

Управление образования администрации ЗАТО Александровск  
муниципальное бюджетное учреждение  
дополнительного образования  
«Дом детского творчества «Дриада»

СОГЛАСОВАНО

Педагогическим советом

МБУДО «ДТ «Дриада»

Протокол № 6 от 14.12.2023 г.



УТВЕРЖДАЮ:

Директор МБУДО «ДТ «Дриада»

Телегина И.Г.

«14» декабря 2023 г.

Дополнительная общеобразовательная общеразвивающая программа  
технической направленности

**«Основы алгоритмики и логики»**

Возраст обучающихся: 8-10 лет

Срок реализации: 1 год

Уровень программы: стартовый

Авторы-составители:

Березина Ольга Фёдоровна,

Тимошина Анастасия Владимировна,

педагоги дополнительного образования

ЗАТО Александровск

г.Снежногорск

2023 г.

## АННОТАЦИЯ

Большое количество времени работы по программе уделяется творческим заданиям, выполнение которых благоприятно скажется на развитии творческого потенциала учащихся. Изучение программирования происходит в среде разработки Scratch. Данный выбор обусловлен тем, что, овладев минимальным набором команд, учащиеся смогут создать законченный проект.

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

### **1. Нормативно-правовая база разработки и реализации программы.**

Данная программа составлена на основе дополнительной общеобразовательной общеразвивающей программы технической направленности «Основы алгоритмики и логики», Востриковой Е.В., Кузнецовой К.В., ЦЦОД «IT-куб», г.Мурманск, 2023 г.

Программа разработана на основе следующих нормативных документов:

- Федерального закона от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утвержденного приказом Министерства просвещения Российской Федерации от 27.07.2022 г. № 629;
- Постановления Главного государственного санитарного врача Российской Федерации 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»;
- Постановления 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»;
- Приказа Минобрнауки России №882, Минпросвещения России №391 от 05.08.2020 «Об организации и осуществлении образовательной деятельности при сетевой форме реализации образовательных программ» (вместе с «Порядком организации и осуществления образовательной деятельности при сетевой форме реализации образовательных программ»);
- Письмо Минпросвещения России от 29.03.2023 г. №АБ-1339/02 «О направлении методических рекомендаций по созданию и функционированию центров цифрового образования «IT-куб»;

– Распоряжение Минпросвещения России №Р-5 от 12.01.2021 г. «Об утверждении методических рекомендаций по созданию и функционированию центров цифрового образования ИТкуб»»;

– Письмо Министерства просвещения РФ от 10 ноября 2021 г. N ТВ-1984/04 «О направлении методических рекомендаций»;

– Стратегии развития воспитания в Российской Федерации на период до 2025 года (Утверждена распоряжением Правительства РФ от 29.05.2015 № 996- 4 р «Об утверждении Стратегии развития воспитания в Российской Федерации на период до 2025 года»);

– Концепции развития дополнительного образования детей до 2030 года, утвержденной распоряжением Правительства Российской Федерации от 31 марта 2022 г. № 678-р.

– Устава МБУДО «ДДТ «Дриада», локальных нормативных актов.

## **2. Новизна, актуальность, педагогическая целесообразность реализации программы.**

Актуальность программы обусловлена широким внедрением информационных технологий в образовательные процессы и обычную жизнь каждого человека. Данная программа способствует развитию мотивации к получению новых знаний, возникновению интереса к программированию. Позволяет школьникам попробовать себя не только в роли пользователя, но и разработчика ПО.

Новизна данной программы заключается в том, что основной акцент в освоении данной программы делается на использование проектной деятельности и самостоятельность в создании проектов, что позволяет получить полноценные и конкурентоспособные продукты. Проектная деятельность, используемая в процессе обучения, способствует развитию основных компетентностей учащегося, а также обеспечивает связь процесса обучения с практической деятельности за рамками образовательного процесса.

Педагогическая целесообразность заключается в том, что знания и умения, приобретенные в результате освоения курса, являются базой для обучения программированию. В ходе освоения программы, учащиеся получают базовые знания для освоения языков программирования высокого уровня.

Направленность программы: техническая.

## **3. Адресат программы.**

Адресатом программы являются дети в возрасте от 8 до 10 лет.

Содержание и объем стартовых знаний, необходимых для начального этапа освоения программы: умение читать и писать, решать арифметические задачи, иметь базовые навыки пользования ПК.

#### **4. Срок реализации программы.**

Срок реализации программы составляет 1 год.

Уровень программы: стартовый.

Программа предполагает использование и реализацию общедоступных и универсальных форм организации материала, минимальную сложность предлагаемого для освоения содержания программы.

#### **5. Форма реализации программы.**

Форма обучения – очная.

Образовательные технологии: информационные технологии, проектная технология, здоровьесберегающие технологии, технология проблемного обучения.

Форма организации содержания и процесса педагогической деятельности – комплексная.

Тип организации работы учеников: групповая работа, индивидуальная, коллективная.

Виды занятий: лекции и практические занятия.

Наполняемость группы: от 10 до 12 человек.

*При сетевой форме реализации программы дополнительно заключается договор о сетевом взаимодействии, в котором закрепляется правовой статус сторон и условия реализации программы.*

#### **6. Объем программы и режим работы**

Объем программы: 144 часа

Режим занятий: 2 раза в неделю по 2 академических часа.

Продолжительность часа – 40 минут.

#### **7. Цель программы.**

Целью программы является создание условий для освоения обучающимися перспективной практики программирования на языке Scratch на стартовом уровне.

#### **8. Задачи программы.**

Программа направлена на решение следующих задач:

– познакомить с правилами безопасного использования цифровыми инструментами и компьютерным оборудованием, организации рабочего места;

– сформировать умения построения различных видов алгоритмов (линейных, разветвляющихся, циклических) для решения поставленных задач;

- сформировать умения использовать инструменты среды Scratch для решения поставленных задач;
- сформировать умения построения различных алгоритмов в среде Scratch для решения поставленных задач;
- развитие алгоритмического, логического и образного мышления;
- развивать умение доводить решение задачи от идеи до работающего проекта;
- сформировать навык представления результатов своей работы окружающим, аргументировать свою позицию;
- развивать умение постановки технической задачи, собирать и изучать нужную информацию, находить конкретное решение задачи и осуществлять свой творческий замысел;
- воспитать умение работать в команде, развитие коммуникативных навыков;
- воспитать трудолюбие и уважительные отношения к интеллектуальному труду.

## **9. Планируемые результаты освоения программы**

### **В результате освоения программы учащиеся будут:**

#### **знать:**

- правила безопасного использования цифровыми инструментами и компьютерным оборудованием, организации рабочего места;
- виды алгоритмов и способы их реализации.

#### **Уметь:**

- использовать различные виды алгоритмов (линейных, разветвляющихся, циклических) для решения поставленных задач;
- составлять программы в среде Scratch для решения поставленных задач.

#### **Владеть:**

- навыком безопасного использования цифровых инструментов и компьютерного оборудования;
- навыком использования инструментов среды Scratch для решения поставленных задач.

#### **Метапредметные результаты:**

- развивать умение доводить решение задачи от идеи до работающего проекта;
- развивать умение постановки технической задачи, собирать и изучать нужную информацию, находить конкретное решение задачи и осуществлять свой творческий замысел;

– формирование умения представлять результаты своей работы окружающим, аргументировать свою позицию.

**Личностные результаты:**

- формирование уважительного отношения к интеллектуальному труду;
- развитие логического, алгоритмического и образного мышления;
- умение работать в команде, развитие коммуникативных навыков.

**10. Формы представления результатов**

**Формы аттестации:** опрос, демонстрация решения, наблюдение, защита проектов, проверочная работа.

**Входной контроль** осуществляется в начале реализации программы в форме беседы и наблюдения и имеет диагностические задачи. Цель входной диагностики – зафиксировать начальный уровень подготовки обучающихся, имеющиеся знания, умения и навыки, связанные с предстоящей деятельностью. Наблюдение осуществляется в течение реализации программы.

**Промежуточный контроль** осуществляется в целях диагностики теоретических знаний и практических умений и навыков по итогам освоения одного из разделов курса. Проводится в форме демонстрации проекта.

**Итоговый контроль** проводится с целью определения степени достижения результатов обучения и получения сведений для совершенствования программы и методов обучения – представляет из себя защиту проекта

**11. Оценочные материалы, формирующие систему оценивания**

Промежуточная аттестация состоит кейс-задания, которое обучающий должен сделать в рамках определенной темы. Примеры заданий приведены в приложении 1. Максимальный балл за промежуточную аттестацию: 40 баллов.

Оценивание кейс-задания осуществляется по следующим уровням:

- высокий уровень – учащийся набрал не менее 80% от максимально возможного количества баллов (от 32 баллов).
- средний уровень – учащийся набрал не менее 50% от максимально возможного количества баллов (от 20 баллов).
- низкий уровень – учащийся набрал менее 50% от максимально возможного количества баллов (менее 20 баллов).

Итоговый контроль представляет из себя защиту проекта. Проект представляет собой игру, которую обучающийся должен создать самостоятельно, используя полученные знания на курсе.

## Критерии оценки проекта

№	Название критерия	Максимальный балл
1	Актуальность и проработанность проблемы	До 5 баллов
2	Четкость формулировки целей и задач	До 5 баллов
3	Технологическая сложность проекта. Наличие нестандартных структур данных и алгоритмов, специфика которых направлена на решение конкретной задачи.	До 10 баллов
4	Новизна и оригинальность решения	До 5 баллов
5	Качество разработанного продукта. Наличие удобного и понятного интерфейса, как абстракции между реализацией и пользователем.	До 5 баллов
6	Защита проекта: – качество презентации; – четкость и ясность изложения, умение взаимодействовать с аудиторией, отвечать на вопросы	До 5 баллов
7	Наличие самооценки и перспектив дальнейшей разработки проекта	До 5 баллов
	<b>ИТОГО</b>	<b>40 баллов</b>

**Оценивание проекта осуществляется по следующим уровням:**

**Высокий уровень** – учащийся набрал не менее 32 баллов по итогам защиты проекта.

**Средний уровень** – учащийся набрал от 20 до 31 балла по итогам защиты проекта.

**Низкий уровень** – учащийся набрал менее 20 баллов по итогам защиты проекта.

**Общими критериями оценки результативности обучения являются:**

- оценка уровня теоретических знаний: свобода восприятия теоретической информации, осмысленность и свобода использования специальной терминологии, свобода ориентации в теоретическом материале;

- оценка уровня практической подготовки: соответствие развития уровня практических умений и навыков программным требованиям, свобода владения специальным оборудованием и оснащением;

- оценка уровня достижения личностных результатов: культура организации самостоятельной деятельности, культура работы с информацией, аккуратность и ответственность при работе.

**Оценка итоговых результатов освоения программы осуществляется по трем уровням:**

**Высокий уровень** – достижение 80- 100% показателей освоения программы.

**Средний уровень** – достижение 50- 79% показателей освоения программы.

**Низкий уровень** – достижение менее чем 50% показателей освоения программы.

Достиженные обучающимся знания, умения и навыки заносятся в сводную таблицу результатов обучения.

### Оценка уровней освоения программы

Уровни	Параметры	Показатели
<b>Высокий уровень (80-100%)</b>	Теоретические знания	Обучающийся освоил материал в полном объеме. Знает и понимает значение терминов, употребляет их осознанно и в полном соответствии с содержанием. Самостоятельно ориентируется в содержании материала по темам.
	Практические умения и навыки	Обучающийся овладел 80-100% умений и навыков, предусмотренных программой за конкретный период. Умет работать самостоятельно, применяя практические умения и навыки. Правильно и по назначению применяет инструменты. Умеет выполнять основные логические действия (анализ, синтез, установление причинно-следственных связей). Способен планировать и регулировать свою деятельность по реализации проекта. Умеет осуществлять поиск информации, в том числе в сети Интернет; выслушивать собеседника и вести диалог; выражать свои мысли в соответствии с задачами и условиями коммуникации.
	Личностные результаты	Обучающийся обладает внутренней мотивацией. Способен самостоятельно организовать собственную деятельность. Сформирована культура работы с информацией. Работу выполняет аккуратно, доводит до конца. Может оценить результаты выполнения своего задания и дать оценку работы своего товарища.
<b>Средний уровень (50-79%)</b>	Теоретические знания	Учащийся освоил базовые знания, ориентируется в содержании материала по темам, иногда обращается за помощью к педагогу. Использует специальную терминологию, однако сочетает её с бытовой.
	Практические умения и навыки	Владеет базовыми навыками и умениями, но не всегда может выполнить задание самостоятельно, просит помощи педагога. В основном выполняет задания на основе образца. Способен разработать проект с помощью преподавателя. Встречаются отдельные случаи неправильного применения инструментов. Делает ошибки в работе, но может устранить их после наводящих вопросов или самостоятельно. Испытывает незначительные затруднения при выполнении основных логических действий (анализ, синтез, установление причинно-следственных связей). Способен планировать и регулировать свою деятельность по реализации проекта с помощью педагога. Испытывает незначительные сложности в осуществлении коммуникации с педагогом и сверстниками.



	Личностные результаты	Внутренняя мотивация к обучению сочетается с внешней. В работе допускает небрежность. Работу не всегда выполняет аккуратно и/или доводит до конца. Оценить результаты своей деятельности может с подсказкой педагога
<b>Низкий уровень (меньше 50%)</b>	Теоретические знания	Владеет минимальными знаниями, ориентируется в содержании материала по темам только с помощью педагога. Избегает употреблять специальные термины.
	Практические умения и навыки	Владеет минимальными начальными навыками и умениями. Учащийся способен выполнять каждую операцию только с подсказкой педагога или товарищей. Часто неправильно применяет необходимый инструмент или не использует его вовсе. В работе допускает грубые ошибки, не может их найти даже после указания преподавателя. В состоянии выполнять лишь простейшие практические задания педагога. Испытывает существенные затруднения при выполнении основных логических действий (анализ, синтез, установление причинно-следственных связей). Не способен планировать и регулировать свою деятельность по реализации проекта. Испытывает значительные сложности в осуществлении коммуникации с педагогом и сверстниками.
	Личностные результаты	Преобладает внешняя мотивация к обучению. Работу часто выполняет неаккуратно и/или не доводит до конца. Не способен самостоятельно и объективно оценить результаты своей работы.

### Сводная таблица результатов обучения по программе

№ п/п	ФИО	Оценка теоретических знаний	Оценка практических умений и навыков (предметных и метапредметных)	Личностные результаты	Итоговая оценка
1					
2					
3					

### Анализ диагностической работы

Количество обучающихся по списку		
Количество выполнявших работу		
Количество	«Высокий уровень»	
	«Средний уровень»	
	«Низкий уровень»	
Успеваемость (в %)		
Качество обученности (в %)		

## УЧЕБНЫЙ ПЛАН

№	Название разделов, тем	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
<b>1</b>	<b>Основы алгоритмизации и программирования</b>	<b>6</b>	<b>3</b>	<b>3</b>	
1.1	Вводное занятие	2	1	1	Наблюдение
1.2	Введение. Понятие алгоритма. Решение задач.	4	2	2	Наблюдение
<b>2</b>	<b>Работа со средой Scratch</b>	<b>70</b>	<b>27</b>	<b>43</b>	
2.1	Знакомство со средой Scratch. Навигация в среде Scratch.	4	2	2	Наблюдение
2.2	Блоки движения и событий	2	1	1	Наблюдение
2.3	Знакомство с линейными алгоритмами в Scratch.	2	1	1	Демонстрация решения
2.4	Знакомство с блоками «Внешний вид». Анимация персонажей с использованием костюмов, фонов из библиотеки	6	2	4	Демонстрация решения
2.5	Знакомство с графическим редактором Scratch. Создание первого спрайта.	6	2	4	Наблюдение
2.6	Работа со звуком Scratch	4	2	2	Демонстрация решения
2.7	Знакомство с пером	4	2	2	Наблюдение
2.8	Знакомство с отрицательными числами и координатами.	2	1	1	Наблюдение
2.9	Условные алгоритмы	6	2	4	Наблюдение
2.10	Понятие переменная. Работа с переменными	4	2	2	Наблюдение
2.11	Циклические алгоритмы. Блоки «Управления».	6	2	4	Наблюдение
2.12	Создание проекта с вычислениями. Случайные числа.	6	2	4	Демонстрация решения
2.13	Создание проекта с подсчетом очков	6	2	4	Демонстрация решения
2.14	Работа со списками	4	2	2	Демонстрация решения
2.15	Создание подпрограмм	6	2	4	Демонстрация решения
2.16	Промежуточное тестирование	2	0	2	Проверочная работа
<b>3</b>	<b>Знакомство со средой программирования Spike App</b>	<b>8</b>	<b>1</b>	<b>7</b>	
3.1	Знакомство с конструктором Lego Spike Prime.	2	1	1	Опрос
3.2	Сборка робота	2	0	2	Опрос
3.3	Программирование робота	4	0	4	Демонстрация решения

<b>4</b>	<b>Создание мультфильмов и игр</b>	<b>60</b>	<b>28</b>	<b>32</b>	
4.1	Создание мультфильма. Разработка темы и персонажей	4	2	2	Демонстрация решения
4.2	Создание мультфильма. Программирование	4	2	2	Демонстрация решения
4.3	Создание игры. Разработка темы и персонажей	4	2	2	Демонстрация решения
4.4	Создание игры. Программирование	4	2	2	Демонстрация решения
4.5	Создание игры с несколькими уровнями. Разработка темы и персонажей	4	2	2	Демонстрация решения
4.6	Программирование игры с несколькими уровнями	4	2	2	Демонстрация решения
4.7	Добавление анимации.	4	2	2	Демонстрация решения
4.8	Добавление подсчета очков.	4	2	2	Демонстрация решения
4.9	Создание собственного проекта. Тема. Цели, задачи, изучение предметной области.	6	2	4	Демонстрация решения
4.10	Создание собственного проекта. Программирование	4	2	2	Демонстрация решения
4.11	Создание собственного проекта. Добавление музыки и анимации	4	2	2	Демонстрация решения
4.12	Создание собственного проекта. Разработка презентации	4	2	2	Демонстрация решения
4.13	Создание собственного проекта. Подготовка к защите	4	2	2	Демонстрация решения
4.14	Создание собственного проекта. Репетиция защиты	4	2	2	Демонстрация решения
4.15	Защита итогового проекта	2	0	2	Защита проекта
	<b>ИТОГО</b>	<b>144</b>	<b>59</b>	<b>85</b>	

## СОДЕРЖАНИЕ УЧЕБНОГО ПЛАНА

### 1. Основы алгоритмизации и программирования

#### 1.1 Вводное занятие

**Теория.** Правила работы в компьютерном кабинете. Инструктаж по технике безопасности. Введение в курс программирования на Scratch.

**Практика.** Игра на знакомство.

#### 1.2 Введение. Понятие алгоритма. Решение задач

**Теория.** Понятие алгоритма и исполнителя алгоритмов. Понятие отладки программы. Основные алгоритмические конструкции: линейный алгоритм, циклический алгоритм, алгоритм ветвления. Способы представления алгоритмов (словесный, графический, программный). Определяемые допустимые действия.

**Практика.** Решения задач с использованием алгоритмов.

## **2. Работа со средой Scratch**

### **2.1 Знакомство со средой Scratch. Навигация в среде Scratch**

**Теория.** Изучение основных элементов интерфейса Scratch

**Практика.** Изучение пользовательского интерфейса 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.

### **2.2 Блоки движения и событий**

**Теория.** Изучение блоков движения и событий. Запуск проектов с помощью различных событий.

**Практика.** Обучение спрайта различным движением. Запуск кода с помощью разных событий.

### **2.3 Знакомство с линейными алгоритмами Scratch**

**Теория.** Изучение блоков, позволяющих создать линейный алгоритм.

**Практика.** Создание простейшего приложения с использованием линейных алгоритмов.

### **2.4 Знакомство с блоками «Внешний вид». Анимация персонажей с использованием костюмов, фонов из библиотеки**

**Теория.** Изучение блоков «Внешний вид». Добавление нескольких персонажей и фонов.

**Практика.** Создание анимированного персонажа с использованием готовых костюмов.

### **2.5 Знакомство с графическим редактором Scratch. Создание первого спрайта**

**Теория.** Графический редактор Scratch. Растровый и векторный режимы. Редактирование спрайта в векторном графическом редакторе. Слои изображения. Группировка фигур. Сохранение отредактированного спрайта в отдельный файл. Создание нового спрайта с несколькими костюмами в редакторах Scratch. Сохранение нового спрайта в отдельный файл. Создание собственного фона.

**Практика.** Изменение готового персонажа из библиотеки. Сохранение нового спрайта. Создание собственного спрайта и фона.

### **2.6 Работа со звуком Scratch**

**Теория.** Изучение назначения блоков «Звук». Запись собственного звука. Добавление звука в проект с рабочего стола.

**Практика.** Работа со звуками.

## **2.7 Знакомство с пером**

**Теория.** Изучение блоков «Перо».

**Практика.** Создание программы для рисования линий.

## **2.8 Знакомство с отрицательными числами и координатами**

**Теория.** Изучение понятия «отрицательное число». Применение отрицательных чисел в различных блоках. Изучение координатной плоскости. Перемещение спрайта по сцене с помощью координат. Изучение понятия «нулевые координаты».

**Практика.** Добавление отрицательных чисел в ранее созданные проекты. Создание приложения, в котором спрайт перемещается по сцене по разным координатам.

## **2.9 Условные алгоритмы**

**Теория.** Изучение понятия «условный алгоритм». Использование сенсоров касания и команды «если ..., то...». Управление движением персонажа с помощью мыши и клавиатуры.

**Практика.** Создание приложения с использованием условных алгоритмов.

## **2.10 Понятие переменная. Работа с переменными**

**Теория.** Изучение понятия «переменная», для чего нужны переменные. Изучение блоков «Переменные».

**Практика.** Создание приложения с использованием переменных.

## **2.11 Циклические алгоритмы. Блоки «Управления»**

**Теория.** Изучение блоков «Управления». Изучение понятия «цикл».

**Практика.** Создание приложения с использованием циклических алгоритмов.

## **2.12 Создание проекта с вычислениями. Случайные числа**

**Теория.** Изучение блоков «Операторы». Работа с арифметическими операциями, изучение понятия «случайное число».

**Практика.** Создание приложения со случайными числами количества костюмов спрайта.

## **2.13 Создание проекта с подсчетом очков**

**Теория.** Создание переменной для подсчета очков. Разработка темы игры.

**Практика.** Создание игры с подсчетом очков.

## **2.14 Работа со списками**

**Теория.** Изучение понятия «Список». Создание списка. Изучение блоков для работы со списками.

**Практика.** Добавление в ранее созданный проект списка.

## **2.15 Создание подпрограмм**

**Теория.** Последовательность и параллельность выполнения скриптов. Управление через обмен сообщениями. Блоки «Передать сообщение» и «Когда я получу сообщение».

**Практика.** Создание проекта с подпрограммами.

## **2.16 Промежуточное тестирование**

**Практика.** Решение кейс-задачи.

## **3. Знакомство со средой программирования Spike App**

### **3.1 Знакомство с конструктором Lego Spike Prime**

**Теория.** Изучение деталей конструктора. Просмотр обучающих видео.

**Практика.** Сборка простого робота.

### **3.2 Сборка робота**

**Практика.** Сборка выбранной модели робота.

### **3.3 Программирование робота**

**Практика.** Программирование выбранного робота.

## **4. Создание мультфильмов и игр**

### **4.1 Создание мультфильма. Разработка темы и персонажей**

**Теория.** Обсуждение темы мультфильма.

**Практика.** Создание собственного мультфильма. Прорисовка персонажей и фона.

Разработка сценария. Озвучивание и анимация персонажей.

### **4.2 Создание мультфильма. Программирование**

**Теория.** Обсуждение программирование мультфильма.

**Практика.** Программирование мультфильма.

### **4.3 Создание игры. Разработка темы и персонажей**

**Теория.** Обсуждение темы и персонажей проекта.

**Практика.** Создание собственной игры. Создание собственного игрового интерфейса. Реализации игры с одним уровнем.

### **4.4 Создание игры. Программирование**

**Теория.** Обсуждение программирование игры.

**Практика.** Программирование игры.

### **4.5 Создание игры с несколькими уровнями. Разработка темы и персонажей**

**Теория.** Обсуждение темы игры. Обдумывание персонажей.

**Практика.** Создание игры с несколькими уровнями и подсчётом оков.

### **4.6 Программирование игры с несколькими уровнями**

**Теория.** Обсуждение программирования игры.

**Практика.** Программирование игры.

#### **4.7 Добавление анимации**

**Теория.** Обсуждение анимации.

**Практика.** Добавление анимации к игре.

#### **4.8 Добавление подсчёта очков**

**Теория.** Обсуждение добавления переменной.

**Практика.** Добавление переменной «очки». Программирование добавления очков в игре.

#### **4.9 Создание собственного проекта. Цели, задачи, изучение предметной области**

**Теория.** Обсуждение темы проекта.

**Практика.** Выбор темы проекта. Разработка целей и задач, изучение предметной области.

#### **4.10 Создание собственного проекта. Программирование**

**Теория.** Обсуждение программирования.

**Практика.** Программирование проекта.

#### **4.11 Создание собственного проекта. Добавление музыки и анимации**

**Теория.** Обсуждение анимации и музыки.

**Практика.** Добавление в проект музыки и анимации.

#### **4.12 Создание собственного проекта. Разработка презентации**

**Теория.** Обсуждение создания презентации.

**Практика.** Разработка презентации к защите проекта.

#### **4.13 Создание собственного проекта. Подготовка к защите проекта**

**Теория.** Приемы написания речи.

**Практика.** Проверка проекта. Создание речи для защиты проекта.

#### **4.14 Создание собственного проекта. Репетиция защиты**

**Теория.** Создание презентации для защиты.

**Практика.** Проверка проекта. Репетиция речи для защиты проекта.

#### **4.15 Защита итогового проекта**

**Практика.** Защита итогового проекта.

## **КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ**

### **Материально-техническое обеспечение**

Комплекс условий реализации программы:

Аппаратное и техническое обеспечение:

а) Рабочее место учащегося

- ноутбук с выходом в сеть Интернет;
- б) Рабочее место наставника
- ноутбук с выходом в сеть Интернет;
- технические средства обучения (мультимедийное устройство)

### **Методическое обеспечение программы**

#### **Методы обучения:**

- объяснительно-иллюстративный (беседы, объяснения);
- репродуктивный (деятельность обучаемых носит алгоритмический характер, выполняется по инструкциям, предписаниям, правилам в аналогичных, сходных с показанным образцом ситуациях);
- метод проблемного изложения;
- эвристический (метод обучения заключается в организации активного поиска решения выдвинутых в обучении (или самостоятельно сформулированных) познавательных задач в ходе подготовки и реализации творческих проектов);
- исследовательский.

Педагогические технологии: информационные технологии, проектная технология, здоровьесберегающие технологии, технология проблемного обучения.

В процессе обучения, учащиеся работают с программами Scratch, Paint, Lego Education Spike и MS Power Point.

Проектная технология дает возможность самостоятельно конструировать свои знания, ориентироваться в информационном пространстве, развивает критическое и творческое мышление, создаёт условия для формирования и развития внутренней мотивации учащихся к более качественному овладению знаниями, повышения мыслительной активности и приобретения навыков логического мышления.

Здоровьесберегающие технологии позволяют создать максимально возможные условия для сохранения, укрепления и развития эмоционального, интеллектуального и физического здоровья, в том числе в условиях работы с компьютерной техникой.

Проблемное обучение — это тип развивающего обучения, содержание которого представлено системой проблемных задач различного уровня сложности, в процессе решения которых учащиеся овладевают новыми знаниями и способами действия, а через это происходит формирование творческих способностей: продуктивного мышления, воображения, познавательной мотивации, интеллектуальных эмоций.

#### **Учебно-методические средства обучения.**

Для реализации программы используются:



- специализированная литература по направлению, подборка журналов;
- наборы технической документации к применяемому оборудованию;
- образцы программ, выполненные обучающимися и педагогом;
- плакаты, фото и видеоматериалы;
- учебно-методические пособия для педагога и обучающихся, включающие дидактический, информационный, справочный материалы на различных носителях, компьютерное и видео оборудование;
- применяемое на занятиях дидактическое и учебно-методическое обеспечение включает в себя электронные учебники, справочные материалы и системы используемых Программ, Интернет, рабочие тетради обучающихся.

#### **Кадровое обеспечение**

Программу реализуют педагоги дополнительного образования МБУДО «ДДТ «Дриада».

#### **Информационное обеспечение**

Для реализации программы планируется использование следующих информационных ресурсов:

- курс «Программирование в Scratch» (<https://younglinux.info/scratch/>);
- курс «Скретч» (<http://letopisi.ru/index.php/%D0%A1%D0%BA%D1%80%D0%B5%D1%82%D1%87>);
- руководство «Скретч» (<https://scratch.mit.edu/ideas>);
- курс «Творческая мастерская Scratch» (<http://www.nachalka.com/scratch/>);
- клавиатурный тренажер Stamina (<https://stamina-online.com/ru>).

## СПИСОК ЛИТЕРАТУРЫ

### Список литературы для педагогов

1. Винницкий Ю. А. Scratch и Arduino для юных программистов и конструкторов. / Винницкий Ю. А. — СПб.: БХВ-Петербург, 2018. — 176 с.
2. Мэтью Хайлэнд. Программируем с детьми. — М.: Эксмо, 2021. — 96 с.
3. Рейна Бердитт. Программирование на Scratch с нуля. — М.: Эксмо, 2023. — 101 с.
4. Рындак В. Г., Дженжер В. О., Денисова Л. В. Проектная деятельность школьника в среде программирования Scratch: учебно-методическое пособие. / Рындак В. Г., Дженжер В. О., Денисова Л. В. — Оренбург: Оренб. гос. ин-т. менеджмента, 2019. — 116 с.
5. Уфимцева П. Е. Обучение программированию младших школьников в системе дополнительного образования с использованием среды разработки Scratch / Уфимцева П. Е., Рожина И. В. // Наука и перспективы. — 2018. — № 1. — С. 29— 35.

### Список литературы для учащихся

1. Голиков Д. В. Scratch для юных программистов. / Голиков Д. В. — СПб.: БХВ-Петербург, 2018. — 192 с.
2. Голиков Д. В. 42 проекта на Scratch 3 для юных программистов. / Голиков Д. В. — СПб.: БХВ-Петербург, 2019. — 192 с.
3. Маржи М. Scratch для детей. Самоучитель по программированию. / Маржи М. — пер. с англ. М. Гескиной и С. Таскаевой. — М.: Манн, Иванов и Фербер, 2019. — 288 с.
4. Пашковская Ю. В. Творческие задания в среде Scratch: рабочая тетрадь для 5—6 классов. / Пашковская Ю. В. — М., 2018. — 195 с.
5. Свейгарт Эл. Программирование для детей. Делай игры и учи язык Scratch! / Свейгарт Эл. — М.: Эксмо, 2019. — 304 с.
6. Торгашева Ю. В. Первая книга юного программиста. Учимся писать программы на Scratch. / Торгашева Ю. В. — СПб.: Питер, 2020. — 128 с.

### Примерные задания для промежуточной аттестации

#### Список примерных заданий:

– написать программу, в которой будут два спрайта. Если первый спрайт касается второго, то они приветствуют друг друга. Если первый спрайт касается стены, то он останавливается;

– написать программу, в которой первый спрайт съедает второго. Сделать 2 фона, когда съедят спрайта, фон изменяется. При старте программы спрайты появляются каждый раз в новом месте. Если касается края, то отталкивается

– написать программу, в которой сделать движение кота естественным, то есть чтобы назад он ходил так, как человек. Создать два уровня произвольного лабиринта, после прохождения второго уровня необходимо чтобы появлялась надпись «ПОЗДРАВЛЯЕМ С ПОБЕДОЙ»;

– написать программу, в которой необходимо нарисовать четыре колбочки разного цвета. При касании каждой колбочки сделать так, чтобы к нашему спрайту применялся какой-то эффект. Также должна существовать колба, при касании к которой спрайт возвращается к своему первоначальному виду. Озвучить процесс выпивания колбы, а также сделать так, чтобы колба выпивалась постепенно.

**Примерные задания для итоговой аттестации**

Список примерных проектов для итоговой аттестации:

- мультфильм «Танцующие балерины»;
- игра «Гонки»;
- обучающая игра «Калькулятор»;
- игра «Переводчик слов с русского на английский»;
- мультфильм «Сборы в школу»;
- игра «Магазин»;
- проект на подобии игры змейка;
- игра «Лабиринт»;
- игра «Обучение математики»;
- игра «Обучение русскому языку».