

СОГЛАСОВАНО
Заместитель
директора по УВР

_____ **Конорева В. В.**

РАССМОТРЕНО
на заседании
методического
объединения
28 августа 2024 г.

УТВЕРЖДЕНО
И.о. директора ОУ

_____ **Гребенькова Н.В.**
(приказ от «30»августа
2024 г. №70/1)

**Отдел образования, опеки и попечительства Администрации
Золотухинского района Курской области
Муниципальное казенное общеобразовательное учреждение «Новоспасская средняя
общеобразовательная школа» Золотухинского района Курской области
(МКОУ «Новоспасская средняя общеобразовательная школа» Золотухинского
района Курской области»)**

РАБОЧАЯ ПРОГРАММА

Внеурочной деятельности по информатике

для обучающихся 6 классов

«Программируем и исследуем в Scratch»

2024–2025 учебный год

Пояснительная записка

Современный уровень развития информационных технологий требует наличия специалистов, обладающих развитым алгоритмическим мышлением, умеющих системно мыслить, обладающих навыками проектной и исследовательской деятельности, способных работать в команде. Развитие указанных способностей в период обучения на уровне основного общего образования может быть достигнуто за счёт включения в образовательную программу внеурочной деятельности, позволяющей реализовать требования ФГОС.

Потенциальным ресурсом для развития компетенций специалиста будущего является пропедевтика программирования в рамках курса «Программируем и исследуем в Scratch». Раннее обучение программированию младших школьников может не только способствовать общеинтеллектуальному развитию обучающегося, но и положить начало индивидуальной образовательной траектории школьника при выборе своего профессионального будущего в области информационных технологий.

Кроме того, пропедевтику программирования в 5-6 классах общеобразовательной школы можно рассматривать в качестве начала подготовки обучающихся к результативному участию в олимпиадах по информатике, которые, начиная с 7-8 класса, содержат задачи по программированию.

Курс разработан таким образом, чтобы наряду с развитием технологических навыков в области ИКТ происходило развитие креативных способностей обучающихся. Такой подход соответствует идеологии STEAM-образованию.

Курс «Программируем и исследуем в Scratch» *нацелен* на:

- **развитие** представлений об алгоритмических конструкциях, образного, алгоритмического и системного мышления, творческого подхода к решению задач;

- **воспитание** интереса к программированию, стремлению использовать полученные навыки для создания образовательных проектов, применимых в других предметных областях;

- **формирование** навыков исследовательской и проектной деятельности, самостоятельного выявления проблемы, поиска решения при ограниченных ресурсах, оптимизации ранее полученного решения при расширении спектра доступных ресурсов, умений составлять алгоритм для реализации проекта, собирать, тестировать и отлаживать программу из блоков программной среды.

Для достижения комплекса поставленных целей в процессе изучения курса необходимо решить следующие *задачи*:

- сформировать представление об алгоритмических конструкциях, используемых в языках программирования;

- научить создавать программы из блоков программной среды Scratch;

- привить навыки информационного моделирования в программной среде Scratch;

- воспитать интерес к программированию через создание мини-проектов;

- сформировать первичные представления об этапах проектной деятельности;

- сформировать исследовательский подход к решению поставленной задачи;

- развить интеллектуальные, творческие и познавательные способности обучающихся;

- создать условия для реализации коммуникаций при коллективном проектировании в команде сверстников.

Место курса в учебном плане

Курс «Программируем и исследуем в Scratch» расширяет и дополняет раздел курса информатики «Алгоритмика» и ориентирован на учащихся 5-6

классов общеобразовательной школы, рассчитан на 34 часа при режиме занятий один час в неделю. Он может рассматриваться как часть курса информатики в 5-м или 6-м классах. Кроме того, он может быть реализован как самостоятельный курс в рамках внеурочной деятельности учащихся 5-6 классов.

Личностные, метапредметные и предметные результаты освоения курса

Основные *личностные результаты*, формируемые в процессе освоения программы курса:

- 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 мотивации к обучению и познанию благодаря среде программирования Scratch;
- повышение уровня самооценки благодаря реализованным проектам;
- формирование коммуникативной компетентности 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 в среде блочного программирования Scratch.

Основные *метапредметные результаты*, формируемые в процессе освоения программы курса:

- умение самостоятельно ставить и формулировать новые для себя задачи, развивать мотивы своей познавательной деятельности;
- умение самостоятельно планировать пути решения поставленной проблемы для получения требуемого результата;
- умение оценивать правильность решения учебно-познавательной задачи;
- умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;
- владение основами самоконтроля, принятия решений;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы при выполнении учебно-исследовательских и проектных работ;
- владение основами ИКТ;
- умение сотрудничать и вести совместную деятельность со сверстниками в процессе проектной и учебно-исследовательской деятельности.

Основные *предметные результаты*, формируемые в процессе изучения программы курса:

- осознание значения математики, информатики и ИКТ в повседневной жизни человека;
- формирование представлений об основных предметных понятиях – «информация», «алгоритм», «исполнитель», «модель» и их свойствах;
- развитие логических способностей и алгоритмического мышления, умений составить и записать алгоритм для конкретного формального исполнителя, знакомство с основными алгоритмическими структурами – линейной, ветвление 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 образных и пространственных представлений, навыков геометрических построений и моделирования процессов, развитие изобразительных умений с помощью средств ИКТ;
- формирование информационной и алгоритмической культуры, развитие основных навыков использования компьютерных устройств и программ;
- формирование представлений о нормах информационной этики и права, умения их соблюдать.

Основное содержание учебного курса

1. Знакомство с программной средой Scratch

Свободное программное обеспечение. Основные элементы пользовательского интерфейса программной среды Scratch. Функциональные блоки. Блоки команд, состояний, программ, запуска, действий и исполнителей. Установка русского языка для Scratch. Создание и сохранение документа. Понятия спрайта, сцены, скрипта. Основной персонаж как исполнитель программ. Система команд исполнителя (СКИ). Блочная структура программы. Библиотеки костюмов и фонов. Иерархия в организации хранения костюмов персонажа и фонов для сцен. Импорт костюма, импорт фона.

Аналитическая деятельность: выделять аппаратное и программное обеспечение компьютера; определять технические устройства для ввода и вывода информации; понимать иерархическую организацию библиотеки данных программной среды; выделять путь к элементам библиотеки; выделять фрагменты изображения для дальнейшей работы с ними; планировать работу по созданию сложных изображений путём копирования и масштабирования простых.

Практическая деятельность: 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; создавать, копировать, переименовывать, перемещать, копировать и удалять файлы; соблюдать требования техники безопасности при работе в компьютерном классе.

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, изображение 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) импортирование костюма из встроенной библиотеки; 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.

Аналитическая деятельность: выделять фрагменты изображения для дальнейшей работы с ними; планировать работу по созданию сложных изображений путём копирования и масштабирования простых; выбирать наиболее подходящий инструмент графического редактора для создания

фрагмента изображения; различать верхний и нижний цвета изображения; придумывать и создавать различные градиенты для заливки замкнутой области; планировать создание симметричных изображений.

Практическая деятельность: использовать простейшие растровые и векторные редакторы для создания и редактирования изображений; изменять центр изображения; вносить изменения в изображения из встроенной библиотеки; создавать сложные графические объекты путём копирования и модификации простых объектов и их фрагментов; использовать возможности работы с цветом.

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. Программное управление исполнителем. Создание программ для перемещения исполнителя в области экрана. Поворот исполнителя в направлении, на угол. 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. Создание программы рисования пунктирной линии при перемещении исполнителя в области экрана. Прямоугольник, квадрат – основные черты. Разработка и создание программ для движения исполнителя вдоль сторон квадрата, прямоугольника. Исследование возможностей программы рисования квадрата при изменении параметров команд. Прерывание программы.

Циклические алгоритмы. Многократное повторение команд как организация циклического алгоритма. Особенности использования циклических конструкций в программе. Исследование и анализ программ при

решении задач с помощью линейных и циклических алгоритмов. Блок-схема циклического алгоритма. Виды циклических конструкций программной среды. Цикл со счётчиком. Исследование программ для исполнителя, рисующего линии, квадраты, прямоугольники при использовании линейных и циклических алгоритмов. Исследование программ исполнителя для рисования нескольких однотипных геометрических фигур, например нескольких квадратов с различным значением сторон, нарисованных из одной вершины. Видимость исполнителя – конструкции программной среды спрятаться/показаться. Создание и отладка программ с применением вложенных циклов. Циклическая конструкция всегда. Повторение смены костюма исполнителя для имитации движения персонажа. Использование циклических конструкций для создания анимации.

Параллелизм в программной среде. Использование нескольких исполнителей. Копирование программы одного исполнителя другим. Исследования выполнения разными исполнителями одинаковых программ при различных начальных условиях. Параллельное выполнение однотипных действий. Использование встроенного таймера для исследования времени выполнения программы. Сравнение и анализ показаний таймера при использовании параллельных действий исполнителей. Интерактивность программ. Возможность организации диалога между исполнителями. Операторы для слияния текстовых выражений. Организация взаимодействия исполнителей с использованием сенсоров касания цвета, спрайта, мыши. Знакомство со слоями. Разработка и создание программ для исполнителей, действующих в разных слоях.

Ветвление в алгоритмах. Использование ветвления при создании программ. Блок-схема разветвляющихся алгоритмов. Короткая форма. Полная форма. Использование разветвляющихся алгоритмов для моделирования ситуации. Цикл пока. Повторение команд исполнителя при выполнении определённого условия.

Последовательное выполнение фрагментов программы разными исполнителями. Типы исполнителей программной среды Scratch. Системы команд исполнителей (СКИ). Различные системы команд для разных типов исполнителей. Управление событиями. Передача сообщений исполнителям для выполнения определённой последовательности команд. Передача управления между различными типами исполнителей.

Аналитическая деятельность: придумывать задачи для исполнителей программной среды; выделять ситуации, для описания которых можно использовать линейные алгоритмы, алгоритмы с ветвлениями, повторениями; определять эффективный способ решения поставленной задачи; находить параллельности в выполняемых действиях и программировать их с использованием нескольких исполнителей; планировать последовательность событий для заданного проекта.

Практическая деятельность: составлять и отлаживать программный код; использовать конструкции программной среды для создания линейных, разветвлённых и циклических алгоритмов; организовывать параллельные процессы; организовывать последовательность событий программы, передачу управления от одних исполнителей другим.

4. Проектная деятельность и моделирование процессов и систем

Планирование проекта. Разработка плана взаимодействий исполнителей. 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 определённой темы; выбирать иллюстративный материал из встроенной 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.

Тематическое планирование

№ п/п	Название темы	Всего часов
1	Знакомство с программной средой Scratch	2
	Техника безопасности и правила поведения в компьютерном классе. Введение. Знакомство со средой Scratch. Внешний вид среды, поля. Анимация. Спрайт, цвет и размер пера	
2	Компьютерная графика	2
	Способы смены внешнего вида (костюма). Основные инструменты встроенного растрового графического редактора. Основные графические примитивы векторного редактора LibreOfficeDraw	
3	Алгоритмы и исполнители	14
	Линейный алгоритм. Блок-схема линейного алгоритма. Рисование линий. Использование режимов пера (опустить, поднять). Рисование квадратов, прямоугольников. Исполнитель Scratch рисует несколько линий и фигур. Копирование фрагментов программы. Циклический алгоритм. Квадраты, линии, другие фигуры. Блок-схема циклического алгоритма. Исследование возможностей программ. Вложенные циклы. Создание пунктирных линий. Фигуры из пунктирных линий. Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов. Сцена как исполнитель. Модель таймера. Использование одной программы для исполнителя с разными костюмами. Параллельное выполнение действий несколькими исполнителями. Планирование работы. Таймер. Исследование показаний таймера при параллельных вычислениях. Планирование и разработка анимации «Солнечные сутки». Алгоритмы с ветвлением. Условие «если». Полная и короткая формы команды «если». Блок-схема разветвляющегося алгоритма. Циклические конструкции с ветвлением. Шарики в лабиринте	
4	Проектная деятельность и моделирование процессов и систем	12
	Случайные числа. Исполнитель определяет цвета. Цикл при условии. Знакомство со слоями. Исполнители в разных слоях. Мини-проект «Самолёт сквозь облака». Перемещение исполнителя из одного слоя в другой.	

	<p>Действия исполнителей в разных слоях. Мини-проект «Дорога». Алгоритмы с ветвлением. Взаимодействие исполнителей. События. Планирование событий в проекте для передачи управления разным исполнителям. Последовательное выполнение команд исполнителями. Алгоритмы с ветвлением. Программирование клавиш. Использование сенсора касания цвета. Интерактивность исполнителей. Создание мини-проекта «Лабиринт». Усложнение игры «Лабиринт», создание нескольких уровней.</p> <p>Моделирование ситуации. Пешеходный переход. Водолей. Исследование учебных моделей «Рисующий карандаш», «Затухание». Создание моделей. Комнатные растения. Планирование и разработка обучающего проекта по маршрутам географических открытий</p>	
5	Итоговая проектная работа	3
	Разработка образовательного мультимедийного проекта по материалам, изученного за год. Организация защиты ученических проектов. Рефлексия	
	Резерв учебного времени	1

Ниже представлено рекомендуемое почасовое планирование занятий курса «Программируем и исследуем в Scratch».

Номер урока	Тема урока
1. Знакомство с программной средой Scratch	
1	Техника безопасности и правила поведения в компьютерном классе. Введение. Знакомство со средой программирования Scratch
2	Спрайт, цвет и размер пера
2. Компьютерная графика	
3	Способы смены внешнего вида (костюма)
4	Знакомство с векторным графическим редактором LibreOfficeDraw
3. Алгоритмы и исполнители	
5	Линейный алгоритм. Линии
6	Линейный алгоритм. Квадраты и прямоугольники
7	Циклические алгоритмы. Квадраты, линии
8	Циклические алгоритмы. Разные фигуры
9	Вложенные циклы. Квадрат из квадратов
10	Вложенные циклы. Квадрат из квадратов
11	Анимация на основе готовых костюмов
12	Сцена как исполнитель. Создаем модель таймера
13	Одинаковые программы для разных исполнителей
14	Параллельное выполнение действий несколькими исполнителями

15	Планирование работы. Таймер
16	Анимация «Солнечные сутки»
17	Алгоритмы с ветвлением. Условие «если»
18	Циклические конструкции с ветвлением. Шарик в лабиринте
4. Проектная деятельность и моделирование процессов и систем	
19	Случайные числа. Исполнитель определяет цвета
20	Знакомство со слоями. Самолет определяет цвета
21	Работа со слоями. Дорога
22	Взаимодействие исполнителей
23	Последовательное выполнение команд исполнителями
24	Алгоритмы с ветвлением. Программирование клавиш
25	Алгоритмы с ветвлением. Сенсоры касания: цвета, спрайта, мыши
26	Интерактивность исполнителей. Создание мини-проекта «Лабиринт»
27	Игра «Лабиринт». Усложнение. Разработка многоуровневой игры
28	Моделирование ситуаций. Пешеходный переход
29	Моделирование ситуации. Водолей. Исследование учебных моделей «Рисующий карандаш», «Затухание»
30	Обучающий проект по маршрутам географических открытий
5. Итоговая проектная работа	
31-32	Творческий образовательный проект
33	Защита творческого проекта
34	Резерв

Учебно-методическое и материально-техническое обеспечение

Учебно-методическое обеспечение

1. Босова Л. Л., Босова А. Ю. Информатика. 5 класс. Учебник. – М.: БИНОМ. Лаборатория знаний.
2. Босова Л. Л., Босова А. Ю. Информатика. 6 класс. Учебник. – М.: БИНОМ. Лаборатория знаний.
3. Практикум по программированию в среде Scratch /Т. Е. Сорокина, А. Ю. Босова; под ред. Л.Л. Босовой. – М.: БИНОМ. Лаборатория знаний.
4. Образовательный канал «Education» (<https://www.youtube.com/playlist?list=PLMIhDclNR1GsZ9CJBZESbm7k3Xpr7awy>)
5. Материалы авторской мастерской Босовой Л. Л. (<http://lbz.ru/metodist/authors/informatika/3>)

Программное обеспечение

Программная среда Scratch (<https://scratch.mit.edu/>), векторный графический редактор LibreOfficeDraw (<https://www.libreoffice.org/>)

Оборудование

Рабочие места учеников с персональными компьютерами с установленным программным обеспечением, выходом в Интернет. Мультимедийный проектор или МЭШ-панель.

Рекомендации по организации учебного процесса

Методы и формы решения поставленных задач

Учитывая требования ФГОС и возрастные психологические особенности младших школьников, для поддержания концентрации внимания можно рекомендовать многократную смену деятельности обучающихся (в рамках одного занятия).

Выделим некоторые основные этапы занятия, которые могут способствовать достижению лучших результатов образовательной деятельности:

- 1) организационный момент;
- 2) актуализация изученного ранее материала (фронтальный опрос, небольшие тесты по функциональному назначению программной среды Scratch, групповая работа по созданию небольших скриптов на интерактивной доске (или МЭШ-панели в программной среде Scratch, короткие вопросы на развитие внимания, небольшие активности с использованием онлайн сервиса для тестирования с моментальным анализом результатов Plickers, проведение тренингов с использованием приложения LearningApps);
- 3) объяснение нового материала или фронтальная работа по решению новых задач, разработке алгоритмов ит. д., сопровождаемая, как правило, компьютерной презентацией. На этом этапе учитель в доступной форме объясняет материал, используя при этом как традиционные наглядные пособия, так и электронные ресурсы, включая видеоуроки, материалы библиотек МЭШ, РЭШ. Учитель постепенно вводит новые понятия,

организует совместную работу обучающихся, при необходимости переходящую в игру или в дискуссию. При составлении простых алгоритмов для формального исполнителя хорошо зарекомендовала себя практика, когда один из обучающихся, выразивший желание стать формальным исполнителем, выполняет команды, формируемые другими участниками группы. В процессе такой игры происходит развитие пространственного восприятия младших школьников. Решение ключевых задач обсуждается подробно, после чего ученикам предлагается одна или несколько задач для самостоятельного решения на компьютере. При самостоятельной работе младших школьников хорошие результаты дает использование игровых практик, например использование соревновательного момента, рейтингование, присуждение более успешным школьникам различных статусов (например «консультант», «эксперт» и др.).

Дифференцированный подход в обучении по материалам курса можно осуществить благодаря подбору разноуровневых по сложности задач для обучающихся разной степени подготовленности;

4) на заключительном этапе можно предложить ученикам выбрать режим презентации в программной среде Scratch и показать свои работы другим обучающимся в группе. При публичном обсуждении полученных результатов можно найти в каждой работе наиболее удачные технические или творческие решения, что создаст ситуацию успеха для каждого ученика и может способствовать дальнейшей мотивации к учению;

5) подведение итогов занятия.

Использование ИКТ является обязательным условием организации курса, что, в свою очередь, способствует формированию и развитию навыков информационной деятельности. Предполагается использование встроенного графического редактора программной среды Scratch и векторного редактора гам свободного офисного пакета LibreOffice.

Большая часть решаемых учениками задач представлена в форме мини-проектов. Организация индивидуальной, групповой, парной работы

обучающихся над созданием мини-проектов предполагает постепенное формирование умений планировать работу как свою, так и исполнителей программной среды, что способствует формированию компетенций в области проектирования.

Формы контроля и возможные варианты его проведения

Для усиления мотивации к учению в рамках дополнительных занятий целесообразно смещение акцента в оценивании на поощрение достигнутых обучающимися результатов, что способствует реализации личностно-ориентированного подхода в обучении и формированию индивидуальной траектории обучения.

В качестве форм промежуточного контроля могут использоваться проводимые в игровой форме короткие тесты с использованием онлайн инструмента Plickers, оценка мини-проектов, создаваемых учениками.

Для тематического контроля могут быть использованы мини-проекты, в которых используются алгоритмические конструкции, средства программной среды Scratch по пройденной теме.

Предполагается, что все созданные за период обучения работы сохраняются в указанной учителем папке и доступны ученику в конце года. Они могут быть использованы для создания итоговой проектной работы на выбранную обучающимся тему. Организация ученической конференции мини-проектных и исследовательских работ является удачной формой итогового контроля и хорошим способом подведения итогов. Лучшие работы могут быть рекомендованы к участию в олимпиадах и городских конкурсах ученических проектных и исследовательских работ (например, в Международной Scratch - олимпиаде).