

Управление образования Администрации Североуральского муниципального округа
Муниципальное автономное учреждение дополнительного образования
«Центр внешкольной работы»

Принята на заседании
педагогического совета
МАУ ДО «ЦВР»

Протокол № 5 от 30.06.2026 г.



Утверждаю
Директор МАУ ДО «ЦВР»
М. В. Василевич

Приказ от № 111-о от 30.06.2026 г.

Дополнительная общеобразовательная общеразвивающая программа
технической направленности
«Scratch-программирование»

Возраст учащихся: 8-10 лет
Срок реализации: 1 год (72 часа)

Автор-составитель:
Реймер М. Д.,
педагог дополнительного
образования

г. Североуральск

Оглавление

1. Комплекс основных характеристик программы	3
1.1. Пояснительная записка.....	3
1.2. Цель и задачи программы.....	7
1.3. Содержание программы	10
2. Комплекс организационно-педагогических условий.....	15
2.1. Календарный учебный график	15
2.2. Условия реализации программы	15
2.3. Формы аттестации/ контроля и оценочные материалы	17
3. Список литературы	21
Приложение.....	24

1. Комплекс основных характеристик программы

1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Scratch-программирование» (далее – программа) технической направленности.

Scratch – это среда, которая позволяет детям создавать собственные анимированные и интерактивные истории, презентации, модели, игры и другие произведения. Программа направлена на знакомство с программированием в среде Scratch и формирование у детей навыков программирования. Программа позволяет организовать среду для самореализации и самоутверждения учащихся.

Нормативно-правовые основания для разработки и реализации Программы:

Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 № 273-ФЗ;

Приказ Министерства просвещения Российской Федерации от 27.07.2022 № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным образовательным программам;

Постановление Главного государственного санитарного врача Российской Федерации от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организации воспитания и обучения, отдыха и оздоровления детей и молодёжи;

Постановление Главного государственного санитарного врача Российской Федерации от 28.01.2021 № 2 «Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания»;

Устав МАУ ДО «ЦВР»;

Положение МАУ ДО «ЦВР» «О порядке разработки и утверждения дополнительных общеобразовательных программ» (утверждено приказом от 16.05.2025 г. № 80-о)

Актуальность раскрывается через соответствие программы следующим документам:

– Концепции развития дополнительного образования детей до 2030 года, утвержденной распоряжением Правительства Российской Федерации от 31.03.2022 № 678-р., одна из целей концепции – это создание условий для самореализации и развития талантов детей;

– Концепции развития образования на территории Свердловской области на период до 2035 года, утвержденной приказом Министерства образования и молодежной политики Свердловской области от 30.03.2018 № 162-Д, которая указывает, что стратегической целью развития системы образования в Свердловской области является «обеспечение высокого качества и доступности образования как основное условие удовлетворения общественных потребностей, создание необходимых условий для выявления и развития интеллектуальных и творческих способностей обучающихся»;

– Стратегии социально-экономического развития Североуральского городского округа на период до 2030 года, утвержденной решением Думы Североуральского городского округа от 26.12.2018 № 85.

Данные документы подтверждают необходимость создания условий для выявления и развития интеллектуальных и творческих способностей детей.

Актуальность программы состоит в том, что мультимедийная среда Scratch позволяет сформировать у детей стойкий интерес к программированию, отвечает всем современным требованиям объектно-ориентированного программирования.

Отличительные особенности программы

В ходе разработки программы были проанализированы дополнительные общеобразовательные общеразвивающие программы по Scratch программированию «Увлекательный мир программирования» (для детей 7-12 лет), автор-составитель Косых Ю. С.; «Визуальное программирование» (для детей 8-10 лет), авторы составители Нагайцева К. Н., Щербакова И. А., Спирин А. В.

Данная программа для детей 8–10 лет и отличается от других программ тем, что это фундаментальный годовой курс, где каждый раздел изучается

подробно. Кроме того, по методической модели обучения данная программа относится к проблемно-исследовательской – педагог дает задачу или проблему, разбирает базовый инструмент, а алгоритм учащийся создает сам.

Адресат программы – дети 8–10 лет, проявляющие интерес к программированию. При составлении программы учтены возрастные психологические особенности учащихся. На данном этапе ведущая деятельность для ребенка учебная. Этот возраст характеризуется тем, что происходит перестройка познавательных процессов ребенка: формируется произвольность внимания и памяти, мышление из наглядно-образного преобразуется в словесно-логическое, формируется способность к созданию умственного плана действий. По мнению В. В. Давыдова, «учебная деятельность имеет специфическое содержание – это развитые формы человеческого сознания. Предметы науки и культуры – теоретические, абстрактные и требуют особого к себе отношения. Именно теоретическое отношение к действительности – проникновение во внутреннюю сущность вещей и адекватные этому способы ориентации – важнейшая, специфическая потребность и мотив учебной деятельности. Познавательные интересы выражены у детей в весьма различной степени. Но, как правило, у младших школьников мотивы познания не занимают ведущего места». Поэтому возникновение и поддержание познавательного интереса у детей 8–10 лет связывается с игровыми и эмоциональными приемами организации занятия, приданием материалу, подлежащему усвоению, занимательности.

Наполняемость группы 5–7 человек.

Указанное количество позволяет уделить достаточное внимание каждому учащемуся и учитывает количество материально-технического оборудования.

При распределении учащихся по группам учитываются возрастные и индивидуальные особенности учащихся, а также имеющиеся у детей знания, умения и опыт конструкторской деятельности.

Условия приема

На обучение по программе принимаются учащиеся общеобразовательных школ.

Объем и срок освоения программы:

- объем программы – 72 часа;
- срок освоения программы – один год.

Особенности организации образовательного процесса**Режим занятий:**

- занятия проводятся или один раз в неделю по два академических часа, или два раза в неделю по одному академическому часу;
- общее количество часов в неделю – 2 часа;
- продолжительность одного академического часа – 45 мин.;
- перерыв между часами – 10 минут.

Форма обучения очная. При необходимости, возможна реализация программы очно-заочная с применением электронного обучения и дистанционных образовательных технологий.

Формы организации образовательного процесса: фронтальная, индивидуальная, индивидуально-групповая, групповая.

Форма реализации программы традиционной модели, которая представляет собой линейную последовательность освоения содержания в течение одного месяца в одной образовательной организации.

Уровень программы стартовый. Освоение программного материала на ознакомительном уровне предполагает получение учащимися первоначальных знаний в области технологии и физики, при помощи которых учащиеся конструируют, собирают и испытывают модели.

Перечень форм проведения занятий: беседы, интерактивные лекции, практические занятия, игровые уроки (геймификация), защита проектов.

Формы подведения итогов реализации программы: мини-выставки, интерактивные викторины, чек-листы самопроверки, квест-лабиринт, защита-презентация проектов.

1.2. Цель и задачи программы

Цель: обучение основам визуального программирования в среде Scratch через создание интерактивной анимации, игры для развития алгоритмического мышления и творческих способностей учащихся 8-10 лет.

Задачи

Обучающие:

- формировать представления об интерфейсе среды Scratch;
- обучить базовым понятиям программирования: алгоритм, условие, координатная сетка;
- сформировать навыки управления объектами;
- научить работать с событиями;
- обучить проектированию, созданию и тестированию линейной анимации, интерактивных открыток и простых игр-кликеров.

Развивающие:

- развивать логическое и алгоритмическое мышление, обучая ребенка разбивать сложную задачу на последовательность мелких шагов;
- формировать навыки планирования, умение структурировать сюжет, логику и этапы создания будущей игры или анимации;
- развивать аналитические способности и навыки отладки (поиск, локализация и исправление ошибок в коде);
- формировать базовые математические навыки, развивать пространственное воображение через практическое применение осей координат на сцене.

Воспитательные:

- воспитывать усидчивость, целеустремленность и волевые качества, помогающие доводить сборку сложного скрипта до рабочего результата;
- развивать самостоятельность и творческую активность, стимулируя поиск уникальных визуальных решений при оформлении проектов;
- формировать сетевую и цифровую культуру, воспитывать уважение к чужой интеллектуальной собственности при совместной работе в сообществе

Scratch;

- повышать учебную мотивацию, формировать устойчивый интерес к дальнейшему углубленному изучению ИТ-технологий и программирования.

Планируемые результаты

Предметные результаты

Учащийся будет знать:

- устройство интерфейса среды Scratch, назначение сцены, палитры блоков, свойства спрайтов и их костюмов;
- базовые понятия программирования: линейный, циклический и разветвляющийся алгоритмы;
- правила работы условных операторов и логику их выполнения;
- принцип работы координатной сетки в рамках двумерного пространства сцены Scratch.

Учащийся будет уметь:

- управлять объектами на сцене: программировать их перемещение, изменять внешний вид (костюмы, эффекты) и подключать звуковое сопровождение;
- настраивать запуск скриптов через различные события (нажатие на зеленый флажок, клавиши клавиатуры, клик мыши по спрайту);
- самостоятельно проектировать, создавать и тестировать собственные линейные анимации, интерактивные открытки и игры-кликеры.

Метапредметные результаты

Регулятивные универсальные учебные действия (УУД):

- умение планировать: ребенок научится определять этапы создания проекта, структурировать сюжетную линию и логику будущей игры или мультфильма;
- навыки отладки и контроля: учащийся сможет самостоятельно контролировать процесс выполнения программы, находить, локализовать и исправлять логические ошибки (баги) в коде.

Познавательные универсальные учебные действия (УУД):

- алгоритмическое мышление: ребенок научится разбивать сложную, комплексную задачу на последовательность простых мелких шагов;
- пространственное воображение и математические навыки: учащийся овладеет навыком практического применения координатной сетки для точного позиционирования объектов в пространстве.

Коммуникативные универсальные учебные действия (УУД):

- умение презентовать свой проект сверстникам, формулировать вопросы и аргументированно высказывать свое мнение в ходе плей-тестов и взаимопроверки работ.

Личностные результаты

У учащегося будут сформированы:

- усидчивость и волевые качества: готовность преодолевать трудности в процессе кодирования, доводить сборку сложного скрипта до рабочего финального результата;
- самостоятельность и творческая активность: стремление искать уникальные визуальные и технические решения, уходить от шаблонного копирования при оформлении собственных проектов;
- цифровая культура: осознанное уважение к чужой интеллектуальной собственности, соблюдение правил этичного общения при совместной работе и ремиксинге проектов в сообществе Scratch;
- мотивация к обучению: устойчивый интерес к дальнейшему изучению ИТ-технологий, технического творчества и программирования.

1.3. Содержание программы

Учебный (тематический) план

№ п/п	Название раздела/темы	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
1.	Введение в Scratch и основы анимации	14	4	10	мини-выставка
1.1.	Знакомство со средой, интерфейс, создание первого спрайта.	2	1	1	
1.2.	Линейные алгоритмы: движение и звук.	4	1	3	
1.3.	Костюмы спрайтов и покадровая анимация.	4	1	3	
1.4.	Сцена и фоны. Управление сменой декораций.	4	1	3	
2.	Управление, события и циклы	16	4	12	чек-листы самопроверки
2.1.	Циклы «всегда» и «повторить N раз». Бесконечное движение.	4	1	3	
2.2.	Управление спрайтом с клавиатуры и мышки.	4	1	3	
2.3.	Координатная сетка (X и Y) для младших школьников.	4	1	3	
2.4.	Сенсоры: касание цвета, края экрана и других спрайтов.	4	1	3	
3.	Сложная логика и основы геймдизайна	20	5	15	интерактивные викторины
3.1.	Условный оператор «если ... то». Логика выбора.	4	1	5	
3.2.	Переменные. Создание счетчика очков и таймера.	4	1	6	
3.3.	Передача сообщений (передать [сообщение]). Переход по уровням.	4	1	3	
3.4.	Случайные числа (блок «выдать случайное от...»).	4	1	3	
3.5.	Клонирование спрайтов (создание множества объектов).	4	1	3	
4.	Творческая лаборатория (Свободные проекты)	12	3	9	квест-лабиринт
4.1.	Модификация и ремиксинг готовых игр. Поиск багов.	4	1	3	
4.2.	Проектирование мини-игр в мини-группах.	8	2	6	
5.	Итоговое проектирование и защита	10	2	8	защита- презентация проектов
5.1.	Работа над индивидуальным итоговым проектом (свободная тема). 6 ч	6	2	4	
5.2.	Тестирование проектов (взаимопроверка).	2	0	2	
5.3.	Публичная защита проектов перед группой (и родителями).	2	0	2	
	Всего:	72			

Содержание учебного (тематического) плана

Раздел 1. Введение в Scratch и основы анимации

Тема 1.1. Знакомство со средой, интерфейс, создание первого спрайта

Теория: Понятие «визуальный язык программирования». Знакомство с интерфейсом Scratch: сцена, палитра блоков, область скриптов, рюкзак. Понятие «спрайт» и «костюм». Техника безопасности при работе в компьютерном классе.

Практика: Запуск Scratch, создание и сохранение первого файла. Экскурсия по библиотеке персонажей. Изменение размеров и имени стартового кота.

Тема 1.2. Линейные алгоритмы: движение и звук

Теория: Что такое алгоритм? Понятие линейного алгоритма (шаг за шагом). Палитра блоков «Движение» и «Звук». Команды идти... шагов, повернуть... градусов.

Практика: Создание мини-проекта «Танцующий кот» — составление первой линейной цепочки блоков, где персонаж перемещается по сцене и издает звуки под музыку.

Тема 1.3. Костюмы спрайтов и покадровая анимация

Теория: Как создаются мультфильмы? Понятие покадровой анимации. Команда следующий костюм. Важность блока ждать... секунд для регулировки скорости анимации. Встроенный графический редактор.

Практика: Создание анимационного мультфильма «Полет птицы» или «Бегущий человечек» с использованием циклической смены костюмов одного персонажа.

Тема 1.4. Сцена и фоны. Управление сменой декораций

Теория: Понятие «Сцена» в Scratch, ее отличие от спрайтов. Управление фонами. Команды смены фона (переключить фон на...).

Практика: Создание интерактивной сюжетной открытки, в которой при клике на объекты меняется место действия.

Раздел 2. Управление, события и циклы

Тема 2.1. Циклы «всегда» и «повторить N раз». Бесконечное движение.

Теория: Зачем нужны циклы в коде? Разница между бесконечным циклом («всегда») и циклом с заданным числом повторений. Блок если касается края, оттолкнуться.

Практика: Создание анимационного проекта «Аквариум» — программирование хаотичного непрерывного движения нескольких рыб в ограниченном пространстве сцены.

Тема 2.2. Управление спрайтом с клавиатуры и мышки

Теория: Понятие интерактивности. Палитра «События». Блоки «когда клавиша ... нажата» и «перейти в указатель мыши». Управление направлением движения («повернуться к...»).

Практика: Разработка мини-тренажера «Управляемый герой», где ребенок учится перемещать персонажа во все четыре стороны с помощью стрелок клавиатуры.

Тема 2.3. Координатная сетка (X и Y) для младших школьников

Теория: Устройство сцены Scratch как координатной плоскости. Оси X (влево-вправо) и Y (вверх-вниз). Границы сцены. Команды «перейти в x:... y:...» и «плыть... секунд в точку x:... y:...».

Практика: Настольная (безкомпьютерная) игра «Морской бой в Scratch» на ковре, затем — программирование точного перемещения объектов по координатам на ПК.

Тема 2.4. Сенсоры: касание цвета, края экрана и других спрайтов

Теория: Палитра блоков «Сенсоры». Как компьютер понимает, что объекты столкнулись? Блоки «касается [спрайта]?» и «касается цвета?».

Практика: Создание полноценной игры-кликера «Лопни шарик», где шарики случайно появляются на сцене, а игрок должен успеть кликнуть по ним мышкой для их исчезновения.

Раздел 3. Сложная логика и основы геймдизайна

Тема 3.1. Условный оператор «если ... то». Логика выбора.

Теория: Что такое условие в жизни и в коде? Конструкция «Если то...». Связка блоков управления и блоков сенсоров.

Практика: Создание первого уровня игры «Лабиринт» — программирование правила: если главный герой касается стены лабиринта (определенного цвета), он отлетает на старт.

Тема 3.2. Переменные. Создание счетчика очков и таймера.

Теория: Что такое переменная (коробка для хранения данных)? Создание и именование переменных. Команды «изменить [переменную] на 1» и «задать [переменной] значение 0».

Практика: Модернизация игры-кликера или лабиринта — добавление переменной «Очки» (счет игры) и переменной «Время» (обратный отсчет).

Тема 3.3. Передача сообщений (передать [сообщение]). Переход по уровням.

Теория: Как спрайты общаются между собой? Механизм радиовещания в Scratch. Блоки «передать [сообщение]» и «когда я получу [сообщение]».

Практика: Программирование перехода на 2-й уровень в игре «Лабиринт» (когда герой касается финиша, сцена получает сообщение и меняет карту лабиринта).

Тема 3.4. Случайные числа (блок выдать случайное от...).

Теория: Понятие случайности и непредсказуемости в играх. Палитра блоков «Операторы». Блок «выдать случайное от ... до ...».

Практика: Создание динамичной игры «Падающие метеориты» или «Лови яблоки», где предметы падают сверху вниз с разной скоростью и из случайных точек по оси X.

Тема 3.5. Клонирование спрайтов (создание множества объектов).

Теория: Зачем создавать много одинаковых спрайтов вручную, если есть клоны? Блоки «создать клон самого себя», «когда я начинаю как клон» и «удалить этот клон». Оптимизация проекта.

Практика: Разработка аркады «Космическая атака» — создание бесконечного потока нападающих звезд или кораблей при помощи технологии клонирования.

Раздел 4. Творческая лаборатория (Свободные проекты)

Тема 4.1. Модификация и ремиксинг готовых игр. Поиск багов.

Теория: Понятие открытого исходного кода. Что такое «ремикс» проекта? Понятие «баг» (ошибка в коде) и «дебаггинг» (отладка).

Практика: Групповой хакатон. Педагог раздает детям сломанные игры. Задача микро-групп — найти логические ошибки в чужих скриптах и починить игру.

Тема 4.2. Проектирование мини-игр в мини-группах.

Теория: Этапы разработки программного продукта (от идеи до релиза). Распределение ролей в команде (сценарист, художник, программист).

Практика: Совместное создание прототипа игры на свободную тему по выбору микро-группы с фиксацией идей на бумаге (раскадровка).

Раздел 5. Итоговое проектирование и защита

Тема 5.1. Работа над индивидуальным итоговым проектом на свободную тему.

Теория: Правила оформления интерфейса финального проекта (кнопка Старт, экран Победы/Проигрыша).

Практика: Самостоятельная работа учащихся за ПК. Педагог выступает в роли технического ментора (консультирует, помогает решать сложные логические тупики в коде).

Тема 5.2. Тестирование проектов (взаимопроверка).

Теория: Понятие «плей-тест». Правила экологичной критики и развивающей обратной связи (техника «гамбургера»: похвала — замечание — похвала).

Практика: Дети меняются рабочими местами, играют в проекты одноклассников, заполняют короткие чек-листы отзывов, автор проекта исправляет последние найденные ошибки.

Тема 5.3. Публичная защита проектов перед группой и родителями.

Теория: Правила публичного выступления и самопрезентации. Как рассказать о своем коде просто и интересно.

Практика: Мини-выставка/фестиваль проектов. Дети по очереди выводят свои игры на большой экран, рассказывают о задумке, самом сложном скрипте и отвечают на вопросы зрителей. Награждение сертификатами.

2. Комплекс организационно-педагогических условий

2.1. Календарный учебный график

№ п/п	Основные характеристики образовательного процесса	
1.	Количество учебных недель	36
2.	Количество учебных дней	36/72
3.	Количество часов в неделю	2
4.	Количество часов по программе	72
5.	Сроки проведения аттестации	20-25 мая
6.	Начало занятий	2 сентября
7.	Праздничные – выходные дни	4 ноября, 31 декабря- 9 января, 23 февраля, 8 марта, 1, 9 мая.
8.	Окончание обучения	26 мая

2.2. Условия реализации программы

Материально-техническое обеспечение

Учебный кабинет в соответствии с санитарно-гигиеническими нормами (включая типовую мебель).

Доска школьная (магнитно-маркерная).

Оборудование:

Мультимедийный проектор – 1 шт.

Ноутбук для педагога – 1 шт.

Персональный компьютер или ученический ноутбук – 7 шт.

Кадровое обеспечение

Программу может реализовать педагог дополнительного образования, имеющий педагогическое образование, владеющий базовыми знаниями и умениями по программированию Scratch.

Методические материалы

№ п/ п	Название раздела	Дидактико-методический материал
1.	Введение в Scratch и основы анимации	Карточки-памятки «Первые шаги» (инструкция по замене фона и добавлению персонажа); дидактический лист «Оживи аквариум».
2.	Управление, события и циклы	Справочная таблица-шпаргалка «События и циклы» в картинках; настольная карточная игра «Робот и Программист».

3.	Сложная логика и основы геймдизайна	Визуальная карта координат Сцены ($\backslash(X)$ и $\backslash(Y)$) в виде пиратской карты сокровищ для легкого понимания перемещения спрайтов; лист планирования игры («Гейм-дизайн документ» для детей) с зонами для рисования главного героя, главного врага и описания цели игры (счет, уровни).
4.	Творческая лаборатория	Дидактические карточки «Рецепты кода» — готовые шаблоны микро-скриптов; карточки «Секретная миссия» для индивидуальной работы.
5.	Итоговое проектирование и защита	Чек-лист для проверки своего проекта; бланки «Отзыв геймера»

На занятиях используются следующие **педагогические технологии**:

- технология проектного обучения;
- игровые технологии;
- информационно-коммуникационные технологии;
- здоровьесберегающие технологии.

При проведении занятий используются следующие **методы**:

- 1) метод подражания (образца);
- 2) метод «След в след» с усложнением;
- 3) словесные и наглядные методы (метод метафор и аналогий; демонстрация и визуализация);
- 4) проблемно-поисковые и исследовательские методы (метод «найди баг (ошибку)», метод свободного выбора);
- 5) методы социального взаимодействия (метод парного программирования, метод взаимного рецензирования)

Информационное обеспечение реализации программы:

1. Программное обеспечение.

- Среда программирования Scratch 3.0;
- Браузеры: Яндекс.Браузер или Mozilla Firefox;
- Программы для демонстрации материалов: для просмотра презентаций (PowerPoint) и медиоплееры для воспроизведения обучающих видеороликов.

2. Сетевые ресурсы и веб-платформы.

- Облачное хранилище: папка общего доступа (с подготовленными педагогом медиафайлами (спрайты, фоны, звуки).

– Интерактивные платформы для викторин: сервисы для проведения быстрых игровых опросов на закрепление материала.

3. Цифровые дидактические и методические материалы

- Библиотека проектов Scratch.
- Видеоинструкции (видеозаписи по сложным темам).
- Электронные карты-подсказки (файлы в формате PDF с визуальными схемами)

2.3. Формы аттестации/ контроля и оценочные материалы

Контроль успеваемости носит безоценочный характер и направлен на стимулирование успеха каждого ребенка.

1. Входной контроль (1–2 академический час)

Форма проведения: интерактивный игровой тест на бумаге или платформе (без программирования) + устное собеседование.

Пример заданий: Задачи на продолжение логических последовательностей, простые лабиринты на бумаге («проведи персонажа по стрелочкам»), проверка умения пользоваться мышкой и клавиатурой.

2. Промежуточный контроль (в течение года, после каждого раздела)

Формы проведения:

- практикумы «Почини код» (Отладка): Детям дается готовый проект, в котором допущена ошибка (например, кот не доходит до цели или счетчик очков не работает). Задача — найти и исправить баг.
- мини-проекты по чек-листу: создание небольшой игры/анимации за 1 занятие по четким техническим требованиям (например, «Создать игру, где есть 2 фона, один персонаж управляется стрелками, при касании края звук прекращается»).
- викторины (квизы): интерактивные опросы (например, через Kahoot или аналогичные платформы) на знание палитры блоков.

3. Итоговый контроль

Форма проведения: публичная защита индивидуального итогового проекта на свободную тему.

Этапы итогового контроля:

- 1) создание прототипа: самостоятельное кодирование игры/анимации по своей задумке;
- 2) плей-тест (взаимопроверка): учащиеся садятся за компьютеры друг друга, играют и оставляют отзывы по принципу: «*Что понравилось больше всего*» и «*Что можно улучшить/какой баг исправить*».
- 3) публичное выступление: ребенок демонстрирует игру на большом экране, объясняет логику главного скрипта и отвечает на вопросы группы (и приглашенных родителей).

Характеристика оценочных материалов

Вид контроля	Цель проведения	Инструментарий (формы и методы диагностики)
Входной контроль	Определение стартового уровня компьютерной грамотности, развития логического мышления и пространственного воображения.	Интерактивный игровой тест на бумаге или платформе; устное собеседование.
Промежуточный контроль	Оценка степени усвоения ключевых блоков и понятий (циклы, условия, координаты).	Формы проведения: – практикумы «Почини код» (отладка); – мини-проекты по чек-листу; – викторины (квизы).
Итоговый контроль	Оценка комплексного усвоения программы, уровня самостоятельности и творческого применения навыков	Публичная защита индивидуального итогового проекта на свободную тему.

Критерии оценивания планируемых результатов

Планируемый результат	Критерии оценивания (Показатели уровней)	Диагностический инструментарий (Формы, методы диагностики)
ПРЕДМЕТНЫЕ: ЗНАНИЯ		
1. Устройство интерфейса Scratch	- Низкий: Путает рабочие зоны, ищет Сцену и палитру блоков по подсказке. - Средний: Уверенно находит палитры, меняет костюмы и фоны. - Высокий: Бегло управляет скрытыми свойствами спрайтов (размер, направление).	- Форма: Игровая викторина на общем экране. - Метод: Устный опрос, интерактивный тест-квест («Найди элемент интерфейса»).
2. Базовые алгоритмы (линейные, циклы, ветвления)	- Низкий: Различает алгоритмы только с наводящими вопросами. - Средний: Понимает разницу между циклами и ветвлениями, применяет их в типовых задачах.	- Форма: Тематический мини-хакатон. - Метод: Анализ выполненной практической работы, настольная игра «Робот и

	• Высокий: Комбинирует вечные циклы и разветвления в одном скрипте.	Программист».
3. Условные операторы	• Низкий: Не понимает логику блока «Если... то», путает вложение блоков. • Средний: Правильно собирает конструкцию «Если... то» по образцу. • Высокий: Свободно использует «Если... то... иначе» и встраивает логические датчики.	• Форма: Проверка скриптов в рабочей среде. • Метод: Метод «Найди баг» (исправление педагогом намеренно сломанного условного оператора).
4. Координатная сетка	• Низкий: Путает оси X и Y, перемещает спрайты наугад. • Средний: Понимает знаки «плюс/минус», верно задает направление по осям. • Высокий: Точно программирует команду «перейти в X, Y», вычисляет центр Сцены.	• Форма: Индивидуальные дидактические листы. • Метод: Практическое задание «Пиратская карта» (размещение объектов по координатам).
ПРЕДМЕТНЫЕ: УМЕНИЯ		
5. Управление объектами (движение, костюмы, звук)	• Низкий: Ограничивается движением вперед, звуки и костюмы меняет хаотично. • Средний: Программирует плавный шаг со сменой костюмов и тематическим звуком. • Высокий: Синхронизирует аудиоэффекты с анимацией и графическими эффектами.	• Форма: Защита микро-проектов «Оживи аквариум». • Метод: Педагогическое наблюдение за ходом кодирования, анализ проекта.
6. Настройка событий	• Низкий: Использует только запуск по зеленому флажку. • Средний: Настраивает события клавиатуры (стрелки) и клика мыши по спрайту. • Высокий: Настраивает одновременный запуск нескольких скриптов от разных триггеров.	• Форма: Фронтальный практический срез. • Метод: Экспресс-задание на время (например: «Сделай управление кнопками за 3 минуты»).
7. Проектирование продуктов (анимации, кликеры)	• Низкий: Создает проект только путем пошагового копирования за педагогом. • Средний: Создает игру-кликер или открытку по аналогии, меняя графику. • Высокий: С нуля разрабатывает авторскую механику кликера с подсчетом очков.	• Форма: Итоговое проектирование. • Метод: Оценка готового цифрового продукта по чек-листу.
МЕТАПРЕДМЕТНЫЕ (УУД)		
8. Регулятивные УУД (планирование и дебаггинг)	• Низкий: Кодит хаотично; при ошибке бросает работу или просит все исправить. • Средний: Описывает этапы работы устно, находит ошибку с помощью подсказок. • Высокий: Ведет «Бортовой журнал», самостоятельно локализует и исправляет баги.	• Форма: Практическая самостоятельная работа. • Метод: Анализ заполнения «Бортового журнала разработчика», фиксация самостоятельных исправлений кода.

9. Познавательные УУД (алгоритмика и декомпозиция)	• Низкий: Видит задачу только целиком, не знает с чего начать. • Средний: Разбивает комплексное задание на 2-3 шага с помощью педагога. • Высокий: Самостоятельно раскладывает сложную механику на цепочку простых команд.	• Форма: Мозговой штурм. • Метод: Моделирование алгоритмов с помощью физических магнитных карточек-пазлов.
10. Коммуникативные УУД (презентация проекта)	• Низкий: Стесняется, отказывается показывать работу группе, не объясняет код. • Средний: Демонстрирует игру, кратко рассказывает правила, отвечает на вопросы. • Высокий: Уверенно выступает, аргументирует технические решения, дает советы другим.	• Форма: Публичная защита-презентация, Фестиваль творчества. • Метод: Экспертное наблюдение в процессе плей-тестов, анализ бланков «Отзыв геймера».
ЛИЧНОСТНЫЕ РЕЗУЛЬТАТЫ		
11. Личностные качества (усидчивость, творчество, культура)	• Низкий: Быстро теряет интерес, копирует чужие шаблоны, критикует чужие работы. • Средний: Доводит работу до конца, вносит небольшие творческие правки, вежлив. • Высокий: Проявляет упорство, уходит от шаблонов, помогает товарищам в паре.	• Форма: Очное сквозное наблюдение на протяжении курса. • Метод: Метод поощрения («ачивки», наклейки), педагогический анализ портфолио работ в «Студии класса».

3. Список литературы

Нормативные документы

1. Федеральный закон РФ «Об образовании в Российской Федерации» от 29.12.2012 № 273.
2. Федеральный закон Российской Федерации от 14.07.2022 № 295-ФЗ «О внесении изменений в Федеральный закон «Об образовании в Российской Федерации».
3. Федеральный закон РФ от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребенка в Российской Федерации».
4. Концепция развития дополнительного образования детей до 2030 года, утвержденная распоряжением Правительства Российской Федерации от 31.03.2022 № 678-р.
5. Указ Президента Российской Федерации от 21.07.2020 № 474 «О национальных целях развития Российской Федерации на период до 2030 года».
6. Указ Президента Российской Федерации от 09.11.2022 № 809 «Об утверждении Основ государственной политики по сохранению и укреплению традиционных российских духовно-нравственных ценностей».
7. Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».
8. Приказ Министерства труда и социальной защиты Российской Федерации от 22.09.2021 № 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»
9. Приказ Министерства просвещения Российской Федерации от 27.07.2022 № 629 «Об утверждении Порядка организации осуществления образовательной деятельности по дополнительным общеобразовательным программам».
10. Приказ Министерства просвещения Российской Федерации от 03.09.2019 № 467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей».

11. Приказ МОиПО Свердловской области от 30.03.2018 № 162-Д «Об утверждении Концепции развития образования на территории Свердловской области на период до 2035 года».

12. Приказ ГАНОУ СО «Дворец молодёжи» от 29.04.2025 № 582-д «Об утверждении методических рекомендаций».

13. Постановление Администрации Североуральского городского округа от 30.08.2023 № 1012 «Об утверждении муниципальной программы Североуральского городского округа «Развитие системы образования в Североуральском городском округе до 2024–2029 годы».

14. Устав МАУ ДО «ЦВР».

Литература, используемая при составлении программы

15. Бердитт, Рейна. Программирование с Scratch с нуля. Создаем веселые игры, охотимся за багами и пишем первые программы! /Рейна Бердитт; перевод с английского Е. В. Жевлаковой. – М.: Эксмо, 2026. – 192 с.

16. Косых, Ю. С. Дополнительная общеобразовательная общеразвивающая программа по Scratch программированию «Увлекательный мир программирования», Дзержинский Дворец детского и юношеского творчества им. Заслуженного учителя РФ Э. И. Закревской. – г. Нижний Тагил, 2025.

17. Лукьянова, Н. С. Сборник практических работ по программированию на Scratch. Учебное пособие к дополнительной программе «Лаборатория компьютерных игр», Тольятти, 2022.

18. Маржи, Мажед. Scratch для детей. Самоучитель по программированию / Мажед Маржи; пер. с англ. М. Гескиной и С. Таскаевой – М.: Манн, Иванов и Фербер, 2017. – 288 с.

19. Методические рекомендации по проектированию дополнительных общеобразовательных программ (включая разноуровневые программы) (письмо Минобрнаука России от 18.11.2015 № 09-3242).

20. Нагайцева, К. Н., Щербакова И. А., Спирин А. В. Дополнительная общеобразовательная общеразвивающая программа по Scratch программированию «Визуальное программирование». – Департамент

образования Администрации города Екатеринбурга ИП Суворова Н.В., г. Екатеринбург, 2026.

21. Пашковская, Ю. В. Программирование на Scratch для детей. Уровень 1 /Ю. В. Пашковская. – М.: Лаборатория знаний, 2024. – 224 с. : ил.

22. Положение МАУ ДО «ЦВР» «О порядке разработки и утверждения дополнительных общеобразовательных программ» (утверждено приказом от 16.05.2025 г. № 80-о).

23. Сборник диагностических методик для специалистов системы дополнительного образования детей / Сост.: В.Н. Нохрин, Н.А. Уварова – Екатеринбург: ГОУ ДООД ЦДОД «Дворец молодежи», 2005 – 68 с.

24. Смирнова Е. О. Детская психология: учебник / Е.О. Смирнова. – М.: КНОРУС, 2016. – 280 с.

25. Устав МАУ ДО «ЦВР».

Литература для учащихся и родителей

26. Голиков, Д. В. 42 проекта на Scratch 3 для юных программистов. – СПб.: БХВ-Петербург, 2026. – 184 с; ил.

27. Маржи, Мажед. Scratch для детей. Самоучитель по программированию / Мажед Маржи; пер. с англ. М. Гескиной и С. Таскаевой. – М.: Манн, Иванов и Фербер, 2017 – 288 с.

28. Пашковская, Ю. В. Программирование на Scratch для детей. Уровень 1 /Ю. В. Пашковская. – М.: Лаборатория знаний, 2024. – 224 с. : ил.

Приложение

Приложение 1

Лист входного тестирования

ФИ ребенка: _____ Дата: _____

Блок 1. Логика и алгоритмы (оценка алгоритмического мышления)

Задание 1. «Помоги Роботу дойти до батарейки»

Робот стоит в клетке А1 и смотрит вправо. Ему нужно попасть в клетку В3. Напиши буквами правильный маршрут, используя стрелочки: ↑ (вверх), ↓ (вниз), → (вправо), ← (влево). Каждая стрелочка — это шаг на 1 клетку. *Поле 3х3 клетки (А1 — левый верхний угол, В3 — правый нижний угол).*

Ответ ребенка: _____

Задание 2. «Правильный порядок»

Миша хочет заварить чай, но перепутал действия. Расставь цифры от 1 до 4 в правильном порядке:

- ✓ [] Налить кипятка в чашку
- ✓ [] Положить пакетик чая в чашку
- ✓ [] Вскипятить воду в чайнике
- ✓ [] Добавить сахар и размешать ложкой

Блок 2. Пространственное мышление и геометрия (оценка готовности к координатной сетке)

Задание 3. «Куда летит самолет?»

Посмотри на картинку (педагог показывает схему или стрелки).

- Если объект движется вверх, его высота увеличивается.
- Если объект движется вправо, он уходит вперед.

Вопрос: Если Кот сделает 5 шагов влево, а потом 3 шага вниз, где он окажется относительно центра экрана?

(Обведи правильный вариант):

- а) Выше и правее б) Ниже и левее в) Выше и левее

Блок 3. Базовая компьютерная грамотность (оценка практических навыков)

Задание 4. Напиши или скажи устно:

1. Как называется кнопка, которой мы чаще всего нажимаем на компьютерной мышке? (Левая / Правая) _____

2. Какую клавишу на клавиатуре нужно нажать, чтобы сделать пробел (отступ) между словами? _____

3. Умеешь ли ты зажимать левую кнопку мыши и перетаскивать предметы на экране? (Да / Нет) _____

Система критериев и интерпретация результатов

Каждое задание оценивается по определенной шкале. Максимальный балл за тест — 10 баллов.

Критерий / Навык	Показатели (Задания)	Баллы	Что это значит для педагога
Алгоритмическое мышление	Задание 1 (Робот): - Маршрут верный: 2 балла. - Ошибка в одном шаге: 1 балл. - Не справился: 0 баллов.	0–2 б.	Показывает, понимает ли ребенок концепцию пошагового выполнения команд персонажем.
Логическая последовательность	Задание 2 (Чай): - Порядок верный (3, 2, 1, 4): 2 балла. - Перепутаны 1-2 действия: 1 балл. - Хаос в действиях: 0 баллов.	0–2 б.	Демонстрирует способность выстраивать линейную логику событий (основа линейного кода).
Пространственная ориентация	Задание 3 (Координаты): - Верный ответ (б): 2 балла. - Ошибка в направлении: 0 баллов.	0–2 б.	Проверяет готовность ребенка к изучению осей X и Y в Модуле 2.
Компьютерные навыки и чтение	Задание 4 (Интерфейс): - Все 3 ответа верные: 2 балла. - Ошибка в кнопках/клавишах: 1 балл. - Нет навыка работы с ПК: 0 баллов.	0–2 б.	Показывает, сколько времени придется потратить на объяснение механики «Drag-and-Drop» (перетаскивание блоков).
Навык чтения инструкций	Общий темп: - Самостоятельно прочитал и выполнил: 2 балла. - Читает медленно, просит пояснить: 1 балл.	0–2 б.	В Scratch блоки текстовые. Быстро читающие дети будут двигаться по программе быстрее.

Итоги тестирования (уровни группы):

- 8–10 баллов (высокий уровень): ребенок полностью готов. Можно давать усложненные творческие задания (ремиксы, свои условия) со второго занятия.
- 5–7 баллов (средний уровень — норма): идеально вписывается в предложенный 72-часовой план. Логика развита, механику Scratch освоит быстро.
- 0–4 балла (низкий уровень): ребенку потребуется усиленная помощь на первых 10 часах. На практических занятиях педагогу нужно чаще подходить к нему, помогать физически управлять мышкой и акцентировать внимание на разборе алгоритмов в игровой бескомпьютерной форме.

Отчетный лист заполняется педагогом в середине курса и может демонстрироваться родителям для наглядной визуализации прогресса ребенка.

ОТЧЕТНЫЙ ЛИСТ УСПЕВАЕМОСТИ (ПРОМЕЖУТОЧНЫЙ КОНТРОЛЬ)

Курс: «Scratch: старт» (72 академических часа)

Период контроля: _____ (в конце каждого раздела)

ФИ учащегося: _____

Возраст / Класс: _____, _____ лет

Преподаватель: _____

1. Оценка уровня сформированности результатов (по критериям)

Педагог отмечает галочкой [X] уровень освоения каждого навыка:

Н — Начальный уровень (требуется помощь педагога, допускает частые ошибки).

Б — Базовый уровень (выполняет задания самостоятельно по образцу/инструкции).

П — Продвинутый уровень (творчески применяет навык, помогает другим, предлагает свои решения).

Контролируемый параметр / Навык	Н	Б	П	Комментарий преподавателя
Предметные результаты (Scratch & Кодинг)				
Интерфейс и графика: работа со сценой, создание/выбор спрайтов, редактирование костюмов.				
Алгоритмика (Линейная логика): составление последовательных скриптов движения и звука.				
Управление циклами: понимание и применение блоков всегда и повторить N раз.				
Координатная сетка: понимание осей X и Y, точное позиционирование персонажей на сцене.				
Работа с условиями и сенсорами: использование блоков если... то и датчиков касания.				
Метапредметные результаты (Soft Skills)				
Алгоритмическое мышление: умение декомпозировать (разбивать) сложную задачу на мелкие шаги.				
Навык отладки (Дебаггинг): самостоятельный поиск и исправление ошибок (багов) в своем коде.				
Планирование: способность удерживать в голове сюжет и структуру мини-игры перед кодированием.				
Личностные результаты (поведение и воспитание)				
Усидчивость и воля: умение концентрироваться на задаче, доводить забавный скрипт до конца.				
Самостоятельность: стремление придумать свое оформление или механику вместо копирования.				
Работа в группе и культура: уважение к чужим проектам в классе, участие в плей-тестах.				

2. Результаты выполнения практических мини-кейсов

Впишите названия созданных ребенком за отчетный период мини-проектов и статус их защиты:

1. Проект №1 «_____»
☐ Выполнен полностью
☐ Требуется доработки
☐ Защищен перед группой
2. Проект №2 «_____»
☐ Выполнен полностью
☐ Требуется доработки
☐ Защищен перед группой
3. Практикум «Почини код» (тест на отладку)
☐ Справился самостоятельно
☐ Нашел ошибки с подсказками
☐ Не справился

3. Общая характеристика и рекомендации педагога

Сильные стороны учащегося (что получается лучше всего):

Зоны роста (на что обратить внимание, что нужно подтянуть):

Рекомендации для родителей (домашние мини-челленджи, полезные упражнения):

Дата заполнения: «___» _____ 20 ____ г.

Подпись преподавателя: _____ / _____ /

**ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

СВЕДЕНИЯ О СЕРТИФИКАТЕ ЭП

Сертификат 652185396560566351996131268363309912619724340089

Владелец Василевич Марина Владимировна

Действителен с 17.11.2025 по 17.11.2026