

Муниципальное автономное общеобразовательное учреждение  
**«Новоатъяловская средняя общеобразовательная школа»**  
ул. Школьная, д. 20, с. Новоатъялово, Ялуторовский р-н, Тюменская обл, 627050  
тел./факс 8 (34535) 34-1-60, e-mail: [novoat\\_school@inbox.ru](mailto:novoat_school@inbox.ru)  
ОКПО 45782046, ОГРН 1027201465741, ИНН/КПП 7228005312/720701001


**Рассмотрено**

На заседании  
педагогического совета

Протокол № 6

от 30.06.2023 г

**Согласовано:**

 В.А. Кривошекова

Руководитель центра «Точка  
роста»

**Утверждено:**



Ф.Ф. Исхакова

Директор МАОУ  
«Новоатъяловская СОШ»

Приказ

№ 107-09 от 31.08.2023 г.

**РАБОЧАЯ ПРОГРАММА ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ**  
**«Программирование на языке Scratch»**  
**Направленность «общеинтеллектуальная»**

Класс: 5 - 8

Количество часов 68 (2 часа в неделю)

Учитель: Большаков М.В.

**Срок реализации: 1 год**

Большаков Максим Валерьевич,  
Учитель информатики,  
высшая квалификационная категория

2023–2024 учебный год

## 1. Пояснительная записка

### **Нормативно правовые документы программы дополнительного образования.**

Федеральный закон от 29.12.2012 № 273 ФЗ «Об образовании в Российской Федерации» ( с изм., внесенными Федеральными законами от 04.04.2014 г. № 145-ФЗ от 06.04.2015 г. № 68-ФЗ (ред. 19.12.2016)) в действующей редакции;

Приказ Минпросвещения России от 03.09.2019 №467 "Об утверждении Целевой модели развития региональных систем развития дополнительного образования детей"

Национальный проект "Образование" - ПАСПОРТ УТВЕРЖДЕН президиумом Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам (протокол от 24 декабря 2018 г. № 16)

Федеральный проект «Успех каждого ребенка» - ПРИЛОЖЕНИЕ к протоколу заседания проектного комитета по национальному проекту "Образование" от 07 декабря 2018 г. № 3

Указ Президента Российской Федерации от 7 мая 2018 г. № 204 «О национальных целях и стратегических задачах развития Российской Федерации на период до 2024 года».

Постановление Правительства Российской Федерации от 31 октября 2018 г. № 1288 «Об организации проектной деятельности

Распоряжение Правительства Российской Федерации от 6 июля 2018 г. N 1375, об утверждении Плана основных мероприятий до 2020 года, проводимых в рамках Десятилетия детства.

Приказ Министерства просвещения Российской Федерации от 09 ноября 2018 N 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

Федеральная целевая программа развития образования на 2016–2020 годы, утвержденная Постановлением Правительства Российской Федерации от 23 мая 2015 г. № 497

Концепция общенациональной системы выявления и развития молодых талантов на 2015-2020 годы (утверждена Президентом Российской Федерации 3 апреля 2012 г. № Пр-827) и комплекс мер по ее реализации (утвержден Правительством Российской Федерации 27 мая 2015 г. № 3274п-П8)

Концепция развития дополнительного образования детей, утвержденная Распоряжением Правительства Российской Федерации от 4 сентября 2014 г. № 1726-р.

Федеральный закон РФ 273-ФЗ «Об образовании в Российской Федерации» от 29 декабря 2012 г.

Стратегия инновационного развития Российской Федерации на период

до 2020 года, утвержденная Распоряжением Правительства Российской Федерации от 8 декабря 2011 г. № 2227-р

Актуальность программы:

Дополнительная общеразвивающая программа «Scratch программирование» (далее Программа) реализуется в соответствии с технической направленностью образования.

В последние годы стал популярным язык и одноименная среда программирования - Scratch. Это можно объяснить потребностью и педагогического сообщества, и самих детей в средстве, которое позволит легко и просто, но не бездумно, исследовать и проявить свои творческие способности.

Данная программная среда дает принципиальную возможность составлять сложные по своей структуре программы, не заучивая наизусть ключевые слова, и при этом в полной мере проявить свои творческие способности и понять принципы программирования.

Отличительные особенности среды программирования Scratch это:

- ✓ объектная ориентированность;
- ✓ поддержка событийно-ориентированного программирования; параллельность выполнения скриптов;
- ✓ дружественный интерфейс;
- ✓ разумное сочетание абстракции и наглядности; организация текстов программ из элементарных блоков;
- ✓ наличие средств взаимодействия программ на Scratch с реальным миром посредством дополнительного устройства;
- ✓ встроенная библиотека объектов; встроенный графический редактор; активное интернет-сообщество пользователей.

**Аспект новизны** заключается в том, что Scratch не просто язык программирования, а еще и интерактивная среда, где результаты действий визуализированы, что делает работу с программой понятной, интересной и увлекательной.

Особенность среды Scratch, позволяющая создавать в программе мультфильмы, анимацию и даже простейшие игры, делает образовательную программу по

программированию практически значимой для современного школьника, т.к. дает возможность увидеть практическое назначение алгоритмов и программ, что будет способствовать развитию интереса к профессиям, связанным с программированием.

**Инновацией программы** является выявление и сопровождение работы одаренного ученика. Обучающимся представляется перечень проектов, по выбору с которыми они смогут работать индивидуально, составляется индивидуально-образовательный маршрут. Значимым условием успешного

развития одаренного ученика является максимальная индивидуализация его творческой деятельности.

Основной вид деятельности- -практическая работа, проектная деятельность. В рамках освоения данной программы создаются условия для разнообразной индивидуальной практической, проектной и исследовательской деятельности обучающихся.

Курс построен таким образом, чтобы помочь учащимся заинтересоваться программированием вообще и найти ответы на вопросы, с которыми им приходится сталкиваться в повседневной жизни при работе с большим объемом информации; при решении практических и жизненных задач. Курс «Программирование на языке Scratch» позволяет создавать собственные программы для решения конкретной задачи. Это является отличительной особенностью данной программы.

Актуальность проектной деятельности сегодня осознаётся всеми. ФГОС нового поколения требует использования в образовательном процессе технологий деятельного типа, методы проектно-исследовательской деятельности определены как одно из условий реализации основной образовательной программы образования.

Следует иметь в виду, что возрастные особенности школьника среднего возраста не позволяют в полной мере реализовать проведение полноценных научных исследований.

Раннее включение в организованную специальным образом проектную деятельность творческого характера позволяет сформировать у школьника познавательный интерес и исследовательские навыки. В будущем они станут основой для организации научноисследовательской деятельности в вузах, колледжах, техникумах и т.д.

Организация научно-познавательной деятельности школьника требует использования инструмента (средства) для выполнения как исследовательских, так и творческих проектов. В качестве такого инструмента можно использовать среду программирования Scratch.

**Перспективность программы:** предлагаемая программа внеурочной деятельности

«Программирование на языке Scratch» является отличной средой для проектной деятельности.

В ней есть все необходимое:

- графический редактор для создания и модификации визуальных объектов;
- библиотека готовых графических объектов (некоторые из них содержат наборы скриптов);
- библиотека звуков и музыкальных фрагментов;
- большое количество примеров.

Scratch является отличным инструментом для организации научно-познавательной деятельности школьника благодаря нескольким факторам:

- эта программная среда легка в освоении и понятна школьникам, но при этом
- она позволяет

составлять сложные программы;

- эта программа позволяет заниматься и программированием, и созданием творческих

проектов;

- вокруг Scratch сложилось активное, творческое международное сообщество.

Язык Scratch особенно интересен для начального уровня изучения программирования (5-6 классы), но этот же язык может быть использован для изучения программирования на продвинутом уровне (7-8 классы). Обучение основам программирования в этой среде

наиболее эффективно при выполнении небольших (поначалу) проектов. При этом естественным образом ученик овладевает интерфейсом новой для него среды, постепенно углубляясь как в возможности Scratch, так и в идеи собственно программирования. Базовый

проект един для всех учеников и выполняется совместно с учителем. Затем предлагаются возможные направления развития базового проекта, которые у разных учеников могут быть различными.

При создании сложных проектов ученик не просто освоит азы программирования, но и познакомится с полным циклом разработки программы, начиная с этапа описания идеи и заканчивая тестированием и отладкой.

Scratch не просто среда для программирования, через нее можно выйти на многие другие темы школьной информатики. Создавая свои собственные игры и мультфильмы, дети научатся разрабатывать проекты, ставить цели и задачи.

Чтобы оформить это, нужно

поработать в текстовом редакторе. Потом надо нарисовать героя, окружение.

Разработать алгоритм действий героя, алгоритмы его реакций на события.

Надо будет озвучить героя и

события (записать, обработать звук). Важно и то, что ребенок имеет возможность поделиться результатами своего творчества с друзьями или другими пользователями: кнопка для размещения созданного проекта в Сети

находится непосредственно в программе. Scratch легко перекидывает мостик между программированием и другими школьными

науками. Так возникают межпредметные проекты. Они помогут сделать наглядными понятия отрицательных чисел и координат или, например, уравнения плоских фигур, изучаемых на

уроках геометрии. В них оживут исторические события и географические карты. А тесты по любым предметам сделают процесс обучения веселым и азартным.

Освоив основы Scratch на начальном уровне, можно будет использовать ее на уроках

других предметов (от математики, физики до литературы, рисования, музыки) в качестве среды для создания моделей явлений, ситуаций и т.д.

Таким образом, первое знакомство со средой программирования можно организовать через Scratch (5-6 класс), что для детей означает – через игру. Через Scratch можно раскрыть многие вопросы школьной информатики для школьников 5-8 классов. Обучающиеся познакомятся не только с языком программирования, но и с текстовым, графическим редакторами, элементами пользовательского интерфейса, логикой, новыми математическими понятиями, элементами проектной деятельности.

### **Возраст и контингент детей**

Программа «Программирование на языке Scratch» разработана для организации внеурочной деятельности общеинтеллектуальной направленности в 5-8 классах (11-14 лет).

**Сроки реализации программы:** 1 года.

Количество обучающихся в группе: 10 человек.

### **Режим проведения занятий**

На реализацию программы отводится 68 часов, 2 часа в неделю.

При проведении занятий используются компьютеры с установленной программой Scratch, проектор, сканер, принтер, компьютерная сеть с выходом в Интернет.

**Цель** – Основной целью программы является обучение программированию через создание творческих проектов, развитие личности ребенка, способного к творческому самовыражению, обладающего технической культурой, аналитическим мышлением, навыками и умениями робототехники и программирования, умеющего работать в коллективе, способного применять полученные знания при решении бытовых и учебных задач.

#### **Задачи:**

##### ***Обучающие задачи***

- ✓ овладеть навыками составления алгоритмов;
- ✓ овладеть понятиями «объект», «событие», «управление», «обработка событий»;
- ✓ изучить функциональность работы основных алгоритмических конструкций;
- ✓ сформировать представление о профессии «программист»;
- ✓ сформировать навыки разработки, тестирования и отладки несложных программ;
- ✓ познакомить с понятием проекта и алгоритмом его разработки;

- ✓ сформировать навыки разработки проектов: интерактивных историй, интерактивных игр, мультфильмов, интерактивных презентаций.

### ***Развивающие задачи***

- ✓ способствовать развитию критического, системного, алгоритмического и творческого мышления;
- ✓ развивать внимание, память, наблюдательность; познавательный интерес;
- ✓ развивать умение работать с компьютерными программами и дополнительными источниками информации;
- ✓ развивать навыки планирования проекта, умение работать в группе.

### ***Воспитательные задачи***

- ✓ формировать положительное отношение к информатике и ИКТ;
- ✓ развивать самостоятельность и формировать умение работать в паре, малой группе, коллективе;
- ✓ формировать умение демонстрировать результаты своей работы.

Реализация учебной программы возможна в формате очного обучения, а также очно-заочного с применением дистанционных образовательных технологий (ДОТ). При очном обучении слушатели посещают аудиторные занятия, прослушивают лекции и выполняют практические задания по темам совместно с тренером и самостоятельно в рамках аудиторной работы.

Очно-заочное обучение с применением дистанционных технологий проходит в формате веб-занятия (вебинары, дистанционные уроки/курсы). В формате вебинаров прослушивается лекционный материал, практические задания выполняются параллельно с тренером (при наличии технической возможности), разбираются вопросы в режиме реального времени.

Промежуточная оценка качества освоения программы осуществляется путем устного опроса в ходе занятий, выполнения заданий (очное обучение, вебинар). Итоговая оценка осуществляется в виде тестирования. По окончании обучения в зависимости от результатов итоговых работ (итоговое тестирование, участие в проектной деятельности) слушателям выдается сертификат либо справка.

## **2. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ**

### **ЛИЧНОСТНЫЕ, МЕТАПРЕДМЕТНЫЕ И ПРЕДМЕТНЫЕ ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ**

**У обучающихся будут сформированы универсальные учебные действия, а именно:**

- **Личностные универсальные учебные действия:**

У ученика будут сформированы:

- ответственное отношение к информации с учетом правовых и этических аспектов ее распространения;
- развитие чувства личной ответственности за качество окружающей информационной среды;
- способность увязать учебное содержание с собственным жизненным опытом, понять значимость подготовки в области лего-конструирования и робототехники в условиях развивающегося общества
- готовность к повышению своего образовательного уровня;
- способность и готовность к принятию ценностей здорового образа жизни за счет знания основных гигиенических, эргономических и технических условий безопасной эксплуатации средств лего-конструирования и робототехники.

### **Метапредметные результаты:**

- **Регулятивные универсальные учебные действия:**

Ученик научится:

- принимать и сохранять учебную задачу;
- учитывать выделенные учителем ориентиры действия в новом учебном материале в сотрудничестве с учителем;
- планировать свои действия в соответствии с поставленной задачей и условиями её реализации, в том числе во внутреннем плане;
- адекватно воспринимать предложения и оценку учителей, товарищей, родителей и других людей;
- вносить необходимые коррективы в действие после его завершения на основе его оценки и учёта характера сделанных ошибок, использовать предложения и оценки для создания нового, более совершенного результата, использовать запись (фиксацию) в цифровой форме хода и результатов решения задачи, собственной звучащей речи на русском, родном и иностранном языках;

- **Познавательные универсальные учебные действия:**

Ученик научится:

- осуществлять поиск необходимой информации для выполнения учебных заданий с использованием учебной литературы, энциклопедий, справочников (включая электронные, цифровые), в открытом информационном пространстве, в том числе контролируемом пространстве Интернета;
- строить сообщения в устной и письменной форме;
- осуществлять анализ объектов с выделением существенных и несущественных признаков;

- **Коммуникативные универсальные учебные действия:**

Ученик научится:

- адекватно использовать коммуникативные, прежде всего речевые, средства для решения различных коммуникативных задач, строить монологическое высказывание, владеть диалогической формой коммуникации;
- допускать возможность существования у людей различных точек зрения, в том числе не совпадающих с его собственной, и ориентироваться на позицию партнёра в общении и взаимодействии;



- учитывать разные мнения и стремиться к координации различных позиций в сотрудничестве;
- формулировать собственное мнение и позицию;
- договариваться и приходить к общему решению в совместной деятельности, в том числе в ситуации столкновения интересов.

**Предметные результаты: знания, умения, владение:**

Проявление технического мышления, познавательной деятельности, творческой инициативы, самостоятельности;

Использование имеющегося технического обеспечения для решения поставленных задач;

Способность творчески решать технические задачи;

Способность продуктивно использовать техническую литературу для поиска сложных решений;

Способность самостоятельно планировать пути достижения поставленных целей;

Готовность выбора наиболее эффективных способов решения задач в зависимости от конкретных условий;

Самостоятельное создание алгоритмов деятельности при решении проблем творческого и поискового характера;

Готовность и способность создания новых моделей, систем;

Способность создания практически значимых объектов;

Способность излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений.

**3. УЧЕБНЫЙ ПЛАН**

№ п/п	Наименование тем	Всего часов	в том числе:		Формы контроля
			теория	практика	
1.	Программирование на Scratch	57	32	41	выполнение практических заданий
2.	Работа над проектом	10	2	8	защита проекта
3.	Итоговое тестирование	1	1		тестирование
<b>Итого часов:</b>		<b>68</b>	<b>35</b>	<b>49</b>	

**ТЕМАТИЧЕСКИЙ ПЛАН**

№ п/п	Наименование тем	Всего часов	в том числе:		Формы контроля
			теория	практика	
<b>1</b>	<b>Программирование на Scratch</b>	<b>57</b>	<b>18</b>	<b>39</b>	
1.1	Вводное занятие.	1	1	0	

	Что такое программирование				
1.2	Начало работы со Scratch	1	1	0	
1.3	Начинаем работать с программой. Блок «Движение». Сцена	1	1	0	выполнение практических заданий
1.4	Что такое алгоритм?	1	1	0	
1.5	Блок «События»	1	0	1	
1.6	Работа с объектами. Простые циклы. Блок «Внешность»	4	0	4	
1.7	Создание нового спрайта	1	0	1	
1.8	Вложенные циклы	2	0	2	
1.9	Переменные. Блок «Данные»	3	1	2	
1.10	Вычисления. Блок «Операторы»	3	1	2	
1.11	Строки и списки. Блок «Сенсоры»	4	1	3	
1.12	Обмен сообщениями	4	2	2	
1.13	Координаты	4	2	2	
1.14	Блок «Перо». Визуализация движения	4	2	2	
1.15	Блок «Звуки». Музыка в Scratch	6	2	4	
1.16	Сложные циклы. Блок «Управление»	3	1	2	
1.17	Создание блоков	2	0	2	
1.18	Логика. Блок «Операторы»	2	0	2	
1.19	Условные операторы. Блок «Управление»	8	2	6	
1.20	Scratch 3.0. Работа на онлайн-платформе	2	0	2	

2	Работа над проектом	10	2	8	защита проекта
3	Итоговое тестирование	1	1	0	тестирование
	Всего часов:	68	21	47	

## 4. СОДЕРЖАНИЕ ПРОГРАММЫ

### 1 Программирование на Scratch

#### 1.1 Вводное занятие. Что такое программирование

Презентация учебного курса, его цели и задачи. Обзор учебного пособия курса. Описание структуры будущих занятий, основных этапов работы. Презентация и демонстрация работы всего оборудования лаборатории. Инструктаж по технике безопасности при работе в лаборатории и правила поведения.

#### 1.2 Начало работы со Scratch

Что такое Scratch? Установка и запуск программы. Карта и интерфейс ScratchDuino 2.0.

#### 1.3 Начинаем работать с программой. Блок «Движение». Сцена

Понятие спрайта, цветные блоки, логика программы. Понятие проекта. Запуск ScratchDuino. Робоплатформы.

#### 1.4 Что такое алгоритм?

Понятия алгоритма, исполнителя, программы. Виды алгоритмов и построение простейших блок-схем.

#### 1.5 Блок «События»

Запуск скрипта, клик мышкой, нажатие клавиш.

#### 1.6 Работа с объектами. Простые циклы. Блок «Внешность»

Понятие объекта, объектно-ориентированный подход к разработке кода. Блок-схемы простых циклов. Организация движения Робоплатформы.

#### 1.7 Создание нового спрайта

Рисование собственного спрайта. Изменение существующих спрайтов. Импорт спрайта из проекта в проект. Добавление спрайта из файла.

#### 1.8 Вложенные циклы

Понятие вложенного цикла. Обоснованное использование вложенного цикла в коде программы.

#### 1.9 Переменные. Блок «Данные»

Для чего нужны переменные, что это такое. Где хранятся переменные.

#### 1.10 Вычисления. Блок «Операторы»

Все типовые математические операции. Понятие случайного числа.

#### 1.11 Строки и списки. Блок «Сенсоры»

Определение строк, разбор понятия массива как списка строк. Работа со списками, удаление, редактирование, добавлением элементов в списке.

Разбиение строк на подстроки.

### **1.12 Обмен сообщениями**

Блоки «События» и «Внешность». Процесс ввода-вывода сообщений. Блоки «Сказать», «Передать сообщение и ждать». Обработка дальнейших ответов.

### **1.13 Координаты**

Понятие системы координат. Перемещение спрайтов. События, связанные с изменением координат. Работа с робоплатформой.

### **1.14 Блок «Перо». Визуализация движения**

Привязка объекта к координатам, написание траекторий движения, изменение спрайтов по ходу движения объекта.

### **1.15 Блок «Звуки». Музыка в Scratch**

Озвучка действий спрайтов. Полная озвучка кода программы. Запись звука для проекта. Создание собственной музыки.

### **1.16 Сложные циклы. Блок «управление»**

Цикл с условием выхода «повторять пока не». Подробное рассмотрение случаев его применения. Переход по меткам. Команда «стоп» из блока «управления».

### **1.17 Создание блоков**

Определение блоков как замены функций. Создание простейшего блока, для подсчета координат и пройденного пути

### **1.18 Логика. Блок «Операторы»**

Булева алгебра. Сравнение показателей датчиков. Простое сравнение чисел.

### **1.19 Условные операторы. Блок «Управление»**

Понятие полного и неполного ветвления. Ветвление программы при разных показаниях датчиков. Вложенные условные блоки.

### **1.20 Scratch 3.0. Работа на онлайн-платформе**

Знакомство с онлайн-средой. Интерфейс среды. Создание ремейков и собственных проектов.

## **2. Работа над проектом**

Понятия проект, проектная деятельность, проектная группа. Особенности проектной деятельности. Этапы проектной деятельности. Работа над проектом и его защита.

## **3. Итоговое тестирование**

Теоретические вопросы по изученной программе.

## **5. Комплекс условий реализации программы.**

### **Календарные периоды учебного года**

1. Дата начала учебного года: 1 сентября 2023 г.

2. Дата окончания учебного года (5–8-е классы) 29 мая 2024г.;

3. Продолжительность учебного года:

– 5–8-е классы – 34 недели;

**Периоды образовательной деятельности**  
**Продолжительность учебных занятий по четвертям в учебных неделях и**  
**рабочих днях**  
**5–8-е классы**

Учебный период	Дата		Продолжительность	
	Начало	Окончание	Количество учебных недель	Количество рабочих дней
I четверть	01.09.2023	27.10.2023	8	41
II четверть	07.11.2023	29.12.2023	8	39
III четверть	09.01.2024	22.03.2024	11	55
IV четверть	01.04.2024	29.05.2024	7	35
<b>Итого в учебном году</b>			<b>34</b>	<b>170</b>

### Воспитательная работа

Большое внимание уделяется воспитательным мероприятиям. Данная работа осуществляется в соответствии с рабочей воспитательной программой школы. Основные направления воспитательной работы:

формирование и развитие творческих способностей обучающихся, выявление и поддержка талантливых детей и молодежи, духовно – нравственное, гражданско – патриотическое воспитание, возрождение семейных ценностей, формирование общей культуры обучающихся, социализация, самоопределения и профессиональная ориентация; формирование культуры здорового и безопасного образа жизни и комплексной профилактической работы;

формирование и развитие информационной культуры и информационной грамотности.

Цель воспитания — формирование, обогащение научными знаниями, совершенствование воспитательной деятельности, способствующей развитию нравственной, физически здоровой личности, способной к творчеству и самоопределению.

- направленность на сотрудничество с людьми, оказание помощи и поддержки окружающим, ответственности за общее дело;

- направленность на формирование коммуникативной компетентности, способности к эффективному межличностному взаимодействию, совместной работе в коллективе и группе;
- направленность на формирование высокой и устойчивой самооценки, чувства собственного достоинства

**Основные виды воспитательной работы:**

- годовой круг традиционных школьных мероприятий
- участие в мероприятиях разного уровня
- нравственно-правовая и гражданско-патриотическая деятельность;
- художественная деятельность и эстетическое воспитание;
- коллективные творческие дела;
- трудовая деятельность;
- связь с социумом;

**План работы**

<b>Направление воспитательной работы</b>	<b>Задачи работы по данному направлению</b>
Гражданское воспитание (в течении учебного года)	1) Формировать у учащихся такие качества, как долг, ответственность, честь, достоинство, личность. 2) Развивать у учащихся качества: активность, самостоятельность, инициатива
Экологическое воспитание (в течении учебного года)	1) Изучение учащимися природы и истории родного края. 2) Формировать правильное отношение к окружающей среде. 3) Содействие в проведении исследовательской работы учащихся. 4) Проведение природоохранных акций.
Духовно-нравственное, культурное наследие (в течении учебного года)	1) Формировать у учащихся такие качества как: культура поведения, эстетический вкус, уважение личности. 2) Создание условий для развития у учащихся творческих способностей.
Патриотическое воспитание (в течении учебного года)	1. Воспитывать любовь и уважение к традициям Отечества, школы, семьи.
формирование культуры здоровья	1) Формировать у учащихся культуру сохранения и совершенствования собственного здоровья.

(в течении учебного года)	2) Пропаганда здорового образа жизни
Популяризация научных знаний (в течении учебного года)	1) Развивать познавательную активность и творческие способности; 2) учить работать с научной литературой, отбирать, анализировать, систематизировать информацию, выявлять и формулировать исследовательские проблемы, грамотно оформлять научную работу; 3) способствовать овладению учащимися искусством дискуссии, выступления перед аудиторией с докладами

### **Традиционные школьные воспитательные мероприятия:**

- Осенний поход (викторина на языке программирования)
- День пожилого человека (поздравительная анимация на языке программирования)
- День учителя; (поздравительная анимация на языке программирования)
- День матери (поздравительная анимация на языке программирования)
- Новогодние праздники (поздравительная анимация на языке программирования)
- День защитника Отечества (викторина на языке программирования)
- 8 марта (поздравительная анимация на языке программирования)
- День победы (викторина на языке программирования)
- День Защиты детей (разработка интеллектуальных мини игр на языке программирования)

### **Материально-техническое и учебно-методическое обеспечение программы внеурочной деятельности.**

#### **Оснащение кабинета**

Персональный компьютер

Проектор

Интерактивная доска

Принтер

Телекоммуникационный блок, устройства, обеспечивающие подключение к сети

Устройства ввода-вывода звуковой информации Колонки Наушники микрофон.

#### **Программные средства**

Операционная система.

Файловый менеджер.  
Антивирусная программа.  
Программа-архиватор.  
Текстовый редактор, растровый и векторный графические редакторы.  
Программа разработки презентаций.  
Программа Scratch 3.0

## СПИСОК ЛИТЕРАТУРЫ

### Список литературы, используемой педагогом

1. Авторская программа курса по выбору «Творческие задания в среде программирования Scratch» Ю.В.Пашковской 5-6 классы, которая входит в сборник «Информатика. Программы для образовательных организаций: 2-11 классы» / составитель М.Н. Бородин. – М.: БИНОМ. Лаборатория знаний, 2015.
2. Иллюстрированное руководство по языкам Scratch и Python «Программирование для детей»/К. Вордерман, Дж.Вудкок, Ш.Макаманус и др.; пер. с англ.С.Ломакин. – М.:Манн, Иванов и Фербер, 2015.
3. Т.Е. Сорокина, поурочные разработки «Пропедевтика программирования со Scratch» для 5-го класса, 2015 г.
4. Учебно-методическое пособие. Проектная деятельность школьника в среде программирования Scratch. /В.Г. Рындак, В.О. Дженжер, Л.В. Денисова. - Оренбург - 2009
5. <http://scratch.mit.edu/pages/source> – страница разработчиков.
6. <http://scratch.mit.edu/> - официальный сайт проекта Scratch.
7. <http://setilab.ru/scratch/category/commun/> Сайт «Учитесь со Scratch»
8. [http://minecraftnavideo.ru/play/vd20J2r5wUQ/scratch\\_lesson\\_01\\_znakomstvo\\_so\\_sredoj\\_programmirovaniya\\_scratch.html](http://minecraftnavideo.ru/play/vd20J2r5wUQ/scratch_lesson_01_znakomstvo_so_sredoj_programmirovaniya_scratch.html)



## Список литературы, рекомендованной обучающимся

1. Первый шаг в робототехнику: практикум для 5–6 классов / Д. Г. Копосов / М.: БИНОМ. Лаборатория знаний, 2012
2. <https://scratch.mit.edu/> – web сайт Scratch
3. <http://robot.edu54.ru/> - Портал «Образовательная робототехника»

## Приложения

### Примеры проектов

#### 1. Научи кота бегать и мяукать.

**Исполнитель Scratch, цвет и размер пера.**

Понятие ИСПОЛНИТЕЛЯ.

Команды управления пером: «Опустить перо», «Поднять перо», «Очистить», «Установить цвет пера», «Установить размер пера».

#### 2. Рисуем разноцветные лужи и облака для прогулки кота.

**Основные инструменты встроенного графического редактора программной среды SCRATCH.**

Инструменты растрового графического редактора — кисточка, ластик, заливка, квадрат, круг, линия.

Копирование, поворот, горизонтальное отражение, вертикальное отражение во встроенном редакторе программной среды Scratch.

Командный блок внешность (фиолетовый) — команды начальной установки эффекта цвет «Установить эффект цвет в значение 0 и команда начальной установки размера «Установить размер 100%».

Команды: «Изменить цвет эффект на \_», «Изменить размер на \_»

Командный блок управления пером (зелёный) — команда «Печать» для копирования графического изображения исполнителя в нужном месте экрана. Эффекты, которые могут быть применены к графическим изображениям действующего исполнителя.

В графическом редакторе можно создавать сложные графические

изображения, почти не прикладывая собственных усилий, а лишь правильно применяя встроенные возможности программной среды.

### **3. Свободное рисование**

Придумай, чем можно дополнить проект с прошлого урока (Нарисовать дерево? Воздушный шар? Дом?) и реализуй это.

### **4. Запиши мой алгоритм!**

Ребята разбиваются на пары, в паре формулируют друг другу какую-то повседневную задачу, для решения которой нужен линейный алгоритм, придумывают этот алгоритм и рисуют блок-схему для задачи, заданной им напарником.

#### **Линейный алгоритм. Рисование линий исполнителем Scratch.**

Решение поставленной задачи в виде последовательного выполнения команд. Создание блок-схемы линейного алгоритма средствами редактора векторной графики.

Последовательное выполнение команд.

Изменение параметров пера.

#### **Что бывает полосатое?**

Придумайте линейный алгоритм для создания полосатого объекта (заранее придумайте, что это будет – шарф, зебра, забор?) Создайте блок-схему этого алгоритма и реализуйте алгоритм в среде Scratch.

#### **Линейный алгоритм. Исполнитель Scratch рисует квадраты и прямоугольники линейно.**

Создание алгоритма для рисования исполнителем квадрата путем последовательного выполнения команд.

Рисование линейного алгоритма, состоящего из двух колонок блоков команд.

Использование векторного редактора офисного пакета LibreOffice в качестве инструмента для создания блок-схем.

Выбор нужного значения из предлагаемого списка вариантов.

Отладка программы для получения верного результата.

Команда «повернуть в направление».

Пошаговое выполнение программы для её отладки.

Центр костюма исполнителя Scratch.

### **5. Конечный цикл. Scratch рисует квадраты, линии.**

Сохранение готовых программ для дальнейшего использования.

Рисование блок-схемы циклического алгоритма.

Использование векторного редактора офисного пакета LibreOffice в качестве инструмента для создания блок-схем;

Использование команд поворота на прямой угол (90°) по часовой и против часовой стрелки;

Использование циклического алгоритма для рисования исполнителем квадрата;

Оптимизация линейного алгоритма за счёт использования циклической конструкции в программной среде Scratch;

Команда открыть... из пункта меню File;  
Команда сохранить как... из пункта меню File;  
Циклический алгоритм;  
Блок-схема циклического алгоритма;  
Команды: «повернуться на 90° по часовой стрелке»,  
«повернуться на 90° против часовой стрелки», «повторить \_».  
Упражнение. Сравнение двух алгоритмов, рисующих квадрат.  
Идти 100 шагов Повторить 4  
Повернуться на 90° Идти 100 шагов  
Идти 100 шагов Повернуться на 90°  
Повернуться на 90°

Идти 100 шагов

Повернуться на 90°

Идти 100 шагов

Повернуться на 90°

Рекомендуется объяснить, что команды, обеспечивающие повторяющееся выполнение одной или нескольких команд называются циклическими, показать, что использование команды цикла всегда сокращает количество команд в программе, а значит делает её более красивой. В более короткой программе легче обнаружить ошибку или неточность.

## **6. Конечный цикл. Scratch рисует несколько линий и фигур.**

### **Копирование фрагментов программы.**

Использование операции цикла для решения учебных задач.

Применение поворота на прямой угол (90°) при создании геометрических фигур и перемещении исполнителя.

Оптимизация линейного алгоритма за счёт использования циклической конструкции в программной среде Scratch.

Тело цикла.

Конечный и бесконечный циклы.

Блок-схема бесконечного цикла.

Имя спрайта и костюма.

Изменение костюма исполнителя.

Копирование фрагмента программы.

Команды: «Следующий костюм», «Перейти к костюму».

Рекомендуется объяснить, что циклические конструкции могут применяться не только для рисования фигур. Например, в цикле можно менять внешность исполнителя.

### **Создай картинку из квадратов.**

Предложите ребятам нарисовать свою картинку из квадратов, может быть, это будет лестница, может быть, ковер, может быть, что-то еще.

## **7. Циклический алгоритм. Цикл в цикле.**

Использование операции цикла в цикле для решения учебных задач.

Использование поворота на прямой угол (90°) по часовой и против часовой стрелки.

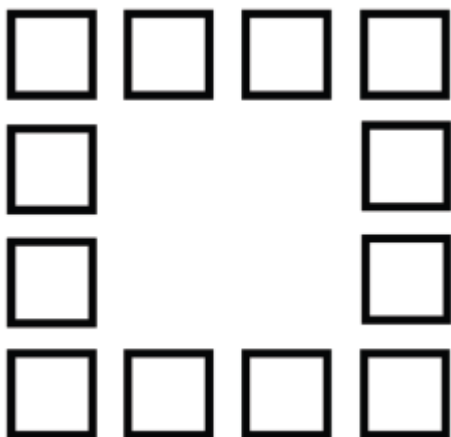
Оптимизация алгоритма за счёт использования конструкции «цикл в цикле».

Блок-схема конструкции цикл в цикле.

Например, расписание уроков составляется на неделю, в месяце в среднем четыре недели. Если расписание не меняется в течение учебного года, то получаем цикл из девяти месяцев по четыре недели с одинаковым расписанием.

Упражнение. Использование циклов «повторить \_»

Нарисуйте квадраты, как показано на рисунке, используя конструкцию цикл в цикле.



## **8. Цикл в цикле. Повторение пунктирной линии с поворотом. Блок-схема цикла.**

Использование конструкции «цикл в цикле».

Создание и реализация алгоритма рисования квадрата несплошными линиями;

Оптимизация алгоритма за счёт использования конструкции «цикл в цикле» в программной среде Scratch;

Использование операции копирования внешности исполнителя путём копирования костюма.

Несплошные линии.

Алгоритм рисования несплошных линий с использованием циклических конструкций.

Рисование квадрата несплошными линиями, используя конструкцию «цикл в цикле».

При перемещении исполнителя можно рисовать линии отличные от сплошной. Эти возможности определяются использованием команд из блока ПЕРО.

### **Мой необычный дом.**

Предложите ребятам нарисовать дом и что-то рядом с ним, используя линейные и циклические алгоритмы, а также смену костюмов Исполнителя. Может быть, над домом будут звезды разного размера, а около дома будет пунктирная разноцветная дорога, или деревья разного размера.

### **Бесконечный цикл. Анимация исполнителя Scratch на основе готовых костюмов.**

Бесконечный цикл.

Анимация исполнителя с помощью смены костюмов.

Эффект «призрак».

Изменение размера исполнителя.

Интерактивное взаимодействие с исполнителем с помощью клавиатуры.

Команды «спросить» и «думать».

**Сцена как исполнитель. Создаем модель таймера.**

Анимация сцены (фона).

Синхронная анимация сцены и исполнителя.

## **10. Одинаковые программы для нескольких исполнителей.**

Анимация с помощью вращения.

Дублирование исполнителей.

При копировании исполнителей копируются и их скрипты.

Использование одинаковых программ, но разных костюмов у исполнителей.

**Параллельное выполнение действий несколькими исполнителями**

Дублирование исполнителей.

Синхронное выполнение скриптов.

При параллельном выполнении скриптов результат работы всех исполнителей получается гораздо быстрее, чем в тех задачах, когда работал один исполнитель.

## **11. Разбиение программы на части для параллельного выполнения исполнителями. Таймер.**

Планирование действий разных исполнителей во времени для решения общей задачи.

Сенсор «таймер»

Сброс «таймера»

Команда «ждать до ...»

**Мини-проект «Смена времени суток».**

Создайте скрипт, в котором будет четыре сцены: «утро», «день», «вечер», «ночь». Используйте «таймер» для своевременного изменения сцен, своевременного запуска скриптов разных исполнителей.

**Два исполнителя со своими программами. Мини-проект «Часы с кукушкой».**

Координаты в Scratch.

Команда «идти в x: \_ y: \_» (переместиться в точку с координатами (x,y))

Команда «плыть \_ секунд в точку x: \_ y: \_» (плавно двигаться в точку с координатами (x,y))

Команды «спрятаться», «показаться».

Кейс 9. Мини-проект «Часы с кукушкой».

Создайте сцену в виде часов, и три исполнителя: часовую стрелку, минутную стрелку и кукушку. Стрелки должны вращаться на циферблате. При наступлении каждого часа (т.е. в \_ часов : 00 минут) должна появляться кукушка.

## **12. Алгоритмы с ветвлением. Условие ЕСЛИ.**

Еще из сказок мы знаем, что на камне было написано: «Налево пойдешь – коня

потеряешь, направо пойдешь – себя не найдешь, прямо пойдешь – в сказку попадешь»

Блок-схема алгоритма с ветвлением.

Полная форма ветвления. Команда «если \_ или»

Неполная форма ветвления. Команда «если \_»

Сенсор «касается» (касания края, или другого спрайта, или указателя мыши)

### **13. Цикл с условием. Мини-проект «Шарики в лабиринте»**

Сенсор «касается цвета»

Программируем отскок шариков от стен

#### **Мини-проект «Шарики в лабиринте»**

Создайте сцену в виде бильярдного стола, на котором три шарика будут двигаться с разными скоростями и отталкиваться от бортиков.

**Исполнитель определяет цвет.**

Сенсор «касается цвета»

Программируем поведение исполнителя в зависимости от цвета фона.

### **14. Оператор случайных чисел.**

Команда «Выдать случайное число от \_ до \_». Случайные перемещения исполнителя в координатной плоскости.

Случайное количество шагов.

Случайные координаты.

Поворот на случайный угол.

Команда «Перейти в верхний слой»

Команда «Перейти назад на \_ слоев»

Эффект «Призрак»

### **15. Взаимодействие исполнителей.**

Исполнители касаются друг друга.

Команда «Касается \_»

Поведение исполнителей при столкновении.

#### **Последовательное выполнение команд исполнителями.**

Команда «Передать»

Команда «Когда я получу»

Связи между программами разных исполнителей.

### **16. Программирование клавиш. Мини-проект «Лабиринт»**

Взаимодействие пользователя с программой.

Клавиши управления перемещением исполнителя

Команда «Изменить значение x на \_»

Команда «Изменить значение y на \_»

#### **Игра «Лабиринт»**

Создайте сцену в виде лабиринта. С помощью клавиш со стрелками

управляйте движением маленькой мышки. Мышка должна добраться до выхода, не задевая стенок лабиринта. В случае касания стенок мышка возвращается в исходную точку.

### **17. Координатная плоскость. Геометрические фигуры.**

Рисование геометрических фигур на координатной плоскости.

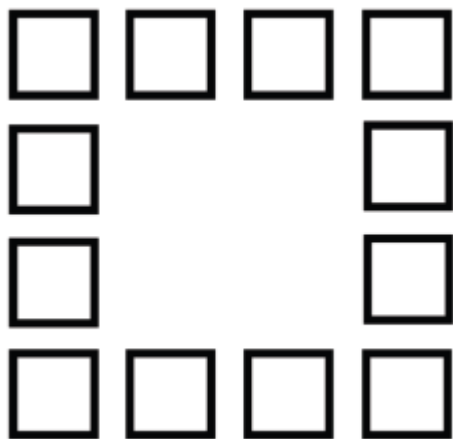
Последовательное выполнение команд одним исполнителем.

Параллельное выполнение команд несколькими исполнителями.

#### **Исследование времени выполнения программ.**

Создайте одного исполнителя, рисующего квадраты, как показано на рисунке.

Затем создайте четыре исполнителя, которые будут параллельно рисовать по три квадрата. Сравните время работы в этих двух случаях.



### **18. Координатная плоскость. Переменные.**

Блок «Переменные»

Рисование геометрических фигур на координатной плоскости с использованием переменных.

Построение перпендикуляров к координатным осям.

#### **Создание списков.**

Название списка.

Элементы списка.

Длина списка.

Команда «Создать список».

Выбор элемента списка.

### **19. Мини-проект «Викторина».**

Создайте проект «Викторина» в котором будет задано 10 вопросов.

Правильный ответ надо выбрать из списка. В случае правильного ответа очки увеличиваются на единицу. В конце выставляется оценка: «отлично» за 9 или 10, «хорошо» за 7 или 8, «удовлетворительно» за 5 или 6 правильных ответов.