

Государственное бюджетное общеобразовательное учреждение  
«Средняя школа № 14 городского округа Иловайск»  
Донецкой Народной Республики

РАССМОТРЕНО

на заседании ШМО

Протокол от «31» августа 2024 г.  
№ 10

Руководитель ШМО

  
А. А. Фоменко

СОГЛАСОВАНО

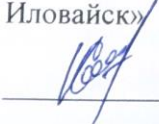
зам. директора по УВР

 Е.И. Дятлова

«29» августа 2024 г.

УТВЕРЖДАЮ

Директор ГБОУ «СШ № 14 г.о.  
Иловайск»

 О.М. Камлева

Приказ № 126

« 29» августа 2024 г.

М.П.

**КАЛЕНДАРНО – ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ**

**Курса внеурочной деятельности**

**«Программирование в Scratch»**

для 5-6 классов

Разработал учитель информатики:

Фоменко Александр Андреевич

2024— 2025 учебный год

## 1. Пояснительная записка

### **Вклад учебного предмета в достижение целей основного общего образования**

Scratch — это простой, понятный и невероятно веселый язык программирования для детей. В нем нет кодов, которые нужно знать наизусть и писать без ошибок. Все, что требуется, — это умение читать и считать. Как из конструктора Lego, при помощи Scratch можно собирать программы из разноцветных «кирпичиков» — блоков. В программу можно вносить любые изменения в любой момент и сразу видеть, как она работает

Scratch (читается как «скрэтч») — визуальный язык программирования, богатая обучающая среда для людей всех возрастов. Он позволяет создавать интерактивные мультимедийные проекты: мультфильмы, книжные обзоры, научные эксперименты, игры и симуляторы. В визуальной среде Scratch вы можете изучить те области знаний, которые были вам недоступны. В нем есть полный набор мультимедийных инструментов, с помощью которых легко создавать чудесные приложения. Причем это гораздо проще, чем в других языках программирования! Scratch хорошо помогает в развитии навыка решения задач. А он важен во всех областях жизни, не только в программировании. В этой среде вы сразу получаете обратную связь и легко и быстро можете проверить свою логику. Визуальная структура позволяет очень просто отслеживать все шаги программ и развивать свое мышление. В целом благодаря Scratch легко понять основы компьютерной науки. Появляется мотивация для учебы и тяга к знаниям; дети учатся в процессе, самостоятельно, исследуя разные области и совершая открытия.

Scratch разработан в 2006г. под руководством Митчела Резника группой Lifelong Kindergarten в лаборатории Media Lab Массачусетского технологического института.

### **Общие цели учебного предмета для уровня обучения**

Цель курса — обучить основам программирования, и Scratch здесь — только средство. А еще показать возможности Scratch как мощного инструмента для обучения. Легко начать обучение, изобретать (создавать свои программы) и соответственно развивать логическое мышление, творческое воображение, креативное мышление.

Скретч - это новая бесплатная среда программирования, которая позволяет детям от 7 лет создавать собственные истории, мультфильмы, игры и другие произведения. Эта программа позволяет детям выразить себя в компьютерном творчестве. Скретч специально разрабатывался как новая учебная среда для обучения школьников программированию. В Скретче можно создавать мультфильмы, играть с различными объектами, видоизменять их вид, перемещать их по экрану, устанавливать формы взаимодействия между объектами. Программы создаются из разноцветных блоков точно так же, как собираются из разноцветных кирпичиков конструкторы Лего.

Целью так же является пропедевтика программирования, математики, естественных наук среди школьников младших классов:

- получить представление об алгоритмах и исполнителях, основных алгоритмических конструкциях языков программирования;
- обеспечить высокую мотивацию к проектной деятельности и дальнейшему изучению языков программирования;
- формировать компьютерную грамотность.

### **Особенности Рабочей программы по предмету**

Программа используется как дополнительный модуль «элективный курс» к основной программе по информатике.

Программа курса имеет инженерно-техническую направленность. Многие задания составлены таким образом, чтобы они решались методами учебно-исследовательской и проектной деятельности.

Большинство заданий встречаются в различных темах для того, чтобы показать возможности решения одной и той же задачи или проблемы различными средствами, обеспечивающими достижение требуемого результата, что в итоге должно привести к выработке способности оптимизировать решение поставленной задачи или проблемы.

Практикум входит в УМК по информатике для 5-6 классов авторского коллектива под руководством Л. Л. Босовой. Практикум предназначен для внеурочной деятельности со школьниками, направленной на формирование информационной культуры обучающихся, развитие их алгоритмического мышления и творческих способностей в процессе освоения популярной во всём мире среды визуального программирования Scratch (Скретч). Следовательно, может применяться для элективного курса. Материал пособия, построенный на основе проектно-исследовательского подхода, соответствует требованиям ФГОС. Большое количество и разнообразие заданий в практикуме даёт учителю возможность адаптировать материал к фактическому уровню подготовки учащихся и выделенному на освоение материала объёму учебного времени.

Данная программа по предмету «Информатика» элективный курс «Практикум по программированию в среде Scratch» основана на учебно-методическом комплекте (далее УМК) по информатике расширенный комплект для углубленного изучения авторов Т.Е. Сорокина, А.Ю. Босова; под ред. Л.Л. Босовой издательства «БИНОМ. Лаборатория знаний».

**Рабочая программа учебного предмета «Информатика» элективный курс «Практикум по программированию в среде Scratch» на уровне расширенный комплект для углубленного изучения информатики разработана на основе:**

- Федерального закона «Об образовании в Российской Федерации» от 29.12.2012 г. № 273-ФЗ (ред. От 31.12.2014 г., с изм. От 02.05.2015 г., с изм. и доп., вступ. в силу с 3.03.2015 г.)

- Федерального государственного образовательного стандарта основного общего образования, утвержденного приказом Министерства образования и науки Российской Федерации 17.12.2010 г. № 1897
- Приказа Министерства образования и науки Российской Федерации от 31.12.2015 г. № 1577 «О внесении изменений в федеральный государственный образовательный стандарт основного общего образования, утвержденный приказом Министерства образования и науки Российской Федерации от 17.12.2010 г. № 1897»
- Примерной основной образовательной программы основного общего образования, одобренной решением федерального учебно – методического объединения по общему образованию (протокол от 08.04.2015 г. № 1/15)
- Базисного учебного (образовательного) плана изучения информатики в основной школе на 2019-2020 учебный год
- Учебного плана ЧОУ «Многопрофильная гимназия им. М.Г. Гамзатова» на 2019 – 2020 учебный год
- Информатика. Программа для основной школы: 5–6 классы: Методическое пособие для учителей общеобразовательных учреждений /[сост. Босова Л.Л., Босова А.Ю.] – М.: БИНОМ. Лаборатория знаний, 2013.

**Рабочая программа реализуется при использовании учебно–методического комплекса:**

1. Сорокина Т. Е. Пропедевтика программирования в курсе информатики 5–6 классов на базе среды SCRATCH: мат. конф./ XII открытая Всерос. конф. (15–16 мая 2014 г.). – Казань: Казанский (Приволжский) федеральный университет, 2014. – 369 с.
2. Сорокина Т.Е., Босова А.Ю., под ред. Босова Л.Л. Информатика. Практикум по программированию в среде Scratch. 5-6 классы.– М.: БИНОМ. Лаборатория знаний, 2019.
3. Учебно-методический материал по среде Scratch на YouTube- канале педагога Сорокина Т.Е  
<https://www.youtube.com/playlist?list=PLMInhDclNR1GsZ9CJBZESbm7k3Xpr7awy>
4. Маржи М., пер. с англ. Гескина М. и Таскаева С. Scratch для детей. Самоучитель по программированию. – М.: Манн, Иванов и Фербер, 2017.
5. Онлайн-ресурсы по среде Scratch на сайте Маржи М.  
<http://nostarch.com/learnscratch/>

**Приоритетные формы и методы работы с учащимися, виды и формы контроля**

**Вид контроля на уроке:**

- Текущий
- Тематический
- Итоговый контроль

**Формы контроля знаний и умений учащихся:**

- Фронтальная.
- Индивидуальная.
- Комбинированная.

**Сроки реализации рабочей программы: 2024-2025 учебный год.**

### **Структура рабочей программы:**

1. Пояснительная записка
2. Общая характеристика учебного предмета
3. Описание место учебного предмета в учебном плане
4. Описание ценностных ориентиров содержания учебного предмета
5. Личностные, метапредметные и предметные результаты освоения учебного предмета
6. Содержание учебного предмета
7. Тематическое планирование с определением основных видов учебной деятельности
8. Описание учебно-методического и материально-технического обеспечения образовательного процесса
9. Приложения в программе

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

### **Особенности содержания учебного предмета.**

В элективном курсе информатики **«Практикум по программированию в среде Scratch»** для 5-6 класса выделяются следующие содержательные основные линии: Знакомство с программной средой Scratch, Компьютерная графика, Алгоритмы и исполнители, Арифметические операторы и функции, Движение и рисование, Внешность и звуки, Процедуры, Переменные, Проектная деятельность и моделирование процессов и систем.

Содержание линии **«Знакомство с программной средой Scratch»** формирует навык выбирать и запускать программную среду Scratch; работать с основными элементами пользовательского интерфейса программной среды; изменять размер и перемещать окно программы, выбирать необходимый режим окна; вводить имя файла с помощью клавиатуры; выбирать необходимый файл из нужной папки библиотеки программы; понимать иерархическую организацию библиотеки данных программной среды; выделять путь к элементам библиотеки

Содержание линии **«Компьютерная графика»** формирует навыки выделять фрагменты изображения для дальнейшей работы с ними; планировать работу по созданию сложных изображений путем копирования и масштабирования простых; выбирать наиболее подходящий инструмент графического редактора для создания фрагмента изображения; использовать возможности работы с цветом; планировать создание симметричных изображений; создавать сложные графические объекты путем копирования и модификации простых объектов и их фрагментов

Содержание линии **«Алгоритмы и исполнители»** формирует умение составлять и отлаживать программный код; использовать конструкции программной среды для создания линейных, разветвленных и циклических алгоритмов; организовывать параллельные вычисления; организовывать последовательность событий программы,

передачу управления от одних исполнителей другим; определять эффективный способ решения поставленной задачи

Содержание линии **«Арифметические операторы и функции»** направлено на овладение учащимися методами и средствами арифметических операторов и их функций для процесса создания простых программ.

Содержание линии **«Движение и рисование»** описание командных блоков разделов «Движение» и «Рисование», практическое применение команд движения, введение в графический редактор Scratch.

Содержание линии **«Внешность и звуки»** описание командных блоков разделов «Звуки» и «Внешность», практическое применение команд внешности, введение в звуковой редактор Scratch.

Содержание линии **«Процедуры»** направлена на изучение основы процедур в Scratch, их структуры, модульного программирования. Зачем так рано? Чтобы сразу научиться программировать грамотно.

Содержание линии **«Переменные»** использование переменных для записи информации. Также мы узнаем, как задавать вопросы пользователю и получать ответы. Это поможет в дальнейшем создавать различные интерактивные приложения.

Содержание линии **«Проектная деятельность и моделирование процессов и систем»** формирует умение создавать план появления событий для отражения определенной темы; выбирать иллюстративный материал из встроенной библиотеки; выбирать метод анимации для конкретной задачи; использовать возможности программной среды Scratch для создания мультимедийных проектов; создавать имитационные модели, интерактивные проекты и игры средствами программной среды.

### **Особенности методического аппарата УМК.**

Информационные процессы и информационные технологии являются сегодня приоритетными объектами изучения на всех ступенях школьного курса информатики. Данное методическое пособие предназначено в помощь учителю, работающему по новому учебно-методическому комплексу (УМК) элективного курса информатики **«Практикум по программированию в среде Scratch»** для 5-6 классов (авторов Т.Е. Сорокина, А.Ю. Босова; под ред. Л.Л. Босовой издательства «БИНОМ. Лаборатория знаний»), имеющему гриф «Допущено» Министерства образования РФ. В состав УМК входят:

1. Сорокина Т. Е. Пропедевтика программирования в курсе информатики 5–6 классов на базе среды SCRATCH: мат. конф./ XII открытая Всерос. конф. (15–16 мая 2014 г.). – Казань: Казанский (Приволжский) федеральный университет, 2014. – 369 с..

2. Сорокина Т.Е., Босова А.Ю., под ред. Босова Л.Л. Информатика. Практикум по программированию в среде Scratch. 5-6 классы.– М.: БИНОМ. Лаборатория знаний, 2019.
3. Учебно-методический материал по среде Scratch на YouTube- канале педагога Сорокина Т.Е  
<https://www.youtube.com/playlist?list=PLMInhDclNR1GsZ9CJBZESbm7k3Xpr7awy>
4. Маржи М., пер. с англ. Гескина М. и Таскаева С. Scratch для детей. Самоучитель по программированию. – М.: Манн, Иванов и Фербер, 2017.
5. Онлайн-ресурсы по среде Scratch на сайте Маржи М.  
<http://nostarch.com/learnscratch/>
6. Пашковская Ю.В. Творческие задания в среде Scratch: рабочая тетрадь для 5-6 классов . -- М. : Лаборатория знаний, 2018. (Школа программиста)
7. Путина А.С., под ред. Тарапаты В.В. Scratch 2.0 от новичка к продвинутому пользователю. Пособие для подготовки к Scratch-Олимпиаде. -- М. : Лаборатория знаний, 2019. (Школа юного программиста)
8. Официальный сайт Scratch <https://scratch.mit.edu/>

Прелесть программирования — в возможности творить. Это УМК обучает основам программирования и может быть использована, как задачник для учеников средней и старшей школы или самоучитель. Чтобы заниматься по этой книге, не требуется опыта программирования. И она доступна всем, кто знает математику на уровне средней школы. Если какое-то задание покажется слишком сложным, его можно пропустить без ущерба для процесса обучения.

В этом УМК представлен практический подход к обучению программированию и основам компьютерных наук. С его помощью любой ученик сможет развить свое воображение и научиться программировать. Предложено создание проектов, в которых подробно объяснены отдельные принципы, а потом совместная разработка программ, иллюстрирующая их в действии. Акцент здесь делается на решении задач, а не конкретных функциях Scratch.

Примеры в этой книге показывают широкий диапазон тем, которые возможно освоить с помощью Scratch. Они были тщательно отобраны и объясняют основы программирования и возможности глубже изучить другие темы в рамках этой среды.

Упражнения и задачи в конце каждой главы — отличный способ проверить полученные навыки программирования. В них также предлагается добавить новые ходы и объединить уже изученные принципы в более широкую задачу. Занимаясь этими упражнениями, ученики придумывают по ходу новые задания – свои собственные идеи. Умение решать собственные задачи показывает, как глубоко учащиеся понимают принципы программирования.

### **Структура курса:**

1. Знакомство со средой Scratch. Внешний вид среды, поля. Анимация.
2. Исполнитель Scratch, цвет и размер пера.
3. Основные инструменты встроенного растрового графического редактора.
4. Линейный алгоритм. Создание блок-схемы. Основные графические примитивы векторного редактора LibreOfficeDraw.
5. Линейный алгоритм. Рисование линий исполнителем Scratch.
6. Линейный алгоритм. Исполнитель Scratch рисует квадраты и прямоугольники линейно.
7. Конечный цикл. Исполнитель Scratch рисует квадраты, линии.

8. Конечный цикл. Исполнитель Scratch рисует несколько линий и фигур. Копирование фрагментов программы.
9. Циклический алгоритм. Цикл в цикле. Вложенные и внешние циклы.
10. Цикл в цикле. Повторение пунктирной линии с поворотом. Блок-схема цикла.
11. Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов.
12. Сцена как исполнитель. Создаем модель таймера.
13. Бесконечный цикл. Одна программа для исполнителя Scratch, но разные костюмы.
14. Одинаковые программы для несколько исполнителей.
15. Несколько исполнителей. Параллельное выполнение действий для ускорения процесса выполнения программы.
16. Разбиение программы на части для параллельного выполнения исполнителями. Таймер. Уменьшение показаний таймера при параллельных вычислениях.
17. Два исполнителя со своими программами. Мини-проект «Часы».
18. Алгоритмы с ветвлением. Условие ЕСЛИ. Два исполнителя.
19. Цикл при условии. Мини-проект «Шарики в лабиринте»
20. Цикл при условии. Исполнитель определяет цвета.
21. Цикл при условии. Исполнители в разных слоях. Мини-проект «Самолет сквозь облака».
22. Перемещение исполнителя из одного слоя в другой. Действия исполнителей в разных слоях. Мини-проект «Дорога».
23. Алгоритмы с ветвлением. Условие ЕСЛИ. Взаимодействие исполнителей. Блок-схема с условием.
24. Сцена как исполнитель. Последовательное выполнение команд исполнителями.
25. Алгоритмы с ветвлением. Программирование клавиш.
26. Алгоритмы с ветвлением. Если касается цвета.
27. Интерактивность исполнителей. Создание мини-проекта «Лабиринт».
28. Игра «Лабиринт». Усложнение.
29. Моделирование ситуации. Мини-проект «Пешеходный переход».
30. Моделирование ситуации. Интерактивность исполнителей. Мини-проект «Водолей».
31. Моделирование. Учебные модели «Рисующий карандаш», «Затухание».
32. Моделирование. Тестовая модель «Комнатные растения».
33. Моделирование. Обучающий проект по маршрутам географических открытий.
34. Арифметические операторы и функции.
35. Движение и рисование.
36. Внешность и звуки
37. Процедуры.
38. Переменные.
39. Урок цифры.
40. Итоговый проект.

Основная цель курса — научить программировать с нуля даже неопытного пользователя параллельно развивая творческие способности.

### **Специфика курса:**

Опыт работы с учениками 5–6 классов в среде программирования Scratch позволяет говорить о целесообразности включения в школьную программу практикоориентированного модуля «Пропедевтика программирования со Scratch» как первого шага к изучению языков программирования высокого уровня. Модуль рекомендован для изучения в 5–6 классах в качестве дополнительного к программе по



информатике для 5–9 классов авторов Босовой Л.Л. и Босовой А.Ю. Основное назначение модуля – изучение алгоритмов и исполнителей, первое знакомство с основными алгоритмическими конструкциями, используемыми в языках программирования; получение позитивного опыта отладки и написания первых завершённых программных продуктов. Программа модуля предполагает знакомство с основными понятиями, используемыми в языках программирования высокого уровня, решение большого количества творческих задач, многие из которых моделируют процессы и явления из таких предметных областей, как информатика, алгебра, геометрия, география, физика, русский язык и др. Многие задания составлены таким образом, чтобы они решались методами учебно-исследовательской и проектной деятельности. Большинство заданий встречаются в разных темах для того, чтобы показать возможности решения одной и той же задачи или проблемы различными средствами, обеспечивающими достижение требуемого результата, что в итоге приведет к способности выбирать оптимальное решение данной задачи или проблемы.

### **Целевые установки для класса:**

- получить представление об алгоритмах и исполнителях, основных алгоритмических конструкциях языков программирования;
- обеспечить высокую мотивацию к проектной деятельности и дальнейшему изучению языков программирования;
- формировать компьютерную грамотность;
- формирование и развитие универсальных учебных действий;
- использовании других программ (например, LibreOfficeDraw) для создания собственных изображений;
- использовании метода проектов для моделирования объектов и систем;
- возможности описания реальных задач средствами программной среды;
- создании анимационных, игровых, обучающих проектов, а также систем тестирования в программной среде Scratch.

### **3. Описание места учебного предмета в учебном плане (на уровень обучения)**

В учебном (образовательном) плане ЧОУ Многопрофильная гимназия им. М.Г. Гамзатова на изучение элективного курса информатики «**Практикум по программированию в среде Scratch**» для 5-6 классов основной школы отводится 1 ч в неделю, 34 уроков в год.

**Количество учебных недель:** 34 за один и 68 за два учебных года.

Элективный курс информатики «Практикум по программированию в среде Scratch» для 5-6 классов содержит 12 практических работ (курс является практико-ориентированным, поэтому каждое занятие содержит как теоретический, так и практический материал.) и 8 контрольных работ за два учебных года.

#### 4. Описание ценностных ориентиров содержания элективного курса

В основу представляемого элективного курса «Практикум по программированию в среде Scratch» для 5-6 классов положены такие принципы как:

- Целостность и непрерывность, означающие, что данный курс является важным звеном единой общешкольной подготовки по информатике и информационным технологиям.
- Воспитательный принцип обучения через мультимедийную среду, которая позволяет формировать информационную культуру обучающихся, дает возможность приобрести навыки проектной деятельности, формирует практические навыки программирования.
- Практико-ориентированность, обеспечивающая отбор содержания, направленного на решение простейших практических задач планирования деятельности, реализующих основные пользовательские возможности информационных технологий. Знакомство детей с ней полезно в плане будущей профессиональной ориентации.
- Принцип сознательности и активности дополнительного образования детей направлен на формирование и развитие как творческих, так и технических способностей детей, на организацию их свободного времени
- Принцип наглядности имеет значительный педагогический потенциал. Ориентированность на графику доказывает эффективность обучения с опорой на наглядно-образное мышление. В Scratch существует как растровый, так и векторный редактор, инструменты, палитра, есть возможность редактирования готовых героев и их костюмов.
- Принцип доступности простоты и дружелюбности интерфейса позволяет начинать изучение программирования, как только дети научатся читать; наглядно знакомит с математическими понятиями, такими как проценты, десятичные дроби, координатная плоскость, отрицательные числа. Это интересно, доступно и понятно и как результат ребенок получает позитивный опыт по достижению своих целей и реализации проектов.
- Принцип дидактической спирали как важнейший фактор структуризации в методике обучения информатике: вначале общее знакомство с понятием с учетом имеющегося опыта обучаемых, затем его последующее развитие и обогащение, создающее предпосылки для научного обобщения в старших классах.
- Принцип прочности, когда редактор текстов как конструктор дает возможность на подсознательном уровне превратить «учёбу» в «не учёбу», он построен на интересной идее конструкторов Lego: все операторы языка и другие его элементы представлены блоками, которые могут соединяться один с другим, образуя скрипт (фрагмент кода). Важной особенностью блоков является их «специализация»: имеется несколько видов блоков и они могут составляться не произвольным образом, а лишь сообразно своему назначению. Это исключает возможность появления синтаксических ошибок.
- Объектная ориентированность позволяет изучить основные способы создания

программ с объектами. Одной из важнейших особенностей Scratch, как языка программирования, является его событийно-ориентированный характер. Это означает, что все объекты взаимодействуют при помощи обмена сообщениями. Такая схема обмена информацией делает Scratch близким к современным объектно-ориентированным языкам и позволяет впоследствии более просто организовать переход к изучению других языков программирования (Java, Delphi, C# и др.)

- Принцип развивающего обучения. Обучение ориентировано не только на получение новых знаний в области информатики и информационных технологий, но и на активизацию мыслительных процессов, формирование и развитие у школьников обобщенных способов деятельности, формирование навыков самостоятельной работы.

## **5. Личностные, метапредметные и предметные результаты освоения элективного курса**

Программа позволяет добиваться следующих результатов освоения образовательной программы основного общего образования:

Основные **личностные результаты**, формируемые в процессе освоения программы модуля «Пропедевтика программирования со Scratch» в 5–6 классах – это:

- формирование ответственного отношения к учению, способности довести до конца начатое дело на примере завершённых творческих учебных проектов;
- формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой благодаря иллюстративной среде программирования мотивации к обучению и познанию;
- развитие опыта участия в социально значимых проектах, повышение уровня самооценки, благодаря реализованным проектам;
- формирование коммуникативной компетенции в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;
- формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;
- формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;
- развитие эстетического сознания через творческую деятельность на базе иллюстрированной среды программирования.

К основным **метапредметным результатам** (осваиваемым обучающимися межпредметным понятиям и универсальным учебным действиям, способности их использования, как в учебной, так и в познавательной и социальной практике), формируемые в процессе освоения программы модуля «Пропедевтика программирования со Scratch» в 5–6 классах, можно отнести:

- умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;

- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата, понимая, что в программировании длинная программа не значит лучшая программа;
- умение оценивать правильность решения учебно-исследовательской задачи;
- умение корректировать свои действия, вносить изменения в программу и отлаживать ее в соответствии с изменяющимися условиями; – владение основами самоконтроля, принятия решений;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;
- ИКТ-компетенцию;
- умение сотрудничества и совместной деятельности со сверстниками в процессе проектной и учебно-исследовательской деятельности.

Основные **предметные результаты**, формируемые в процессе изучения модуля «Пропедевтика программирования со Scratch» в 5–6 классах направлены на:

- осознание значения математики и информатики в повседневной жизни человека;
- формирование представлений об основных предметных понятиях – «информация», «алгоритм», «модель» и их свойствах;
- развитие логических способностей и алгоритмического мышления, умений составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами – линейной, условной и циклической;
- развитие представлений о числах, числовых системах;
- овладение символьным языком алгебры, умение составлять и использовать сложные алгебраические выражения для моделирования учебных проектов, моделировать реальные ситуации на языке алгебры;
- развитие пространственных представлений, навыков геометрических построений и моделирования таких процессов, развитие изобразительных умений с помощью средств ИКТ;
- формирование информационной и алгоритмической культуры, развитие основных навыков использования компьютерных устройств и программ;
- формирование умения соблюдать нормы информационной этики и права.

#### **Контрольно-измерительные материалы:**

- Сорокина Т.Е., Босова А.Ю., под ред. Босова Л.Л. Информатика. Практикум по программированию в среде Scratch. 5-6 классы.– М.: БИНОМ. Лаборатория знаний, 2019.
- Маржи М., пер. с англ. Гескина М. и Таскаева С. Scratch для детей. Самоучитель по программированию. – М.: Манн, Иванов и Фербер, 2017.

- Пашковская Ю.В. Творческие задания в среде Scratch: рабочая тетрадь для 5-6 классов . -- М. : Лаборатория знаний, 2018. (Школа программиста)
- Путина А.С., под ред. Тарапаты В.В. Scratch 2.0 от новичка к продвинутому пользователю. Пособие для подготовки к Scratch-Олимпиаде. -- М. : Лаборатория знаний, 2019. (Школа юного программиста)
- Официальный сайт Scratch <https://scratch.mit.edu/>

### **Планируемые результаты изучения элективного курса в 5-6 классах:**

В результате освоения модуля «Пропедевтика программирования со Scratch» школьники будут уметь:

- самостоятельно устанавливать программную среду на домашний компьютер;
- изменять некоторые стандартные установки пользовательского интерфейса (например, язык отображения информации);
- использовать различные способы отладки программ, включая пошаговую отладку;
- уверенно использовать инструменты встроенного графического редактора, включая работу с фрагментами изображения и создание градиентов;
- создавать собственные изображения в других программах (например, LibreOfficeDraw) и импортировать их в программную среду Scratch;
- использовать графические примитивы векторного редактора LibreOfficeDraw для создания объектов;
- создавать изображения из пунктирных и штрихпунктирных линий с изменением цвета и толщины линии;
- упрощать программы за счёт использования циклических команд и применять их;
- составлять простые параллельные алгоритмы;
- создавать программы и игры с использованием интерактивных технологий;
- моделировать ситуации с использованием необходимых форм ветвления алгоритма, включая цикл по условию;
- передавать сообщения исполнителям для выполнения последовательности команд (включая разные типы исполнителей);
- планировать и создавать анимации по определенному сюжету;
- создавать игры, используя интерактивные возможности программной среды Scratch;
- планировать и создавать обучающие программы для иллюстрации пройденного материала других предметных областей;
- продумывать и описывать интерактивное взаимодействие для создания простейших тренажеров;
- подходить творчески к построению моделей различных объектов и систем.

### **Основные инструменты для оценивания результатов:**

- практическая работа;
- самостоятельная анимация или проект;
- онлайн-тест;
- контрольная работа;
- диагностические контрольные работы;

- БеджБорд (Dnevnik.ru), альтернативный неформальный метод оценивания поведения ученика класса;
- наблюдение;
- анализ работы в группе при проведении интерактивных деловых / ролевых игр;
- опросник;
- рефлексивные карты;
- рефлексивный дневник;
- портфолио;
- контент-анализ;
- анализ проблемных ситуаций.

### **Формы текущего контроля знаний, умений, навыков; промежуточной и итоговой аттестации учащихся**

*Текущий контроль* осуществляется с помощью устных ответов, самостоятельных работ, практических работ (компьютерного практикума).

*Тематический контроль* осуществляется по завершении крупного блока (темы) в форме интерактивного тестирования, теста по опросному листу или компьютерного тестирования.

*Итоговый контроль* осуществляется по завершении учебного материала за год в форме интерактивного тестирования, теста по опросному листу или компьютерного тестирования, творческой работы.

### **Критерии и нормы оценки**

#### **Оценка устных ответов**

**Оценка «5»** ставится в том случае, если учащийся

- правильно понимает сущность вопроса, дает точное определение и истолкование основных понятий;
- правильно анализирует условие задачи, строит алгоритм и записывает программу;
- строит ответ по собственному плану, сопровождает ответ новыми примерами, умеет применить знания в новой ситуации;
- может установить связь между изучаемым и ранее изученным материалом из курса информатики, а также с материалом, усвоенным при изучении других предметов.

**Оценка «4»** ставится, если:

- ответ ученика удовлетворяет основным требованиям к ответу на оценку 5, но дан без использования собственного плана, новых примеров, без применения знаний в новой ситуации, без использования связей с ранее изученным материалом и материалом, усвоенным при изучении других предметов;
- учащийся допустил одну ошибку или не более двух недочетов и может их исправить самостоятельно или с небольшой помощью учителя.

**Оценка «3»** ставится, если учащийся

- правильно понимает сущность вопроса, но в ответе имеются отдельные пробелы в усвоении вопросов курса информатики, не препятствующие дальнейшему усвоению программного материала;
- умеет применять полученные знания при решении простых задач по готовому алгоритму;
- допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более двух-трех негрубых ошибок, одной негрубой ошибки и трех недочетов;
- допустил четыре-пять недочетов.

**Оценка «2»** ставится, если учащийся не овладел основными знаниями и умениями в соответствии с требованиями программы и допустил больше ошибок и недочетов, чем необходимо для оценки 3.

### **Оценка практических работ**

#### **Оценка «5»**

- выполнил работу в полном объеме с соблюдением необходимой последовательности действий;
- проводит работу в условиях, обеспечивающих получение правильных результатов и выводов;
- соблюдает правила техники безопасности;
- в ответе правильно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления;
- правильно выполняет анализ ошибок.

**Оценка «4»** ставится, если выполнены требования к оценке 5, но допущены 2-3 недочета, не более одной ошибки и одного недочета.

#### **Оценка «3»** ставится, если

- работа выполнена не полностью, но объем выполненной части таков, что позволяет получить правильные результаты и выводы;
- в ходе проведения работы были допущены ошибки.

#### **Оценка «2»** ставится, если

- работа выполнена не полностью и объем выполненной работы не позволяет сделать правильных выводов;
- работа проводилась неправильно.

### **Оценка письменных контрольных или тестовых работ**

#### **Оценка 5** ставится в том случае, если учащийся

- выполнил работу в полном объеме с соблюдением необходимой последовательности действий;
- допустил не более 5% неверных ответов.

**Оценка 4** ставится, если выполнены требования к оценке 5, но допущены ошибки (не более 20% ответов от общего количества заданий).

#### **Оценка 3** ставится, если учащийся

- выполнил работу в полном объеме, неверные ответы составляют от 20% до 50% ответов от общего числа заданий;

- если работа выполнена не полностью, но объем выполненной части таков, что позволяет получить оценку.

**Оценка 2** ставится, если

- работа, выполнена полностью, но количество правильных ответов не превышает 50% от общего числа заданий
- работа выполнена не полностью и объем выполненной работы не превышает 50% от общего числа заданий.

Учитель может повысить отметку за оригинальный ответ на вопрос или оригинальное решение задачи, которые свидетельствуют о высоком математическом развитии обучающегося; за решение более сложной задачи или ответ на более сложный вопрос, предложенные обучающемуся дополнительно после выполнения им каких-либо других заданий.

#### **Оценка итогового проекта.**

**Зачтено** – проект выполнен полностью или выполнен на 50% и более.

**Не зачтено** -- проект не выполнен или выполнен менее 50%.

### **6. Содержание учебного предмета**

Структура содержания элективного курса информатики **«Практикум по программированию в среде Scratch»** для 5-6 классов основной школы определена следующими укрупнёнными тематическими блоками (разделами):

- Знакомство с программной средой Scratch
- Компьютерная графика
- Алгоритмы и исполнители
- Арифметические операторы и функции
- Движение и рисование
- Внешность и звуки
- Процедуры
- Переменные
- Проектная деятельность и моделирование процессов и систем
- Резерв

Представим более детально содержание модуля **«Практикум по программированию в среде Scratch»**.

#### **Раздел 1. Знакомство с программной средой 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. Основной персонаж как исполнитель программ. Система команд



исполнителя (СКИ). Блочная структура программы. Непосредственное управление исполнителем. 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.

## **Раздел 2. Компьютерная графика**

Компьютерная графика. Векторные и растровые графические редакторы. Встроенный растровый графический редактор. Основные инструменты графического редактора – кисточка, ластик, заливка (цветом или градиентом), рисование линий, прямоугольников, квадратов, эллипсов и окружностей, выбор фрагмента изображения и отражение его по горизонтали или вертикали, использование инструмента печать для копирования выделенной области изображения, работа с текстом. Масштаб фрагмента изображения. Палитра цветов, установка цвета переднего плана и фона, выбор цвета из изображения с помощью инструмента пипетка. Изменение центра костюма. Изменение размера костюма. Основные возможности изменения внешнего вида исполнителя: 1) использование встроенной библиотеки данных путём импорта её элемента; 2) редактирование выбранного элемента с помощью инструментов встроенного растрового графического редактора; 3) 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.

## **Раздел 3. Алгоритмы и исполнители –**

Алгоритм. Понятие алгоритма как формального описания последовательности действий исполнителя, приводящих от исходных данных к конечному результату. Схематическая запись алгоритма. Использование геометрических фигур для схематической записи алгоритма. Создание блок-схем в свободном векторном редакторе LibreOfficeDraw. Линейные алгоритмы. Основные признаки линейного алгоритма. Схематическое описание линейного алгоритма. Геометрические примитивы, используемые для описания линейного алгоритма. Программное управление исполнителем. Создание программ для перемещения исполнителя по экранному полю. Понятие поворота исполнителя в определенное направление. Прямой угол. Поворот исполнителя на прямой угол по часовой стрелке и против часовой стрелки. Создание программ для рисования линий. Изменение цвета и толщины рисуемой линии. Особенности пунктирной линии. Написание программы для исполнителя, чтобы он оставлял пунктирную линию при перемещении по экранному полю. Прямоугольник, квадрат – основные черты. Написание программ для движения исполнителя вдоль сторон квадрата, прямоугольника. Внесение изменений в программу рисования квадрата, если необходимо получить другой размер стороны квадрата. Прерывание программы. Циклические алгоритмы. Многократное повторение команд как организация цикла. Особенности использования цикла в программе. Упрощение программы путём сокращения количества команд при переходе от линейных алгоритмов к циклическим. Схематическая запись циклического алгоритма. Типы циклических алгоритмов. Основные конструкции программной среды, используемые для написания программ исполнителям с

применением циклов. Конечный цикл. Сокращение программы для исполнителя, рисующего линии, квадраты, прямоугольники при использовании цикла. Программа исполнителя для рисования нескольких однотипных геометрических фигур, например, нескольких квадратов из одной вершины, но с различным значением стороны. Конструкции программной среды спрятаться / показаться. Выполнение программы исполнителем, не показанным на поле выполнения программы. Написание и отладка программ с применением конструкции цикл в цикле. Бесконечный цикл. Повторяющаяся смена внешности исполнителя для имитации движения персонажа. Использование бесконечного цикла для создания анимации. Получение различного эффекта воспроизведения программы при изменении костюма исполнителя Scratch. Параллелизм в программной среде. Использование нескольких исполнителей. Копирование программы одного исполнителя другим. Выполнение одинаковых программ разными исполнителями с использованием различных начальных условий. Параллельное выполнение однотипных действий. Принцип суперкомпьютерных технологий. Таймер для вычисления времени выполнения программы. Уменьшение показаний таймера при использовании параллельных вычислений. Интерактивность программ. Возможность организации диалога между исполнителями. Операторы для слияния текстовых выражений. Взаимодействие исполнителей путём касания друг друга или цвета. Использование сенсоров при взаимодействии исполнителей. Задержка выполнения программы. Работа исполнителей в разных слоях изображения. Ветвление в алгоритмах. Использование ветвления при написании программ. Короткая форма. Полная форма условного оператора. Конструкции ветвления для моделирования ситуации. Цикл пока. Повторение команд исполнителя при выполнении определенного условия. Последовательное выполнение фрагментов программы разными исполнителями. Типы исполнителей программной среды Scratch. Системы команд исполнителей. Различные системы команд для разных типов исполнителей. Управление событиями. Передача сообщений исполнителям для выполнения определенной последовательности команд. Передача управления между различными типами исполнителей.

#### **Раздел 4. Арифметические операторы и функции**

Из этого раздела вы узнаете о разных блоках, с которыми работает Scratch, выучите их названия и области применения, познакомитесь с математическими операторами и функциями Scratch, создадите программы решающие различные математические задачи. Наша цель — разобраться в терминологии Scratch использует четыре типа блоков: командные, функции, триггеры и контрольные. Посмотрим на арифметические операторы и функции, поддерживаемые средой Scratch. При помощи блоков из раздела Операторы, Scratch поддерживает четыре основных арифметических действия: сложение (+), вычитание (-), умножение (\*) и деление (/). Блоки, которые используются для осуществления этих операций, называются операторами. Scratch также поддерживает оператор модуля (модуль), который показывает остаток при делении одного числа на другое. Для активного программирования есть возможность сгенерировать случайное число, особенно если создавать игры и симуляции. Scratch именно в этих целях предлагает блок выдать случайное число. Он выдает случайные числа в указанном диапазоне. Scratch также поддерживает многие математические функции. Блок квадратный корень объединяет 14 математических функций, которые можно выбрать из выпадающего меню,

в том числе квадратный корень, тригонометрические, логарифмические и экспоненциальные функции.

## **Раздел 5. Движение и рисование**

Этот раздел знакомит с командами блоков Движение и Перо, учит анимировать спрайты и передвигать их по Сцене, рисовать геометрические узоры и создавать игры, узнать, почему клонирование спрайтов — такой ценный инструмент. Если вы хотите делать игры или другие анимированные программы, чтобы перемещать спрайты по Сцене, нужно использовать блоки из раздела Движение. Более того, вам нужно будет давать спрайтам команду переместиться в конкретную точку на Сцене или повернуть в определенном направлении.

Из этой главы вы узнаете, как перемещать спрайты на конкретные позиции на Сцене при помощи команд абсолютного движения. Затем вы использовали команды относительного движения, чтобы перемещать спрайты относительно их собственного местоположения и направления. После этого вы создали несколько отличных рисунков при помощи команд Пера. Рисуя различные фигуры, вы открыли для себя возможности блока Повторить, который позволяет создавать более короткие и более эффективные скрипты. Вы также узнали о команде печать и стали использовать ее вместе с блоком повторить, чтобы легко создавать сложные узоры. В конце этой главы вы создали две игры и узнали о функции клонирования.

## **Раздел 6. Внешность и звуки**

В этой главе подробно рассматриваются блоки, которые могут быть использованы для того, чтобы добавить нашим программам ярких моментов. С помощью этих блоков мы можем ввести цвета, анимацию, графические эффекты, музыку и многое другое. Команды раздела Внешность позволят вам создавать анимацию и применять графические эффекты, такие как завихрение, рыбий глаз, призрак и пр., в работе с костюмами и фонами. Мы разобрались с блоками раздела Внешность и дали несколько примеров их использования. Вы анимировали спрайты, меняя их костюмы, узнали, как рисовать слои, и посмотрели, как они влияют на внешний вид перекрывающихся друг друга спрайтов. Затем мы обсудили команды раздела Звуки и объяснили, как проигрывать аудиофайлы, звуки барабанов и отдельные ноты. Команды из раздела Звуки пригодятся вам, если вы захотите добавить в свои программы звуки, голоса или музыку. Вы создали готовую анимированную сцену танцев при помощи команд разделов Внешность и Звуки, а закончили мощным аккордом, сделав симуляцию фейерверков.

## **Раздел 7. Процедуры**

В этом разделе объясняется, как применить к программированию принцип «разделяй и властвуй». Вместо того чтобы создавать программы целиком, вы сможете писать отдельные процедуры, которые потом сложите в единое целое. Использование процедур упростит написание, тестирование и исправление программ. Из этой главы вы узнаете, как: координировать поведение нескольких спрайтов с помощью сообщений; использовать передачу сообщений для внедрения процедур; использовать имеющуюся в Scratch функцию создай свой блок; использовать техники структурированного

программирования помогающие синхронизировать работу спрайтов. В этой главе мы будем координировать работу нескольких спрайтов, используя механизм передачи сообщений, применим доступную в Scratch функцию пользовательского блока для структурирования объемных программ, разделяя их на более мелкие, легче управляемые фрагменты, которые называются процедурами. Процедура — последовательность команд, выполняющая определенную функцию. В этой главе мы познакомились с рядом основополагающих идей, которыми будем активно пользоваться на протяжении всей учебы. Первым делом разберемся с передачей сообщений для коммуникации между спрайтами и синхронизации их работы. Затем введем понятие структурного программирования и методы использования передачи сообщений при внедрении процедур. Посмотрим, как работают пользовательские блоки в среде Scratch, и разберем, как внедрять в процедуры аргументы, чтобы сделать алгоритмы более гибкими. Приведем несколько примеров, показывающих, как крупная задача делится на более мелкие, удобные для работы части, и объясним, как использовать процедуры в качестве строительного материала при создании больших программ. Наконец, мы рассмотрим восходящую технику решения задач, при которой мы складываем уже имеющиеся решения, чтобы получить результат для масштабной задачи.

## **Раздел 8. Переменные**

В этом разделе рассказывается, как создавать скрипты, которые умеют считывать и запоминать значения. Используя переменные, можно писать приложения, которые взаимодействуют с пользователями и реагируют на введенную ими информацию. Вот темы, которые раскрываются в этой главе: какие типы данных поддерживает среда Scratch; как создавать значения и управлять ими; как получать информацию от пользователей и писать интерактивные программы. Скрипты, которые писались в предыдущих главах, помогли приобрести важные навыки программирования в Scratch, но в них отсутствовали многие ключевые элементы полномасштабного приложения. Более замысловатые программы способны запоминать значения и самостоятельно принимать решение, какие действия совершать при тех или иных условиях. В процессе выполнения скрипты обрабатывают различные типы данных и манипулируют ими. Данные могут быть введены в блоки-команды (например, число 10 вставлено в команду Идти 10 шагов, а строка «Привет!» — в команду сказать Привет!); поступать из блоков-функций (например, Мышка по x, y и Выдать случайное); вводиться пользователем в ответ на команду Спросить и Ждать. В более сложных программах часто приходится сохранять и видоизменять данные для выполнения определенных задач. Управление данными в Scratch осуществляется с помощью переменных и списков. Эта глава посвящена переменным. Глава начинается с обзора типов данных, которые поддерживает Scratch. Потом вводятся переменные и рассказывается о том, как создавать и использовать их в программах. Овладев базовыми знаниями, вы научитесь применять команду Спросить и Ждать, чтобы получать информацию от пользователя. Переменные — одна из основ программирования. Это имя области в машинной памяти, где мы можем сохранить единственную величину: число или строку. Из этой главы узнаем, какие основные типы данных поддерживает Scratch и какие операции разрешены с этими типами. Создадим несколько приложений, которые используют переменные для демонстрации тех или иных

функций. Разуберёмся с мониторами переменных и используем их для создания разных видов интерактивных программ.

## **Раздел 9. Проектная деятельность и моделирование процессов и систем**

Мультимедийный проект. Описание сюжетных событий. Анимация. Создание эффекта анимации с помощью последовательной смены изображений. Имитационные модели. Интерактивные проекты. Игры. Аналитическая деятельность: – создавать план появления событий для отражения определенной темы; – выбирать иллюстративный материал из встроеной библиотеки; – выбирать метод анимации для конкретной задачи; – планировать последовательность событий для создания эффекта анимации по выбранному сценарию. Практическая деятельность: – использовать возможности программной среды Scratch для создания мультимедийных проектов; – создавать имитационные модели, интерактивные проекты и игры средствами программной среды.

## **7. Тематическое планирование с определением основных видов учебной деятельности**

В тематическом планировании разделы основного содержания по элективному курсу разбиты на темы в хронологии их изучения по соответствующим учебникам Сорокина Т.Е., Босова А.Ю., под ред. Босова Л.Л. «Информатика. Практикум по программированию в среде Scratch» и Маржи М., пер. с англ. Гескина М. и Таскаева С. «Scratch для детей. Самоучитель по программированию».

Тематическое планирование содержит описание возможных видов деятельности учащихся в процессе усвоения соответствующего содержания, направленных на достижение поставленных целей обучения.

Тематическое планирование предназначено для 5а, классов, нацеленного на повышенный уровень подготовки учащихся по элективному курсу информатики «Пропедевтика программирования на Scratch». В основное программное содержание включены дополнительные разделы, способствующие развитию логического кругозора, освоению более продвинутого аппарата информационных технологий, алгоритмического мышления и творческого созидания. Расширение содержания образования информатики элективным курсом в этом случае даёт возможность существенно обогатить круг решаемых информационных задач и заложить основы грамотного программирования.

## **Список литературы:**

1. Банкрашков, А.В. Программирование для детей на языке Python / А.В. Банкрашков. - М.: АСТ, 2018. - 288 с.
2. Вордерман, К. Программирование на Python. Иллюстрированное руководство для детей / К. Вордерман, К. Стили, К. Квигли. - М.: Манн, Иванов и Фербер, 2017. - 346 с.
3. Воронин, И. Программирование для детей. От основ к созданию роботов / И. Воронин. - СПб.: Питер, 2018. - 192 с.
4. Воронин, И. Программирование для детей. От основ к созданию роботов / И. Воронин, В. Воронина. - СПб.: Питер, 2019. - 304 с.
5. Свейгарт, Э. Программирование для детей. Делай игры и учи язык Scratch! / Э. Свейгарт. - М.: Эксмо, 2015. - 200 с.
6. Торгашева, Ю.В. Программирование для детей. Учимся создавать игры на Scratch / Ю.В. Торгашева. - СПб.: Питер, 2017. - 608 с.
7. Торгашева, Ю.В. Программирование для детей. Учимся создавать игры на Scratch / Ю.В. Торгашева. - СПб.: Питер, 2018. - 128 с.
8. Торгашева, Ю.В. Программирование для детей. Мои первые программы на Scratch / Ю.В. Торгашева. - СПб.: Питер, 2017. - 608 с.
9. Торгашева, Ю.В. Программирование для детей. Мои первые программы на Scratch / Ю.В. Торгашева. - СПб.: Питер, 2018. - 96 с.
10. Уитни, Д. Программирование для детей. Учимся создавать сайты, приложения и игры. HTML, CSS и JavaScript / Д. Уитни. - СПб.: Питер, 2018. - 301 с.

## **9. Приложения**

### ***Темы для проектов:***

- Информатика и информация.
- Многообразие форм представления информации.
- Действия с информацией: поиск, сбор, обработка, хранение и передача информации.
- Кодирование информации.
- Метод координат как универсальный способ кодирования графической информации с помощью чисел.
- Понятие алгоритма, примеры алгоритмов.
- Исполнители алгоритмов, СКИ.
- Способы записи алгоритмов.
- Аппаратное обеспечение компьютера.
- Файл, основные операции с файлами.

- Техника безопасности и санитарно-гигиенические нормы при работе на компьютере.
- Текстовый редактор: назначение и основные функции.
- Графический редактор: назначение и основные функции.
- Калькулятор и его возможности.
- Мультимедийные технологии.
- История развития ПК.
- От телег к беспилотникам
- Создание часов с тремя стрелками.
- Анимация «Как я хожу в школу».
- Анимация «Моя семья».
- Исследовательский проект по любому учебному предмету.
- Экранизация любой сказки (стихотворения, басни и т.п.).
- Игра -- своя или аналог существующей.
- Язык программирования Scratch.
- Визуальное программирование.
- Исполнители вокруг нас.
- Алгоритмы в нашей жизни.
- Модель и моделирование.
- Компьютерная грамотность.
- Информационная безопасность.
- Цифровая культура.
- Цифровой этикет.

**Темы проектов на платформе «ГлобалЛаб»:**

1. *Ребусы – путь к развитию мышления*  
([https://globallab.org/ru/help/topic/curriculum\\_5cl.html#.Xnss-kzaCg](https://globallab.org/ru/help/topic/curriculum_5cl.html#.Xnss-kzaCg))
2. *Исследование клавиатуры*  
([https://globallab.org/ru/project/cover/issledovanie\\_klaviatury.ru.html#.XnsuHfkzaCg](https://globallab.org/ru/project/cover/issledovanie_klaviatury.ru.html#.XnsuHfkzaCg))
3. *Делаем новую Морзянку*  
([https://globallab.org/ru/project/cover/delaem\\_novuju\\_morzjanku.ru.html#.XnsuOvkzaCg](https://globallab.org/ru/project/cover/delaem_novuju_morzjanku.ru.html#.XnsuOvkzaCg))
4. *Как мы воспринимаем информацию?* (  
[https://globallab.org/ru/project/cover/kak\\_my\\_vosprinimaem\\_informaciju.ru.html#.XnsutfkzaCh](https://globallab.org/ru/project/cover/kak_my_vosprinimaem_informaciju.ru.html#.XnsutfkzaCh))
5. *Информационная безопасность*  
([https://globallab.org/ru/project/cover/informatsionnaja\\_bezopasnost.ru.html#.XnsuzkzaCh](https://globallab.org/ru/project/cover/informatsionnaja_bezopasnost.ru.html#.XnsuzkzaCh))
6. *Кредит доверия*  
([https://globallab.org/ru/project/cover/cmon\\_just\\_trust\\_me.ru.html#.XnsvA\\_kzaCh](https://globallab.org/ru/project/cover/cmon_just_trust_me.ru.html#.XnsvA_kzaCh))
7. *Компьютерная лексика и сленг*  
([https://globallab.org/ru/project/cover/kompjuternaja\\_leksika\\_i\\_sleng.ru.html#.XnsvWPkzaCh](https://globallab.org/ru/project/cover/kompjuternaja_leksika_i_sleng.ru.html#.XnsvWPkzaCh))
8. *Маленькие художники*  
([https://globallab.org/ru/project/cover/malenkie\\_hudozhniki.ru.html#.Xnsvc\\_kzaCh](https://globallab.org/ru/project/cover/malenkie_hudozhniki.ru.html#.Xnsvc_kzaCh))
9. *Дети о детях*  
([https://globallab.org/ru/project/cover/deti\\_o\\_detjakh.ru.html#.Xnsvl\\_kzaCh](https://globallab.org/ru/project/cover/deti_o_detjakh.ru.html#.Xnsvl_kzaCh))
10. *Социальные сети*  
([https://globallab.org/ru/project/cover/socialnye\\_seti.html#.Xnsv7vkzaCg](https://globallab.org/ru/project/cover/socialnye_seti.html#.Xnsv7vkzaCg))

11. *Мой любимый смартфон.*  
([https://globallab.org/ru/project/cover/ispolzovanie\\_smartfonov\\_detmi\\_v\\_povsednevnoi\\_zhizni.html#.XnswGfkzaCg](https://globallab.org/ru/project/cover/ispolzovanie_smartfonov_detmi_v_povsednevnoi_zhizni.html#.XnswGfkzaCg))
12. *Проверь свой IQ. Кодирование в информатике.*  
([https://globallab.org/ru/project/cover/informatika\\_v\\_povsednevnoi\\_zhizni.html#.XnswgPkzaCg](https://globallab.org/ru/project/cover/informatika_v_povsednevnoi_zhizni.html#.XnswgPkzaCg))
13. *Один без цифрового мира*  
([https://globallab.org/ru/project/cover/odin\\_den\\_bez\\_cifrovogo\\_mira.html#.Xnsw9fkzaCg](https://globallab.org/ru/project/cover/odin_den_bez_cifrovogo_mira.html#.Xnsw9fkzaCg))
14. *Символы где-то рядом ...*  
([https://globallab.org/ru/project/cover/simvoly\\_gde\\_to\\_rjedom.html#.XnsxPfkzaCg](https://globallab.org/ru/project/cover/simvoly_gde_to_rjedom.html#.XnsxPfkzaCg))



## Тематическое планирование

5 класс – 1 час в неделю, всего 34 часа.

Номер пункта	Содержание материала	Количество часов	Дата		Характеристика основных видов деятельности ученика (на уровне учебных действий)
			План	Факт	
1	Знакомство со средой программирования Scratch	1	<b>06.09</b>		<p>Аналитическая деятельность: – понимать иерархическую организацию библиотеки данных программной среды; – выделять путь к элементам библиотеки; – выбирать наиболее подходящий инструмент графического редактора для создания фрагмента изображения; – придумывать и создавать различные градиенты для заливки замкнутой области; – планировать создание симметричных изображений.</p> <p>Практическая деятельность: – выбирать и запускать программную среду Scratch; – работать с основными элементами пользовательского интерфейса программной среды; – выбирать необходимый режим окна; – выбирать необходимый файл из нужной папки библиотеки программы; – соблюдать требования техники безопасности при работе в компьютерном классе.</p>
2	Спрайт, цвет и размер пера.	1	<b>13.09</b>		
3	Способы смены внешнего вида (костюма).	1	<b>20.09</b>		
4	Знакомство с векторным графическим редактором LibreOffice Draw.	1	<b>27.09</b>		
5	Линейные алгоритмы. Линии.	1	<b>04.10</b>		
6	Линейные алгоритмы. Квадраты и прямоугольники.	1	<b>11.10</b>		
7	Циклические алгоритмы. Квадраты, линии.	1	<b>18.10</b>		
8	Циклические алгоритмы. Разные фигуры.	1	<b>25.10</b>		
9	Вложенные циклы. Квадрат из квадратов.	1	<b>08.11</b>		
10	Вложенные циклы. Пунктирная линия с поворотом.	1	<b>15.11</b>		
11	Анимация на основе готовых	1	<b>22.11</b>		

Номер пункта	Содержание материала	Количество часов	Дата		Характеристика основных видов деятельности ученика (на уровне учебных действий)
			План	Факт	
	костюмов.				<p>– находить параллельности в выполняемых действиях и программировать их с помощью нескольких исполнителей; – планировать последовательность событий для заданного проекта.</p> <p>Практическая деятельность: – составлять и отлаживать программный код; – использовать конструкции программной среды для создания линейных, разветвленных и циклических алгоритмов; – организовывать параллельные вычисления; – организовывать последовательность событий программы, передачу управления от одних исполнителей другим.</p> <p>Аналитическая деятельность: – создавать план появления событий для отражения определенной темы; – выбирать иллюстративный материал из встроенной библиотеки; – выбирать метод анимации для конкретной задачи; – планировать последовательность событий для создания эффекта анимации по выбранному сценарию.</p>
12	Сцена как исполнитель. Модель таймера.	1	29.11		
13	Одинаковые программы для разных исполнителей.	1	06.12		
14	Параллельное выполнение действий несколькими исполнителями.	1	13.12		
15	Планирование работы. Таймер.	1	20.12		
16	Анимация "Солнечные сутки".	1	27.12		
17	Алгоритмы с ветвлением. Условие "если".	1	17.01		
18	Циклические конструкции с условием. Шарики в лабиринте.	1	24.01		
19	Случайные числа. Анимация "Случайные перемещения".	1	31.01		
20	Знакомство со слоями. Анимация "Самолет сквозь облака".	1	07.02		
21	Работа со слоями. Дорога.	1	14.02		
22	Взаимодействие исполнителей.	1	21.02		
23	Последовательное выполнение команд исполнителями.	1	28.02		
24	Программирование клавиш.	1	07.03		

Номер пункта	Содержание материала	Количество часов	Дата		Характеристика основных видов деятельности ученика (на уровне учебных действий)
			План	Факт	
	Создание игры.				<p>Практическая деятельность: – использовать возможности программной среды Scratch для создания мультимедийных проектов; – создавать имитационные модели, интерактивные проекты и игры средствами программной среды.</p>
25	Управление событиями. Игра "Лабиринт"	1	14.03		
26	Координатная плоскость. Геометрические фигуры	1	21.03		
27	Координатная плоскость. Переменные	1	04.04		
28	Создание модели "Координаты"	1	11.04		
29	Создание списков	1	18.04		
30	Периметр прямоугольника	1	25.04		
31	Площадь прямоугольника	1	02.05		
32	Использование подпрограмм	1	16.05		
33	Мини-проект "Многоугольники"	1	23.05		
34	Мини-проект "Палиндромы"	1	30.05		



Всего прошито, пронумеровано и  
скреплено печатью

\_\_\_\_\_ ЛИСТОВ

Директор ГБОУ «Средняя школа  
№ 14 г. Иловайск»

\_\_\_\_\_ О. М. Камлева