

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

по выполнению внеаудиторной работы при реализации программы дополнительного
общеобразовательного образования

«Программирование на Scratch»

ВВЕДЕНИЕ

Настоящие методические рекомендации разработаны с целью систематизации работы по формированию и развитию ключевых компетенций в области программирования на платформе Scratch. Документ представляет собой комплексное руководство, объединяющее теоретические основы, практические рекомендации и конкретные примеры организации различных видов учебной деятельности.

Цель документа

Основная цель - создать рекомендации для эффективного освоения курса визуального программирования в среде Scratch. Рекомендации помогают преподавателям организовать текущий контроль, а обучающимся понять, чего от них ожидают на каждом этапе.

Рекомендации решают следующие задачи:

- дать преподавателям чёткие ориентиры для проведения тестирования и оценки практических работ на каждом из занятий по программированию;
- объяснить обучающимся требования к каждому виду учебной деятельности простым и понятным языком;
- стандартизировать критерии оценивания: тест, практическая работа в Scratch, итоговый проект;
- поддержать мотивацию и уверенность юных программистов через чёткие, достижимые результаты;
- обеспечить безопасную и комфортную образовательную среду при работе в среде программирования Scratch.

Структура документа

Рекомендации включают два раздела, соответствующих трём ключевым видам учебной деятельности курса программирования:

- тестирование — проверка теоретических знаний по темам курса Scratch;
- практическая работа в среде Scratch — выполнение конкретного задания по программированию;
- заключение и рекомендации.

ТЕСТИРОВАНИЕ

Рекомендации для преподавателя

Цели проведения тестирования:

- проверить, что ребёнок усвоил ключевые понятия занятия: спрайт, скрипт, блоки управления, событийная модель Scratch;
- выявить непонимание темы до перехода к следующему занятию;
- поддержать интерес через короткие, нестрашные задания с картинками и примерами из проектов Scratch;
- создать положительный опыт проверки знаний — тест не должен пугать, он должен помогать.

Планируемые результаты:

- обучающийся называет основные части интерфейса Scratch: сцена, список спрайтов, область блоков, область скриптов;
- различает виды блоков в Scratch: движение, внешность, звук, события, управление, сенсоры, операторы, переменные;
- знает назначение ключевых конструкций: цикл — повторение действий, условие — ветвление, переменная — хранение данных, клонирования, переменных — хранение значений;
- объясняет принцип событийного программирования в Scratch: программа запускается при наступлении события;
- понимает разницу между спрайтом и его костюмом, знает как поменять фон сцены;
- умеет находить и исправлять типичные ошибки в блочном коде Scratch.

Методические рекомендации:

- используйте тесты в игровой форме: «Собери правильный скрипт», «Какой блок пропущен?», карточки с блоками;
- добавляйте скриншоты из Scratch — дети лучше воспринимают визуальные вопросы: «Что сделает этот скрипт?»;
- ограничивайте время: 8–12 минут на тест, не больше — внимание у детей короткое;
- разбирайте ошибки сразу и весело: «Почти! Давай вместе вспомним, как правильно».

Типы заданий для теста:

- выбор одного правильного ответа из 4 вариантов;
- соответствие: блок Scratch — его действие;
- «верно/неверно» — утверждения о работе блоков и конструкций;
- визуальное задание: назови блок или конструкцию на скриншоте;
- ситуационное: «Какой блок нужно добавить, чтобы спрайт остановился?».

Рекомендации для обучающихся

Как подготовиться к тесту:

- вспомни, что делали на прошлом уроке — расскажи об этом маме или другу;
- повтори основные слова: как называются блоки Scratch, что делает каждый из них;
- если что-то забыл — попроси преподавателя объяснить ещё раз до начала теста;

- не волнуйся: тест — это не экзамен, это просто способ проверить, что ты запомнил.

Как выполнять тест:

- читай каждый вопрос внимательно — до самого конца;
- если не знаешь ответ — подумай, какой вариант неправильный, и исключи его;
- отвечай на лёгкие вопросы сначала, потом возвращайся к сложным;
- если совсем не знаешь — попробуй угадать: иногда интуиция помогает!
- проверь все ответы перед тем, как сдать работу.

Полезные подсказки:

- спрайт = персонаж программы. Скрипт = набор блоков для спрайта. Сцена = экран, где всё происходит;
- цикл «всегда» — повторяет блоки бесконечно. Цикл «повторить N раз» — повторяет N раз. Условие «если...то» — выполняется при выполнении условия;
- событие «когда флажок нажат» — запускает программу. Событие «нажата клавиша» — реагирует на кнопки;
- переменная — это ячейка для хранения числа или текста. Блок «изменить переменную на 1» — увеличивает её значение;
- клонирование — это когда один спрайт создаёт свои копии. Каждый клон живёт по своему скрипту!

Пример задания и образец выполнения

Темы: Интерфейс Scratch. Блоки и скрипты. Циклы, условия, переменные.

Тестовые задания:

1. Что делает блок «идти 10 шагов» в Scratch?

- a) Перемещает спрайт вправо на 10 шагов
- b) Удаляет спрайт со сцены
- c) Поворачивает спрайт на 10 градусов
- d) Увеличивает спрайт на 10%

2. Какой блок запускает скрипт при нажатии на зелёный флажок?

- a) Когда нажата клавиша
- b) Когда флажок нажат
- c) Когда получено сообщение
- d) Когда спрайт нажат

3. Сколько раз выполнятся блоки внутри цикла «повторить 5 раз»?

- a) 1 раз
- b) 5 раз
- c) 10 раз
- d) Бесконечно

4. Что такое спрайт в Scratch?

- a) Цвет фона сцены

- b) Персонаж, которым можно управлять через скрипты
- c) Набор блоков для движения
- d) Кнопка запуска программы

5. Верно ли утверждение: «Цикл "всегда" повторяет блоки бесконечно»?

- a) Верно
- b) Неверно

Образец выполнения:

- a) Клавиатура
- b) Когда флажок нажат
- c) 5 раз
- d) Специальное место для хранения файлов
- e) Верно

Критерии оценивания

Оценка	Правильных ответов	Характеристика выполнения
5	90–100%	Все основные понятия усвоены, допущена не более 1 ошибки. Отличный результат!
4	75–89%	Большинство тем усвоено, допущено 2–3 ошибки. Молодец, почти всё правильно!
3	60–74%	Больше половины верно, но некоторые темы требуют повторения
2	менее 60%	Нужно повторить материал — преподаватель поможет разобраться

Примечания:

- Тест проводится после завершения каждого раздела или по завершении темы.
- Вопросы оцениваются по сути ответа, а не точной формулировке.
- После теста рекомендуется короткое обсуждение: «Что было сложнее всего?».

ПРАКТИЧЕСКАЯ РАБОТА ЗА КОМПЬЮТЕРОМ

Рекомендации для преподавателя

Практическая работа — основной вид деятельности курса. На каждом из занятий обучающийся выполняет конкретное задание в среде Scratch: создаёт анимацию, программирует движение спрайтов, собирает игру из блоков кода. Практика закрепляет теорию и даёт ощущение реального результата — это ключевой мотиватор для юных программистов.

Цели практической работы:

- освоить практические навыки работы с конкретным видом блоков или конструкцией в Scratch;
- закрепить теоретический материал через выполнение реальной задачи;
- развить логическое и алгоритмическое мышление через составление скриптов;
- воспитать аккуратность при сборке скриптов и организации структуры проекта;
- сформировать привычку сохранять проект и делиться ссылкой через платформу Scratch.

Перечень практических работ:

- Занятие 1: зарегистрироваться на Scratch, изучить интерфейс, создать анимацию «кот говорит Привет»
- Занятие 2: скрипт движения по координатам — спрайт обходит экран по периметру прямоугольника
- Занятие 3: создать открытку — смена 3 фонов, танцующий персонаж и фоновая музыка
- Занятие 4: настроить управление стрелками, добавить блок «оттолкнуться от края»
- Занятие 5: оптимизировать код через цикл «всегда», создать анимацию летящей птицы
- Занятие 6: игра «Лабиринт» — касание стены возвращает спрайт в начало, выход = беда
- Занятие 7: игра «Догонялки» — один спрайт бежит за курсором, другой убегает при приближении
- Занятие 8: добавить переменную «Счёт» при касании и таймер на 30 секунд
- Занятие 9: игра «Сбор яблок» — фрукты появляются случайно и исчезают при касании
- Занятие 10: диалог двух героев через блоки «передать сообщение» для очередности реплик
- Занятие 11: эффект «Снегопад» — клонирование снежинки со случайными координатами
- Занятие 12: расширение «Перо» — рисование квадрата и круга по нажатию клавиш
- Занятие 13: создать блок «Прыжок» (функция) и применить его к разным спрайтам
- Занятие 14: разработка комплексной игры (часть 1) — сюжет, фоны, персонажи
- Занятие 15–16: разработка комплексной игры (часть 2) — механика, очки, итоговая презентация

Методические рекомендации:

- показывайте каждый шаг на экране — маленьким детям нужна наглядная пошаговая демонстрация;
- используйте правило «покажи — попробуй сам»: сначала демо, потом ребёнок повторяет;

- поощряйте за самостоятельные попытки, даже если результат неидеален;
- для знакомства с интерфейсом — используйте режим пошагового выполнения (жёлтая подсветка блоков);
- напоминайте о правиле сохранения: «Сохранить сейчас» после каждого важного изменения;
- заканчивайте каждое занятие «показом» работы: ребёнок гордится результатом.

Рекомендации для обучающихся

Правила работы за компьютером:

- Садись прямо: спина ровная, ноги на полу, экран на уровне глаз.
- Перетаскивай блоки аккуратно: захвати блок и присоединяй к скрипту до щелчка.
- Не ешь и не пей рядом с компьютером — это может его сломать
- Перед началом работы убедись, что знаешь, какой результат должен получиться.
- Нажимай «Сохранить сейчас» в меню Файл почаще — чтобы не потерять проект.
- Если что-то пошло не так — не паникуй, позови преподавателя.

Алгоритм выполнения практической работы:

- Послушай объяснение преподавателя и посмотри, как он делает.
- Повтори шаги самостоятельно — не торопись, делай аккуратно.
- Если забыл шаг — можно посмотреть в инструкцию или спросить.
- Сохрани готовую работу в нужную папку с правильным именем файла.
- Покажи результат преподавателю и расскажи, что ты сделал.

Полезные горячие клавиши (запомни!):

- Ctrl+S — сохранить файл.
- Ctrl+Z — отменить последнее действие (если ошибся!).
- Ctrl+C — скопировать, Ctrl+V — вставить.
- Ctrl+P — распечатать.
- Delete — удалить выделенный объект.
- Win — открыть меню Пуск.

Пример задания и образец выполнения

Тема: «Создание интерактивной игры в Scratch. Движение, условия, счёт».

Цель работы: создать в Scratch интерактивную игру, где игрок управляет спрайтом и собирает объекты.

Задание:

- Открой Scratch и создай новый проект.
- Добавь спрайт игрока (например, кот) и настрой управление стрелками клавиатуры.
- Добавь спрайт «яблоко», запрограммируй его появление в случайных координатах.
- Добавь переменную «Счёт»: при касании яблока счёт увеличивается, яблоко появляется снова.
- Сохрани проект в своём аккаунте с названием «Игра_[имя]».

Критерии оценивания

Оценка	Критерии	Характеристика выполнения
5	Отлично	Задание выполнено полностью: спрайты настроены, счёт работает, проект сохранён. Ребёнок уверенно объяснил логику программы
4	Хорошо	Задание выполнено с 1–2 незначительными недочётами в логике или оформлении скриптов
3	Удовл.	Задание выполнено частично: основные скрипты есть, но механика игры работает не полностью
2	Неудовл.	Задание не выполнено или выполнено с помощью преподавателя полностью, без самостоятельной сборки скриптов

Примечания:

- Частичная помощь преподавателя допустима — оценивается степень самостоятельности.
- Если ребёнок старался, но не успел — дайте дополнительное время или оцените частичный результат.

ЗАКЛЮЧЕНИЕ

Общие рекомендации для преподавателя:

- **Создавайте атмосферу открытия:** каждый урок — это «ого, я теперь умею это!». Маленький успех — огромная мотивация.
- **Используйте игровые форматы:** квизы, соревнования «кто быстрее сохранит файл», командные задания.
- **Темп обучения у всех разный** — не торопите тех, кто медленнее, поддерживайте тех, кто опережает.
- **Возвращайтесь к принципам алгоритмического мышления на каждом занятии** — это не занудство, это защита ребёнка.
- **Родители — ваши союзники:** давайте простые домашние задания («покажи маме свой проект в Scratch») — это закрепляет навык.
- **Фиксируйте прогресс:** папка с работами ребёнка за курс — лучший показатель роста.
- **Завершайте каждое занятие «итогами дня»:** «Сегодня ты научился...» — это формирует осознанность.

Общие рекомендации для обучающихся:

- **Практикуйся дома:** открой Scratch и доработай свой проект — добавь новый спрайт или новый уровень.
- **Экспериментируй смело:** в Scratch нельзя «сломать» проект — ты всегда можешь отменить действие или начать заново.
- **Scratch — это только начало!** Потом ты сможешь научиться программировать на настоящих языках.
- **Береги аккаунт:** не делись паролем ни с кем — в нём хранятся все твои проекты.