.**

Вставка таблицы в документ.

Вставка и удаление строк и столбцов в таблицу.

Ширина и высота ячеек таблицы.

**Кейс 9. Продолжение.** Дополните свой файл таблицей с вашим расписанием на неделю, и расписанием кружков.

#### **Урок №42. Работа с таблицами.**

Объединение ячеек.

Цвет фона.

Цвет границ.

Толщина границ.

Стиль.

**Кейс 9. Продолжение.** Красиво раскрасьте таблицу со своим расписанием, созданную на прошлом уроке.

#### **Урок №43. Работа с диаграммами.**

Вставка диаграммы в документ.

Google – Таблица.

Виды диаграмм: линейчатая, столбчатая, круговая, график.

Область диаграммы, название, легенда.

Расширенные настройки.

Функция sparkline для создания миниатюрной диаграммы внутри ячейки.

#### **Кейс 11. Как я учусь?**

Возьмите свой дневник и создайте таблицу вида:

	Сколько пятерок	Сколько четверок	Сколько троек	Сколько двоек
Математика				
Русский язык				
Английский язык				
География				
Физкультура				

Заполните ячейки с помощью дневника, записав туда все свои оценки, полученные за прошедшее время. Какие диаграммы вы сможете построить по этой таблице?

#### **Урок №44. Работа с рисунками.**

Графический редактор внутри Google Docs.

Линии, фигуры, текстовые поля.

Инструмент Word Art.

**Кейс 9. Продолжение.** С помощью изученным инструментов украсьте свой файл.

**Урок №45. Работа с формулами.**

Вставка формулы в документ.

Редактор формул.

**Кейс 12. Задачи для друга.**

Возьмите учебник математики (или найдите задачи в интернете) и создайте для друга гугл-документ с пятью задачами, например, на решение уравнений.

**Урок №46. Настройка стилей в Google – docs.**

Создание своих стилей.

Шрифт, размер, цвет, междустрочный интервал, эффекты.

**Урок №47. История изменений в Google – docs.**

Хронология изменений документа.

Возврат к нужной версии документа.

**Кейс 12. Вернись назад.**

Разбейтесь на пары. Зайдите в файл (из кейса 9) своего напарника, и внесите в него несколько изменений. Задача каждого ученика – восстановить версию до исправления.

**Урок №48. Полезные сервисы в Google – docs.**

Статистика.

Проверка правописания.

Автозамена.

Голосовой ввод.

GOOGLETRANSLATE и DETECTLANGUAGE

**Кейс 13. Таинственный текст.**

Разбейтесь на пары. Создайте Google-таблицу. Запишите таинственное послание на русском языке, затем с помощью любого онлайн-переводчика переведите это послание на какой-нибудь язык, отличный от русского и английского, запишите результат в ячейке своей таблицы. Откройте доступ к таблице вашему напарнику. С помощью функций DETECTLANGUAGE определите, на каком языке записан текст в файле, который вам открыли. С помощью функции GOOGLETRANSLATE переведите этот текст на русский язык.

**Урок №49. Настройки доступа в Google – docs.**

Доступ по ссылке.

Уровни доступа: выключено, просматривать, комментировать, редактировать.

Доступ определенным пользователям.

Общий доступ.

Доступ к папке с файлами.

**Совместная работа над документом.**

Комментарии.

Посоветовать правки.

Закладки.

Интеграция с Google Keep.

Веб-буфер обмена.

## **Урок №50. Горячие клавиши.**

Сервис. Навигация.

Форматирование текста.

Форматирование абзаца.

Редактирование.

### **Плагины для Google – docs.**

Установка плагинов из интернет-магазина Chrome

Google Docs offline – позволяет создавать документы при отсутствии подключения к Интернет.

### **Галерея шаблонов в Google – Docs.**

При создании стандартного документа, например, письмо или резюме можно воспользоваться заранее настроенным шаблоном.

### **Кейс 14. Шаблон-реферат**

С помощью шаблона Реферат подготовьте небольшой рассказ о своем увлечении, может быть, это любимый вид спорта, решение задач, а, возможно, вы напишете о любимом домашнем животном.

## **Урок №51.**

### **Кейс 15. Любимые блюда нашего класса**

Каждый ученик с помощью шаблона Рецепт создает рецепт своего любимого блюда. Затем в созданную учителем таблицу каждый добавляет строку со своей фамилией и именем и ссылкой на файл, содержащий рецепт.

Публикация документов.

Организация коллективной работы с документами.

Настройка прав доступа.

### **Кейс 16. Окончательное оформление папки класса.**

На общем диске класса каждый ученик создает папку со своей фамилией, в папке размещает все созданные им в течение учебного года материалы. В общей папке должны быть также файл, содержащий ссылки на личные страницы и файл, содержащий ссылки на рецепты.

### **Обучающийся научится:**

- создавать и редактировать документы в текстовом процессоре;
- работать с панелями инструментов текстового процессора;
- работать с блоками текста: выделять, копировать, удалять;
- использовать необходимые шрифты;
- форматировать документ;
- вставлять в документ таблицы, схемы, рисунки;
- создавать и редактировать документы в Google – docs;
- работать с инструментами Google – docs;
- размещать документы в облачном хранилище;
- организовывать коллективную работу с документами;
- настраивать права доступа к документам.



Литература:

1. Сорокина Т.Е. МОДУЛЬ «ПРОПЕДЕВТИКА ПРОГРАММИРОВАНИЯ СО SCRATCH»

Видео-, аудиоматериалы:

1. Видеоуроки по Scratch <http://www.youtube.com/watch?v=vd20J2r5wUQ>

Цифровые ресурсы:

Курс «Введение в Scratch» [http://window.edu.ru/resource/056/78056/files/scratch\\_lessons.pdf](http://window.edu.ru/resource/056/78056/files/scratch_lessons.pdf)

<https://scratch.mit.edu/>

<https://ru.libreoffice.org/>

Полное руководство по Google Docs: все, о чем вы не знали, но боялись спросить <https://texterra.ru/blog/polnoe-rukovodstvo-po-google-docs.html>

## **7. Планируемые результаты обучения.**

**Важнейшими умениями/знаниями являются следующие:**

- умение пользоваться персональным компьютером и его периферийным оборудованием;
- умение следовать требованиям техники безопасности, гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий;
- умение осуществлять взаимодействие посредством электронной почты, чата, форума;
- умение искать информацию с применением правил поиска (построения запросов), в компьютерных сетях, некомпьютерных источниках информации (справочниках и словарях, каталогах, библиотеках) при выполнении заданий и проектов по различным учебным дисциплинам;
- умение составлять сценарии проектов среды Scratch;
- умение составлять алгоритмы, определять последовательность выполнения команд; использовать обширную библиотеку готовых сцен и исполнителей;
- умение создавать линейные алгоритмы для исполнителя; умение создавать циклические и ветвящиеся алгоритмы;
- умение управлять одновременной работой нескольких исполнителей; умение передавать сообщения между исполнителями;
- умение тестировать и оптимизировать алгоритмы исполнителей;
- умение создавать и редактировать документы в текстовом процессоре;
- умение работать с блоками текста: выделять, копировать, удалять; использовать необходимые шрифты; форматировать документ;
- умение создавать и редактировать документы в Google – docs; работать с инструментами Google – docs;
- умение размещать документы в облачном хранилище; организовывать коллективную работу с документами; настраивать права доступа к документам;
- умение выбирать способ представления своего проекта с использованием соответствующих программных средств.

# Тематическое планирование уроков

№ урока	Тема урока
<b>Модуль 1. Программирование в Scratch.</b>	
1.	Знакомство со средой программирования Scratch. Исполнитель Scratch, цвет и размер пера.
2.	Основные инструменты встроенного графического редактора программной среды SCRATCH
3.	Алгоритм. Линейный алгоритм. Создание блок-схемы. Линейный алгоритм. Рисование линий исполнителем Scratch.
4.	Линейный алгоритм. Исполнитель Scratch рисует квадраты и прямоугольники линейно.
5.	Конечный цикл. Scratch рисует квадраты, линии.
6.	Конечный цикл. Scratch рисует несколько линий и фигур. Копирование фрагментов программы
7.	Циклический алгоритм. Цикл в цикле.
8.	Цикл в цикле. Повторение пунктирной линии с поворотом. Блок-схема цикла.
9.	Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов
10.	Одинаковые программы для нескольких исполнителей.
11.	Параллельное выполнение действий несколькими исполнителями
12.	Разбиение программы на части для параллельного выполнения исполнителями. Таймер
13.	Два исполнителя со своими программами. Мини-проект «Часы с кукушкой».
14.	Алгоритмы с ветвлением. Условие ЕСЛИ
15.	Цикл с условием. Мини-проект «Шарики в лабиринте»
16.	Цикл с условием. Исполнитель определяет цвет.
17.	Перемещение исполнителей между слоями.
18.	Действия исполнителей в разных слоях
19.	Взаимодействие исполнителей.
20.	Последовательное выполнение команд исполнителями
21.	Программирование клавиш. Мини-проект «Лабиринт»
22.	Управление событиями
23.	Координатная плоскость. Геометрические фигуры.
24.	Координатная плоскость. Переменные
25.	Создание списков
26.	Мини-проект «Викторина»
27.	Использование подпрограмм.
28.	Сообщество Scratch
29.	Отладка программ с ошибками
30.	Итоговый проект
<b>Модуль 2. Работа с текстовым процессором LibreOffice.org Writer.</b>	
31.	Знакомство с офисным пакетом LibreOffice. Сложное форматирование.
32.	Использование списков. Колонтитулы
33.	Изображения в текстовых документах. Графика в текстовых документах
34.	Практическая работа.
35.	Создание таблиц
36.	Работа с таблицами
37.	Печать документа
38.	Знакомство с Google-docs
39.	Работа с текстом в Google-docs.

40.	Работа с изображениями.
41.	Работа со списками
42.	Работа с таблицами.
43.	Работа с диаграммами
44.	Работа с рисунками
45.	Работа с формулами
46.	Настройка стилей в Google – docs.
47.	История изменений в Google – docs.
48.	Полезные сервисы в Google – docs.
49.	Настройки доступа в Google – docs
50.	Горячие клавиши.
51.	Практическая работа