

Кировское областное государственное профессиональное  
образовательное автономное учреждение  
«Вятский торгово-промышленный техникум»

**ПРИНЯТО**

На заседании педагогического совета  
Протокол № 11 от 22.04.2024 г.

**УТВЕРЖДАЮ**

Директор КОГПОАУ ВТТТ  
С.Н. Репина

Приказ № 53-ОД от «26» апреля 2024 г.



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА  
технической направленности  
«Основы алгоритмики и логики»**

Возраст обучающихся: 10-12 лет

Срок реализации: 1 год

## 1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Современное общество предъявляет новые требования к подрастающему поколению. Необходимо обладать умениями планирования своей деятельности, поиска информации, необходимой для решения поставленной задачи, построения информационной модели исследуемого объекта или процесса, эффективного использования новых технологий. Такие умения необходимы сегодня каждому человеку. Поэтому важнейшей задачей обучения основам алгоритмики и логики является формирование у учащихся алгоритмического стиля мышления.

Под способностью алгоритмически мыслить понимается умение решать задачи различного происхождения, требующие составления плана действий для достижения желаемого результата. Для того чтобы записать алгоритм решения задачи, необходимо какой-то формальный язык, например, блок-схемы. Программа разработана на основе:

- Федерального закона от 29.12.2012 № 273-ФЗ (ред. от 31.07.2020) «Об образовании в Российской Федерации» (с изм. и доп., вступ. в силу с 01.09.2020);
- Методические рекомендации по организации образовательной деятельности с использованием сетевых форм Письмо Министерства образования и науки Российской Федерации от 28 августа 2015 года №АК-2563/05 «О методических рекомендациях»;
- Приказ 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»);
- Паспорта национального проекта «Образование» (утв. президиумом Совета при Президенте РФ по стратегическому развитию и национальным проектам, протокол от 24.12.2018 № 16);
- Государственной программы Российской Федерации «Развитие образования» (Утверждена Постановлением Правительства РФ от 26.12.2017 № 1642

(ред. от 22.02.2021) «Об утверждении государственной программы Российской Федерации «Развитие образования»;

- Методических рекомендаций по созданию и функционированию центров цифрового образования «IT-куб» (утверждены распоряжением Министерства просвещения Российской Федерации от 12 января 2021 № Р-5);

- Рекомендаций Министерства просвещения Российской Федерации по реализации дополнительной общеобразовательной программы по направлению «Разработка виртуальной и дополненной реальности» с использованием оборудования Центра цифрового образования детей «IT-куб»;

- Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утвержденным приказом Министерства просвещения Российской Федерации от 09.11.2018 № 196;

- 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»;

- Распоряжения Правительства Российской Федерации от 31 марта 2022 года № 678-р «Концепция развития дополнительного образования детей до 2030 года»;

- Распоряжения Правительства Российской Федерации от 29 мая 2015 года № 996-р «Стратегия развития воспитания в Российской Федерации на период до 2025 года»;

- Приказа Министерства труда и социальной защиты Российской Федерации от 22.09.2021 № 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых».

**Направленность программы:** техническая.

**Уровень:** стартовый.

**Новизна, актуальность и педагогическая целесообразность программы.**

Актуальность данной образовательной программы состоит в том, что мультимедийная среда Scratch позволяет сформировать у детей интерес к программированию, отвечает всем современным требованиям объективно-ориентированного программирования. Изучение языка значительно облегчает последующий переход к изучению других языков программирования. Также стоит отметить, что большое количество времени уделяется творческим заданиям, выполнение которых благоприятно скажется на развитии творческого потенциала учащихся. Данная программа способствует развитию мотивации к получению новых знаний, возникновению интереса к программированию.

Новизна программы состоит в более углубленном изучении и раскрытии особенно важных элементов в обучении программированию, формировании у обучающихся умения владения компьютером как средством решения практических задач, связанных с алгоритмизацией, программированием, и работой с информацией, подготовке учеников в активной полноценной жизни и работе в условиях технологически развитого общества.

Педагогическая целесообразность данной программы состоит в том, что при изучении программирования в среде Scratch у обучающихся не только развивается логическое мышление, но и формируются навыки работы с мультимедиа.

**Отличительной особенностью** программы является использование метода дифференцированного обучения, основанного на принципах преемственности. Освоение программы происходит в основном в процессе практической творческой деятельности. Знания и умения, приобретенные в результате освоения курса,

являются базой для обучения программированию.

**Цель программы:** освоение базовых навыков для изучения языков программирования высокого уровня в процессе проектной деятельности, направленной на развитие алгоритмического мышления, творческих способностей, и логических компетенций учащихся.

Для достижения цели планируется решить следующие задачи программы:

**Обучающие:**

- 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.

**Развивающие:**

- 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.

**Воспитательные:**

- воспитать умение работать в команде, развитие коммуникативных навыков;
- воспитать трудолюбие и уважительные отношения к интеллектуальному труду.

**Адресат программы:** дети в возрасте от 10 до 12 лет.

**Наполняемость группы:** от 10 до 12 человек.

**Содержание и объем стартовых знаний, необходимых для начального этапа освоения программы:** умение читать и писать, решать арифметические задачи.

**Срок реализации программы:** 1 год.

**Объем программы:** 72 часа.

**Режим занятий:** 2 раза в неделю по 1 академическому часу.

**Формы организации учебной деятельности:** групповая, индивидуальная, коллективная.

## 2. ПРОГНОЗИРУЕМЫЕ РЕЗУЛЬТАТЫ

### **Предметные результаты:**

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:**

- строить различные виды алгоритмов (линейных, разветвляющихся, циклических) для решения поставленных задач;
- строить различные алгоритмы 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.

### 3. СОДЕРЖАНИЕ ПРОГРАММЫ

#### Учебно-тематический план

№ п/п	Название разделов, тем	Количество часов			Форма аттестации/контроля
		Всего	Теория	Практика	
	<b>1. Основы алгоритмизации и программирования</b>	<b>2</b>	<b>1</b>	<b>1</b>	
1.1	Вводное занятие	1	1	0	Наблюдение
1.2	Знакомство со средой Scratch. Навигация в среде Scratch.	1	0,5	0,5	Наблюдение
	<b>2. Работа со средой Scratch</b>	<b>34</b>	<b>15</b>	<b>19</b>	
2.1	Блоки движения и событий	1	0	1	Наблюдение
2.2	Знакомство с линейными алгоритмами в Scratch.	2	1	1	Демонстрация проекта
2.3	Знакомство с блоками «Внешний вид». Анимация персонажей с использованием костюмов, фонов из библиотеки.	2	1	1	Демонстрация проекта
2.4	Знакомство с графическим редактором Scratch. Создание первого спрайта и фона.	2	1	1	Наблюдение
2.5	Работа со звуком Scratch	2	1	1	Демонстрация проект
2.6	Знакомство с пером	2	1	1	Наблюдение
2.7	Знакомство с отрицательными числами и координатами.	2	1	1	Наблюдение
2.8	Введение. Понятие алгоритма. Решение задач.	2	1	1	Наблюдение
2.9	Условные алгоритмы	3	1	2	Наблюдение
2.10	Понятие переменная. Работа с переменными.	2	1	1	Наблюдение
2.11	Знакомство с клонами	2	1	1	Демонстрация проекта
2.12	Циклические алгоритмы. Блоки «Управления».	2	1	1	Наблюдение
2.13	Создание проекта с вычислениями. Случайные числа.	3	1	2	Демонстрация проекта
2.14	Создание проекта с подсчетом очков	3	1	2	Демонстрация проекта
2.15	Работа со списками	2	1	1	Демонстрация проекта
2.16	Создание подпрограмм	2	1	1	Демонстрация проекта
	<b>3. Создание проектов в среде программирования Scratch</b>	<b>36</b>	<b>17</b>	<b>19</b>	

3.1	Последовательность и параллельность выполнения скриптов	2	1	1	Наблюдение
3.2	Взаимодействие между спрайтами. Управление через обмен сообщениями.	2	1	1	Наблюдение
3.3	Виды компьютерных игр. Алгоритмическая разработка программы	2	1	1	Демонстрация проекта
3.4	Разработка базовых спрайтов для игры. Формирование базовых скриптов	2	1	1	Демонстрация решения
3.5	Синхронизация работы скриптов от разных спрайтов	2	1	1	Демонстрация решения
3.6	Переход из одной сцены в другую	2	1	1	Демонстрация решения
3.7	Создание игры с несколькими уровнями. Разработка темы и персонажей	2	1	1	Демонстрация решения
3.8	Программирование игры с несколькими уровнями.	2	1	1	Демонстрация решения
3.9	Добавление анимации.	2	1	1	Демонстрация решения
3.10	Добавления подсчета очков.	2	1	1	Демонстрация решения
3.11	Создание собственного проекта. Тема. Цели, задачи, изучение предметной области.	4	2	2	Демонстрация решения
3.12	Создание собственного проекта. Программирование	2	1	1	Демонстрация решения
3.13	Создание собственного проекта. Добавление музыки и анимации	2	1	1	Демонстрация решения
3.14	Создание собственного проекта. Разработка презентации	2	1	1	Демонстрация решения
3.15	Создание собственного проекта. Подготовка к защите	2	1	1	Демонстрация решения
3.16	Создание собственного проекта. Репетиция защиты	2	1	1	Демонстрация решения
3.17	Защита итогового проекта	2	-	2	Защита проекта.
	<b>Итого</b>	<b>72</b>	<b>33</b>	<b>39</b>	

## Содержание учебно-тематического плана

**1. Основы алгоритмизации и программирования.** (*теория – 1 часа, практика – 1 час*)

### **1.1 Вводное занятие**

Теория: Правила работы в компьютерном кабинете. Инструктаж по техники безопасности. Введение в курс программирования на Scratch.

### **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.** (*теория – 15 часов, практика – 19 часов*)

### **2.1 Знакомство со средой 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 и фона**

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. Знакомство с 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 Знакомство с клонами**

Теория: Изучение блоков «Клоны». Изучение возможностей клонов.

Практика: Создание проекта с клонами.

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

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

Практика: Создание приложения с использованием циклических алгоритмов.

## **2.13 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.

## **2.14 Создание проекта с подсчетом очков**

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

Практика: Создание игры с подсчетом очков.

### **2.15 Работа со списками**

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

Практика: Добавление в ранее созданный проект списка.

### **2.16 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.

## **3. Создание проектов в среде программирования Scratch (теория -17 часов, практика - 19 часов)**

### **3.1 Последовательность и параллельность выполнения скриптов**

Теория: Последовательные и параллельные потоки в программах Scratch. Одновременная и попеременная работа нескольких исполнителей.

Практика:

Создание Scratch-историй с одновременной и попеременной работой нескольких исполнителей.

### **3.2 Взаимодействие между спрайтами. Управление через обмен сообщениями**

Теория:

Решение проблемы появления новых исполнителей только после того, как старые исполнители выполнили свои действия. Взаимодействие спрайтов с неподвижными объектами с помощью команд «касается» и «касается цвета». Взаимодействие спрайтов с помощью команд «передать» и «когда я получу». Использование сообщений для создания событий.

Практика:

Создание Scratch-историй с взаимодействием нескольких исполнителей и неподвижных объектов. Создание Scratch-историй с взаимодействием нескольких исполнителей.

### **3.3 Виды компьютерных игр. Алгоритмическая разработка программы**

Теория:

Компьютерные игры: вред или польза. Виды компьютерных игр. Этапы разработки игр программистами.

Практика:

Алгоритмическая разработка проекта, запись на естественном языке событий и точек взаимодействия героев будущей игры.

### **3.4 Разработка базовых спрайтов для игры. Формирование базовых скриптов**

Теория:

Логика создания персонажей для игры. Перевод алгоритма, написанного на естественном языке, в коды Scratch.

Практика:

Разработка и создание основных спрайтов и их костюмов для будущей игры.

Разработка скриптов для спрайтов и объектов.

### **3.5 Синхронизация работы скриптов для разных спрайтов**

Практика:

Доработка программы с целью установления связей между спрайтами.

Тестирование и отладка программы.

### **3.6 Переход из одной сцены в другую**

Теория:

Односторонний (без возможности вернуться назад) переход из одного пространства в другое. Понятие интерфейса. Элементы интерфейса. Основные принципы дизайна интерфейсов. Обратная связь. Необходимые элементы меню.

Практика:

Создание программы для перемещения объекта по игровой карте и разработка интерфейса для Scratch-проекта.

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

Теория: Обсуждение темы игры. Обдумывание персонажей.

Практика: Создание игры с несколькими уровнями и подсчётом очков

### **3.8 Программирование игры с несколькими уровнями**

Теория: Обсуждение программирования игры

Практика: Программирование игры.

### **3.9 Добавление анимации**

Теория: Обсуждение анимации.

Практика: Добавление анимации к игре

### **3.10 Добавление подсчёта очков**

Теория: Обсуждение добавления переменной.

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

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

Теория: Обсуждение темы проекта.

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

### **3.12 Создание собственного проекта. Программирование**

Теория: Обсуждение программирования.

Практика: Программирование проекта.

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

Теория: Обсуждение анимации и музыки.

Практика: Добавление в проект музыки и анимации.

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

Теория: Обсуждение создания презентации.

Практика: Разработка презентации к защите проекта.

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

Теория: Обсуждение речи.

Практика: Проверка проекта. Создание речи для защиты проекта.

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

Теория: Создание презентации для защиты.

Практика: Проверка проекта. Репетиция речи для защиты проекта.

### **3.17 Защита итогового проекта**

Практика: Защита итогового проекта.

#### 4. ФОРМЫ АТТЕСТАЦИИ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Формы аттестации: наблюдение, опрос, защита проекта, демонстрация проекта, беседа, решения задач, участие в мероприятиях различного уровня. Наблюдение предполагает просмотр работ учащихся на занятие. Опрос и беседа предполагает разговор с учениками в конце занятия. Демонстрация проекта предполагает выход обучающегося к доске и показ выполненного проекта. Решение задач предполагает выполнение заданий или коллективно, или индивидуально. Участие в мероприятиях предполагает участие обучающегося в мероприятиях различного уровня.

Входной контроль - имеет диагностические задачи и осуществляется в начале учебного года. Цель предварительной диагностики – зафиксировать начальный уровень подготовки обучающихся, имеющиеся знания, умения и навыки, связанные с предстоящей деятельностью. Входной контроль будет проводиться в форме творческого задания.

Промежуточная аттестация проводится на основании диагностики теоретических знаний и практических умений и навыков по итогам освоения модуля. Промежуточная аттестация проводится в следующих формах: защита кейсов, защита творческих или исследовательских работ и проектов, конференции, выставочный просмотр, смотр знаний и умений, викторины, олимпиада, конкурс, соревнование, турнир и пр.

Итоговый контроль проводится в конце года с целью определения степени достижения результатов обучения и получения сведений для совершенствования программы и методов обучения.

#### Критерии оценки проекта

№	Название критерия	Описание	Максимальный балл
1	Технологическая сложность проекта	Уровень детализации и сложности проекта. Использование алгоритмических структур.	До 6 баллов
2	Новизна решения	Работа содержит оригинальные решения.	До 6 баллов
3	Гармоничность, грамотность и яркость работы	Яркая анимация. Цветовая гамма, формы, размеры объектов подобны с учетом правил гармонии.	До 4 баллов

4	Эмоциональное воздействие на аудиторию	Проект ученика вызывает положительные эмоции.	До 2 баллов.
5	Качество проекта	Проект завершен, соответствует поставленным целям и задачам	До 6 баллов
6	Четкость формулировки целей и задач	Цели и задачи сформулированы и озвучены	До 2 баллов
7	Защита проекта: – оформление презентации; – культура речи; – ответы на вопросы.	– презентация оформлена грамотно, выбран подходящий шаблон, имеется анимация, на слайдах минимум тестовой информации; – ученик говорит уверенно; – ученик смог ответить на все вопросы жюри.	До 3 баллов  До 3 баллов  До 3 баллов
Максимум баллов: 35 баллов			

Анализ результатов итоговой аттестации – защиты проекта.

Высокий уровень – учащийся набрал не менее 28 баллов по итогам защиты проекта.

Средний уровень – учащийся набрал от 17 до 27 баллов по итогам защиты проекта.

Низкий уровень – учащийся набрал менее 17 баллов по итогам защиты проекта.

Итоговая оценка развития личностных качеств воспитанника производится по трём уровням:

– «высокий»: положительные изменения личностного качества воспитанника в течение учебного года признаются как максимально возможные для него;

– «средний»: изменения произошли, но воспитанник потенциально был способен к большему;

– «низкий»: изменения не замечены.

Результатом усвоения обучающимися программы: устойчивый интерес к занятиям по алгоритмике и логике, результаты достижений в массовых мероприятиях различного уровня.

Общими критериями оценки результативности обучения являются:

– оценка уровня теоретических знаний: широта кругозора;

– свобода восприятия теоретической информации, развитость практических навыков работы со специальной литературой, осмысленность и свобода использования специальной терминологии;

– оценка уровня практической подготовки учащихся: соответствие развития уровня практических умений и навыков программным требованиям, свобода владения специальным оборудованием и оснащением, качество выполнения практического задания, технологичность практической деятельности;

– оценка уровня развития и воспитанности обучающихся: культура организации самостоятельной деятельности, аккуратность и ответственность при работе, развитость специальных способностей, умение взаимодействовать с членами коллектива.

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.

Достигнутые 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 на использует его вовсе. В работе допускает грубые ошибки, не может их найти их 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	Фамилия, имя обучающегося	Оценка теоретических знаний	Оценка практических умений и навыков (предметных и метапредметных)	Личностные результаты	Итоговая оценка
1.					
2.					
3.					

## **5. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ**

### **Материально-техническое обеспечение**

- кабинет, оснащенный компьютерной техникой, не менее 1 ПК на 1 ученика.
- образовательный набор Lego Spike Prime;
- ПО: Scratch 3, любой браузер.

Рабочее место учащегося:

- ноутбук с выходом в сеть

Интернет. Рабочее место наставника

- ноутбук с выходом в сеть Интернет;
- технические средства обучения (ТСО)  
(мультимедийное устройство).

### **Методическое обеспечение программы**

Методы обучения:

- объяснительно-иллюстративный (беседы, объяснения, дискуссии);
- 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: проектная технология, здоровьесберегающие технологии, технология проблемного обучения. Проектная технология дает возможность самостоятельно конструировать свои знания, ориентироваться в информационном пространстве, развивается критическое и творческое мышление. Если проектная технология является спланированной и постоянной составляющей частью образовательного процесса, то будут созданы условия для формирования и

развития внутренней мотивации учащихся к более качественному овладению знаниями, повышения мыслительной активности и приобретения навыков логического мышления.

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:**

- 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 учащихся.

## **6. СПИСОК ЛИТЕРАТУРЫ**

### **Список литературы для педагога**

1. Босова Л. Л. Информатика. 8 класс : учебник. / Босова Л. Л. — М.: БИНОМ. Лаборатория знаний, 2016. — 176 с.
2. Винницкий Ю. А. Scratch и Arduino для юных программистов и конструкторов./ Винницкий Ю. А. — СПб.: БХВ-Петербург, 2018. — 176 с.
3. Поляков К. Ю. Информатика. 7 класс (в 2 частях): учебник. Ч. 1 / Поляков К. Ю., Еремин Е. А. — М.: БИНОМ. Лаборатория знаний, 2019. — 160 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, 2009. — 116 с.
5. Семакин, И. Г. Информатика и ИКТ: учебник для 9 класса. / Семакин, И. Г., Залогова, Л. А. и др. М: БИНОМ. Лаборатория знаний, 2014. — 171 с.
6. 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 р литературы для учащихся**

1. Голиков Д. В. Scratch для юных программистов. / Голиков Д. В. — СПб.: БХВ-Петербург, 2017. — 192 с.
2. Маржи М. Scratch для детей. Самоучитель по программированию. / Маржи М. — пер. с англ. М. Гескиной и С. Таскаевой. — М.: Манн, Иванов и Фербер, 2017. — 288 с.
3. Пашковская Ю. В. Творческие задания в среде Scratch: рабочая тетрадь для 5—6 классов. / Пашковская Ю. В. — М., 2018. — 195 с.

4. Торгашева Ю. В. Первая книга юного программиста. Учимся писать программы на Scratch. / Торгашева Ю. В. — СПб.: Питер, 2016. — 128 с.
5. Свейгарт Эл. Программирование для детей. Делай игры и учи язык Scratch! / Свейгарт Эл. — М.: Эксмо, 2017. — 304 с.