

Управление образования администрации  
муниципального образования Кандалакшский район  
Муниципальное автономное учреждение дополнительного образования  
«Детско-юношеский центр «Ровесник»  
имени Светланы Алексеевны Крыловой»  
муниципального образования Кандалакшский район

**ПРИНЯТА**  
педагогическим советом  
от 30.05.2023 г.  
Протокол № 6

**УТВЕРЖДЕНА**  
приказом директора  
от 31.05.2023 г. № 84  
Директор  О.Ю. Савенкова



Дополнительная общеобразовательная общеразвивающая программа  
технической направленности  
**«Школа программирования»**  
Возраст обучающихся: 13-17 лет  
Срок реализации программы: 2 года  
Уровень сложности: базовый

Автор-составитель:  
Забродин Павел Викторович,  
педагог дополнительного  
образования

г. Кандалакша, 2023

## Пояснительная записка

к дополнительной общеобразовательной общеразвивающей программе  
технической направленности  
«Школа программирования»

Настоящая дополнительная общеобразовательная общеразвивающая программа «Школа программирования» разработана с учетом:

- Федерального Закона Российской Федерации от 29.12.2012 г. № 273 «Об образовании в Российской Федерации»;
- Распоряжения Правительства Российской Федерации от 31.03.2022 г. № 678-р «Концепция развития дополнительного образования детей до 2030 года»;
- Приказа Министерства просвещения Российской Федерации от 27.07.2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- Приказа Министерства просвещения Российской Федерации от 03.09.2019 г. № 467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей»;
- Постановления Главного государственного санитарного врача Российской Федерации от 28.09.2020 г. № 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»;
- Письма Министерства образования и науки Российской Федерации от 18.11.2015 г. № 09-3242 «Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)»;
- Устава МАУДО ДЮЦ «Ровесник» им. С.А. Крыловой.

Дополнительная общеобразовательная общеразвивающая программа «Школа программирования» имеет техническую направленность составлена на основе:

– сборника примерных дополнительных общеразвивающих программ государственного бюджетного образовательного учреждения г.Москвы дополнительного образования (повышения квалификации) специалистов Городского методического центра Департамента образования г. Москвы, М.: 2019

– авторской программы С.М. Шпыневой «Технологии SCRATCH» МАОУ «Центр образования №13 имени Героя Советского Союза Н.А. Кузнецова», Тамбов, 2014 год

– «Программирование на языке Pascal» Мыльниковой Г.А., Пеньковой Н.А., Сиражитдиновой О.А. г. Приозерск, 2009 г.

**Вид программы:** дополнительная общеобразовательная общеразвивающая программа.

**Актуальность программы.** Действительно ли всем нужно учиться программированию? Еще в 1981 году на 3-й Всемирной конференции Международной федерации по обработке информации и ЮНЕСКО по применению ЭВМ в обучении в Лозанне (Швейцария) Андрей Ершов, советский ученый, один из пионеров теоретического и системного программирования, выступает с [докладом](#) под названием «Программирование — вторая грамотность». В своем выступлении, проводя параллели между распространением книгопечатания и развитием технологий, он последовательно делает вывод: «если развитие и распространение книгопечатания привело к всеобщей грамотности, то развитие и распространение ЭВМ приведет ко всеобщему умению программировать».

IT-сфера в настоящее время активно развивается, а программисты и аналитики - одни из самых востребованных специалистов на рынке труда. Но всё чаще программирование осваивают и представители других профессий: те, кто пишут тексты, управляют продуктами, рисуют, ведут собственные блоги. Например, в Университете ИТМО с сентября 2018 года все студенты-бакалавры и магистранты нового набора вне зависимости от направления подготовки и образовательной программы будут изучать цифровую культуру. В рамках нового направления студенты вне зависимости от того, имеют ли они профильное образование в области IT, будут получать компетенции, позволяющие им использовать инструменты, которые предлагают современные информационные технологии.

Между тем, спрос на навыки в области программирования продолжает расти год от года. По данным аналитиков платформы Courseburg, больше всего в российском сегменте интернета пользователи ищут курсы Java, PHP, JavaScript и Python. Так зачем изучать программирование тем, кто не планирует стать профессиональным IT-шником?

1. Это свобода творчества. Многие думают, что для программирования нужны пятёрки по математике. Это заблуждение. Программирование больше про логику и творчество. Например, ты с друзьями в шутку придумал классную игру для смартфона. Можно, конечно, подождать, пока какая-нибудь компания выпустит её, а можно сделать это самому. Как эта игра будет работать, какие в ней будут правила, интерфейс и функции - решать тебе. С навыками программирования появляется возможность с нуля создать продукт или реализовать идею - приложение, сайт, программу - так, как это нужно именно тебе.

2. Это окупается. Простая истина: чем больше у тебя навыков, тем дороже твои услуги на рынке труда. Например, копирайтеры, которые могут сверстать простые html-страницы, бухгалтеры или менеджеры, которые могут автоматизировать работу [Excel](#) или Access с помощью кода, стоят «дороже», чем их коллеги только с базовым набором скиллов.

Крупные компании быстро развиваются и ищут при этом новые способы оптимизации. Нанять команду разработчиков и платить каждому по 2 000–3 000 долларов в месяц — дорогое удовольствие. Нанять человека, который будет писать статьи, верстать и ещё придумает, как улучшить мобильную версию

сайта, — выгодное решение. У таких специалистов больше шансов попасть в крупную компанию.

Например, в The New York Times уже более половины журналистов умеют программировать. Когда издание изменило стратегию и сделало акцент на развитие веб- и мобильной версии - обучили программированию сотрудников, которые отвечают за цифровой контент.

3. Это сближает. Чтобы делать совместные проекты, понимать, как ставить задачи, вносить правки и на каком вообще языке говорить с программистами, нужно самому хотя бы немного понимать код. Даже руководителям компаний и владельцам [стартапов](#) полезно научиться программировать. Например, чтобы найти на рынке хорошего специалиста, лично провести собеседование и адекватно оценить уровень кандидата.

4. Это экономит время. Сколько раз ты вздыхал, когда приходилось делать рутинную работу? Например, ты журналист и пишешь для научно-популярного издания. Тебе нужно изучить, в каких странах чаще всего молнии ударяют в землю, и понять, что общего у этих регионов. На это уйдёт не один день. Программирование ускорит этот процесс: ты потратишь 30–40 минут на код, который поможет быстро проанализировать данные и составить отчёт.

5. Это развивает полезные привычки. Программирование — кропотливый процесс. Здесь не получится почитать учебник на досуге, посмотреть видео на YouTube и сходить на пару лекций. Нужно сесть и начать учить язык, копать в коде и всё время практиковаться. Например, чтобы писать на Python, надо каждый день заниматься по 2–3 часа на протяжении нескольких месяцев. Это дисциплинирует и помогает мыслить структурно и стратегически.

**Педагогическая целесообразность программы.** Педагогическая целесообразность дополнительной общеобразовательной общеразвивающей программы технической направленности «Школа программирования» обусловлена тем, что ее содержание, используемые технологии, формы и методы обучения создают и обеспечивают необходимые условия для личностного развития и творческого труда обучающихся, носят пропедевтический характер изучения курса программирования на следующих этапах обучения. Программа обучения способствует предпрофессиональной ориентации подростков и их подготовке к получению специальности программиста, даёт возможность оценить свои перспективы в этой области. Основной упор при обучении делается на овладение обучающимися умением составлять алгоритмы, развитие логического мышления.

Также одной из составляющих педагогической целесообразности является метапредметность данного курса, полученные знания, умения и навыки по программе помогут каждому ребенку в его дальнейшей жизни, позволят успешно реализоваться в других областях знаний. Учащийся должен понять и принять то, что компьютер – это техническое устройство, которое может стать большим помощником в повседневной жизни и учебе.

**Отличительная особенность программы.** Обучение основам программирования на начальном этапе должно осуществляться на специальном языке программирования, который будет понятен обучающимся, будет легок для освоения и соответствовать современным направлениям в программировании. В

данном курсе предполагается начать изучение программирования в игровой, увлекательной форме. Scratch – это объектно-ориентированная среда, в которой блоки программ собираются из разноцветных кирпичиков. Scratch – это мультимедийная система. Большая часть операторов языка направлена на работу с графикой и звуком, создание анимационных и видеоэффектов. Манипуляции с медиаинформацией – главная цель создания Scratch. В процессе работы по данной программе формируется логическое мышление, алгоритмическое мышление, так же развиваются творческие способности учащихся.

Следующей ступенью знакомства с многообразным миром программирования является изучение классического языка программирования Паскаль. Именно чётко структурированный и достаточно простой язык учит именно программированию как процессу, а не зазубриванию служебных слов и их комбинаций. Ещё одной причиной изучения Паскаля является то, что лежит в основе других очень популярных сегодня языков программирования – Delphi, Object Pascal, Pascal Script, ST, который используется для создания программ для ПЛК (программируемый логический контроллер). Опытному программисту для изучения основ нового языка программирования потребуется всего несколько дней. То есть перестроиться с Delphi на Visual C++ или обратно не так уж и сложно (и не так уж и долго). Но для этого надо уметь программировать!

Посредством изучения программирования, обучающиеся учатся вариативно мыслить и рассуждать логически, быть дисциплинированным и использовать комплексный подход для решения задач; находятся в атмосфере постоянного поиска, включаются в работу коллектива, увлеченного решением задачи. Данная программа носит практическую направленность, и будет требовать самостоятельной работы подростков не только на занятиях, но и в свободное время.

**Новизна программы** заключается и в том, что включение детей в творческий процесс через постановку проблемы, а не сообщение темы занятия педагогом, позволяет повысить мотивацию деятельности учащихся, более осознанно воспринимать структуру занятия, определять цель, обнаруживать и формулировать проблему с помощью педагога и самостоятельно выдвигать версии, выбирать средства достижения цели с помощью педагога и в группе, планировать деятельность с помощью педагога и самостоятельно, а также работать по плану, т.е. способствует достижению нового образовательного результата.

**Уровень программы:** базовый.

**Адресат программы.** Программа рассчитана на детей в возрасте 13-17 лет: первый год обучения – 13-16 лет, второй год обучения – 14-17 лет.

**Объем и срок реализации программы.** Программа рассчитана на 2 года обучения: первый год обучения – 72 часа, второй год обучения – 144 часа.

**Режим занятий:** 1 год обучения – 1 раз в неделю по 2 часа, 2 год обучения – 2 раза в неделю по 2 часа. Продолжительность занятий – 45 минут. Режим занятий соответствует санитарно-эпидемиологическим требованиям к учреждениям дополнительного образования детей (СП 2.4. 3648-20, СанПиН 1.2.3685-21).

**Количество обучающихся:** 12 человек.

**Условия приема.** Набор свободный, осуществляется в соответствии с «Положением приема, перевода, отчисления обучающихся и комплектования объединений в Муниципальном автономном учреждении дополнительного образования «Детско-юношеский центр «Ровесник» имени Светланы Алексеевны Крыловой муниципального образования Кандалакшский район (утверждён приказом директора МАУДО «ДЮЦ «Ровесник» от 06.05.2020г. № 39/3).

Обучающиеся зачисляются в учебные группы при наличии заявления родителей (законных представителей).

**Форма обучения:** очная.

**Форма организации занятий:** фронтальная, групповая, индивидуальная.

**Форма обучения и виды занятий:**

Формы проведения учебных занятий в объединении подбираются с учетом возрастных особенностей детей, целей и задач образовательной программы, специфики предмета и других факторов. Наиболее подходящими для данного объединения являются следующие формы:

- практикумы;
- беседа с игровыми элементами;
- сюжетно-ролевые игры;
- игра-имитация;
- викторины, творческие конкурсы и задания,
- олимпиады по профилю обучения.

**Цель:** развитие интеллектуальных и творческих способностей учащихся, формирование и развитие алгоритмического мышления на основе изучения алгоритмов и программирования.

**Задачи:**

**Обучающие:**

- сформировать компетентности учащихся в области алгоритмизации и программирования;
- овладеть навыками составления алгоритмов;
- изучить функциональность работы основных алгоритмических конструкций;
- сформировать представление о профессии «программист»;
- сформировать навыки разработки программ в различных средах программирования;
- познакомить с понятием проекта и алгоритмом его разработки.

**Развивающие:**

- способствовать формированию навыков разработки проектов: интерактивных историй, квестов, интерактивных игр, обучающих программ, мультфильмов, моделей и интерактивных презентаций.
- 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 и достижения результативности проекта;

– способствовать формированию умения демонстрировать результаты своей работы.

**Воспитательные:**

- формировать положительное отношение к информатике и ИКТ;
- развивать самостоятельность и формировать умение работать в паре, малой группе, коллективе;
- мотивировать к участию в олимпиадах и конкурсах по информатике и программированию.

**Прогнозируемые результаты:**

Предметные:

- владение навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов;
- наличие представления о видах алгоритмов (линейный, разветвляющийся, циклический) и о способах их реализации в различных средах программирования;
- наличие представления об основных типах данных, их характеристиках, диапазонах допустимых значений и правилах описания/использования в различных средах программирования;
- понимание программ, написанных на выбранном для изучения универсальном алгоритмическом языке;
- знание основных языковых конструкций программирования;
- владение стандартными приемами написания на алгоритмическом языке программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ;
- владение навыком использования готовых прикладных компьютерных программ по выбранной специализации.

Метапредметные:

- умение самостоятельно планировать пути достижения целей, в том числе альтернативные, и осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;
- умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата, определять способы действий в рамках предложенных условий и требований, корректировать свои действия в соответствии с изменяющейся ситуацией;
- умение оценивать правильность выполнения учебной задачи, собственные возможности её решения;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;
- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
- умение организовать учебное сотрудничество и совместную деятельность с педагогом и сверстниками; работать индивидуально и в группе: находить общее решение и разрешать конфликты на основе согласования позиций и учёта интересов; формулировать, аргументировать и отстаивать своё мнение;

– формирование и развитие компетентности в области использования информационно-коммуникационных технологий (ИКТ-компетенций).

Личностные:

– проявление ответственного отношения к учению, готовности и стремления, обучающихся к саморазвитию и самообразованию на основе мотивации к обучению и познанию, осознанному выбору и построению дальнейшей индивидуальной траектории образования на базе ориентировки в мире профессий и профессиональных предпочтений, с учетом устойчивых познавательных интересов, а также на основе формирования уважительного отношения к труду, развития опыта участия в социально значимом труде;

– проявление коммуникативной компетенции в общении и сотрудничестве со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, общественно полезной, учебно-исследовательской, творческой и других видов деятельности.

В результате первого года обучения учащиеся будут **знать/владеть:**

– как правильно и безопасно вести в кабинете информатики и ВТ, работать с ПК/ноутбуком;

– интерфейс среды программирования Scratch;

– основными понятиями среды программирования Scratch: спрайт, костюм, сцена;

– принципы работы и параметры основных модулей, эффектов среды программирования Scratch;

– этапы работы над проектами (мультфильмами, играми);

– виды алгоритмов, способы описания (словесный, графический);

– возможности реализации алгоритмов в среде программирования Scratch.

**уметь:**

– устанавливать программу среды программирования Scratch и работать в режиме онлайн, оффлайн;

– использовать графические средства среды программирования Scratch для реализации различных видов алгоритмов (линейных, разветвляющихся, циклических) начального и базового уровня сложности;

– использовать среду программирования Scratch для создания мультфильмов и простейших игр;

– создавать, презентовать готовый продукт.

В результате второго года обучения учащиеся будут

**знать:**

– как правильно и безопасно вести в кабинете информатики и ВТ, работать с ПК/ноутбуком;

– понятия «исполнитель», «система команд исполнителя»;

– интерфейс среды программирования Pascal ABC, КУМИР;

– основными понятиями среды программирования Pascal ABC: типы данных, функции для работы с основными типами данных;



– принципы работы сред программирования Pascal ABC, КУМИР: создание программы/среды исполнителя, компиляция, трассировочные таблицы для проверки результатов работы программы;

– виды алгоритмов, способы описания (словесный, графический);

– возможности реализации алгоритмов в средах программирования Pascal ABC, КУМИР для различных исполнителей.

**уметь:**

– устанавливать программу среды программирования Pascal ABC;

– использовать программные средства среды программирования Pascal ABC для реализации различных видов алгоритмов (линейных, разветвляющихся, циклических, вложенных циклических структур) базового и повышенного уровня сложности;

– использовать среды программирования Pascal ABC, КУМИР для решения задач различного уровня сложности;

– оценивать эффективность использования алгоритма для решения задач.

### Учебный план первый год обучения

№ п/п	Название раздела, темы	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
1.	<b>Раздел 1. Введение</b>				
1.1	Первичный инструктаж. Знакомство с курсом обучения. История языков программирования.	1	1	-	Текущий контроль
1.2	Введение в среду программирования Scratch	2	1	1	Текущий контроль
1.3	Входное тестирование	1	-	1	Текущий контроль
	<b>ИТОГО</b>	<b>4</b>	<b>2</b>	<b>2</b>	
2.	<b>Раздел 2. Знакомство со средой Scratch</b>				
2.1	Интерфейс среды программирования Scratch. Линейный алгоритм	2	1	1	Текущий контроль
2.2	Спрайты. Библиотеки костюмов и сцен	2	1	1	Текущий контроль
2.3	Графический редактор Scratch	2	1	1	Текущий контроль
2.4	Добавление нового спрайта	2	1	1	Текущий контроль
2.5	Учимся управлять спрайтами	2	1	1	Текущий контроль

					контроль
2.6	Создание спецэффектов команды «Внешность» -	2	1	1	Текущий контроль
2.7	Знакомство с пером	2	1	1	Текущий контроль
2.8	Работа со звуком	2	1	1	Текущий контроль
2.9	Контрольная работа №1 «Знакомство со средой Scratch»	2	-	2	Текущий контроль
	<b>ИТОГО</b>	<b>18</b>	<b>8</b>	<b>10</b>	
<b>3.</b>	<b>Раздел 3. Знакомство с эффектами среды программирования Scratch</b>				
3.1	Цветовой эффект	2	1	1	Текущий контроль
3.2	Эффект рыбьего глаза	2	1	1	Текущий контроль
3.3	Эффект завихрения	2	1	1	Текущий контроль
3.4	Эффект укрупнения пикселей	2	1	1	Текущий контроль
3.5	Эффекты мозаики и яркости.	2	1	1	Текущий контроль
3.6	Эффект призрака	2	1	1	Текущий контроль
3.7	Практическая контрольная работа №2 «Эффекты среды программирования Scratch»	2	-	2	Текущий контроль
	<b>ИТОГО</b>	<b>14</b>	<b>6</b>	<b>8</b>	
<b>4.</b>	<b>Раздел 4. Математика в среде программирования Scratch</b>				
4.1	Знакомство с координатами x и y	4	1	3	Текущий контроль
4.2	Случайные числа	4	1	3	Текущий контроль
4.3	Кот математик	6	2	4	Текущий контроль
	<b>ИТОГО</b>	<b>14</b>	<b>4</b>	<b>10</b>	
<b>5.</b>	<b>Раздел 5. Команды управления в среде программирования Scratch</b>				
5.1	Ветвление. Реализация алгоритма ветвления в среде программирования Scratch	2	1	1	Текущий контроль
5.2	Цикл. Реализация циклического алгоритма в среде программирования Scratch	2	1	1	Текущий контроль

5.3	Реализация вложенных циклов в среде программирования Scratch	2	1	1	Текущий контроль
	<b>ИТОГО</b>	<b>6</b>	<b>3</b>	<b>3</b>	
<b>6.</b>	<b>Раздел 6. Создание мультфильмов в среде программирования Scratch</b>				
6.1	Этапы работы над проектом	2	1	1	Текущий контроль
6.2	Мультфильм «Глубокое синее море» по технологической карте	2	-	2	Текущий контроль
6.3	Мультфильм «Кот и летучая мышь» (частичное использование технологической карты)	2	-	2	Текущий контроль
6.4	Мультфильм «Лабиринт»	2	-	2	Текущий контроль
6.5	Мультфильм «Гига и приведение»	2	-	2	Текущий контроль
	<b>ИТОГО</b>	<b>10</b>	<b>1</b>	<b>9</b>	
<b>7.</b>	<b>Раздел 8. Мой первый проект в среде программирования Scratch</b>				
8.1	Создание мультфильма	6	1	5	Итоговый контроль (защита проекта)
	<b>Итого</b>	<b>6</b>	<b>1</b>	<b>5</b>	
	<b>Итого за год:</b>	<b>72</b>	<b>25</b>	<b>47</b>	

Содержание программы  
первый год обучения

**Раздел 1. Введение**

Тема 1.1. Первичный инструктаж. Знакомство с курсом обучения.

Теория (1 час):

- Режим труда и отдыха при работе на ПК, санитарные нормы времени.
- Защита от вредного воздействия ПК на состояние психики человека и его физическое состояние, меры профилактики.
- Рациональная организация труда и отдыха.
- Действие тока на организм; факторы, влияющие на исход поражения; мероприятия по защите от поражения электрическим током.
- Понятие о пожаре, горении; причины пожаров; средства пожаротушения
- Знакомство с курсом обучения.

Тема 1.2. Введение в среду программирования Scratch

Теория (1 час):

- Происхождение языка программирования, история развития, причины популярности

Практика (1 час):

- Установка программы и работа со средой в режимах онлайн и оффлайн

Тема 1.3. Входное тестирование.

Практика (1 час).

## **Раздел 2. Знакомство со средой Scratch**

Тема 2.1. Интерфейс среды программирования Scratch. Линейный алгоритм

Теория (1 час):

- Знакомство с окном среды программирования Scratch: режим переключения языка интерфейса; область Сцена, область Спрайтов область Скриптов. Основа Scratch: знакомство с цветовыми блоками
- Блок События: применение некоторых команд
- Блок Движение: применение некоторых команд
- Блок Звук, регулирование громкости, звуки из библиотеки

Практика (1 час):

- Составление программы (скрипта) «Бегущий котёнок», импорт в программу звука
- Сохранение проекта

Тема 2.2. Спрайты. Библиотеки костюмов и сцен

Теория (1 час):

- Спрайты. Информация спрайта. Что могут спрайты. Операции над спрайтами.
- Костюмы. Выбор костюмов.
- Создание декораций - фонов.
- Изменение внешнего вида спрайта и фона. Импорт изображений.
- Экспорт спрайтов и их использование в проектах.

Практика (1 час):

- Творческие задания для детей - создание исполнителей и фонов в проекте: «Круглые рисунки».
- Создание костюмов из встроенной библиотеки и интернета.

Тема 2.3. Графический редактор Scratch

Теория (1 час):

- Растровая и векторная графика. Особенности работы с растровым и векторным режимами графического редактора Scratch.
- Общие инструменты рисования. Обозначение центра изображения.
- Установка прозрачных цветов.
- Инструменты растрового и векторного режимов рисования.
- Слои изображения.
- Группировка фигур.

Практика (1 час):

- Рисуем в Scratch в растровом режиме: домик, кувшинку, кораблик, рыбку, робота.

- Рисуем в векторном режиме: машинку, монстра.
- Сохраняем спрайты для их использования в проектах.

#### Тема 2.4. Добавление нового спрайта

##### Теория (1 час):

- Знакомство с библиотекой спрайтов

##### Практика (1 час):

- Написание программы движения для нового персонажа
- Одновременное движение персонажей
- Настройка управления персонажами (клавиша пробел, стрелка и др.)

#### Тема 2.5. Учимся управлять спрайтами

##### Теория (1 час):

- Основные команды движения и события. Коричневые блоки - команды группы «События» - «Когда щёлкнут по флажку» и «Когда клавиша на клавиатуре нажата». Синие блоки - команды группы «Движение».
- Знакомство с отрицательными числами. Движение задом наперед.

##### Практика (1 час):

- Создание алгоритма первого проекта в среде Scratch. Анимация «Научим кота бегать», запуская скрипт разными событиями: клик мышкой, нажатие разных клавиш.
- Задание: «Автомобиль с пятью скоростями».

#### Тема 2.6. Создание спецэффектов - команды «Внешность»

##### Теория (1 час):

- Фиолетовые блоки - «Внешность» - блоки управления внешностью для спрайтов и сцены.
- Что такое анимация. Анимирование костюма. Эффекты изображений: учим спрайты исчезать и появляться - телепортация с помощью эффектов, изменять цвет и т. д.
- Размер и видимость. Диалоги - спрайты умеют думать и говорить.
- Слои.

##### Практика (1 час):

- Задания: «Осьминог», «Танцующая балерина», «Бегущий человек», «Оживляем динозавра», «Программируем страшилки» - спрайты неожиданно появляются и исчезают, как привидения

#### Тема 2.7. Знакомство с пером

##### Теория (1 час):

- Темно-зеленые блоки - группа «Перо». Спрайты умеют рисовать.
- Перьевая черепаха.
- Исследуем печать.

##### Практика (1 час):

- Игры с пером. Рисуем разные фигуры, буквы, лесенки, штрихпунктирные линии, спирали, цветные пятна, узоры из фигур.
- Задания: «Конфетти», «След от полёта», «Паук плетёт паутину».

#### Тема 2.8. Работа со звуком

##### Теория (1 час):

- Добавление звуков к спрайтам. Проигрывание звука. Запись звука. Как проигрывать аудиофайлы.
- Переворачиваем звуки. Игра на барабанах и другие звуки.
- Создание своей музыки. Полный звукоряд. Устанавливаем темп и такт.
- Контроль громкости звука. Ноты.
- Как рисуют ноты.

Практика (1 час):

- Задания: «Организуем ансамбль», «Танцы на сцене», «Фейерверки».
- Создаем песни: «Маленькой ёлочке», «Чижик-пыжик», «Гуси» и другие.

Тема 2.9. Контрольная работа №1 «Знакомство со средой Scratch»

Практика (2 часа):

- Создание простейшего мультфильма.

### **Раздел 3. Знакомство с эффектами среды программирования Scratch**

Тема 3.1. Цветовой эффект

Теория (1 час):

- Знакомство с понятием эффект, назначение эффектов, разумность использование
- Параметры использования эффекта Цвет, изменение цветов

Практика (1 час):

- Применение эффекта Цвет для изменения внешнего вида костюма

Тема 3.2. Эффект рыбьего глаза

Теория (1 час):

- Параметры использования эффекта Рыбий глаз – эффекта снятия широкоугольной камерой

Практика (1 час):

- Применение/исправление эффекта для изображений с сильной дисторсией

Тема 3.3. Эффект завихрения

Теория (1 час):

- Параметры использования эффекта Завихрение

Практика (1 час):

- Применение эффекта для спрайта

Тема 3.4. Эффект укрупнения пикселей

Теория (1 час):

- Знакомство с понятием «пиксель», размер пикселя
- Параметры использования эффекта Укрупнение пикселей

Практика (1 час):

- Применение эффекта для спрайта

Тема 3.5. Эффекты мозаики и яркости.

Теория (1 час):

- Параметры использования эффектов Мозаика и Яркость

Практика (1 час):

- Применение эффекта для спрайта

Тема 3.6. Эффект призрака (исчезания)

Теория (1 час):

- Параметры использования эффекта Призрака

Практика (1 час):

- Применение эффекта для спрайта

Тема 3.7. Практическая контрольная работа №2 «Эффекты среды программирования Scratch»

Практика (2 часа)

- Создание мультфильма с использованием различных эффектов спрайта.

#### **Раздел 4. Математика в среде программирования Scratch**

Тема 4.1. Знакомство с координатами x и y

Теория (1 час):

- Понятие «координатная плоскость», перемещение по координатной плоскости
- Новые блоки перемещения по координатной плоскости: изменить X, изменить Y: диапазон значений координаты Y в среде Scratch
- Понятие «сравнения», блок Сравнение

Практика (3 часа):

- Практикум «Рисование по координатам на плоскости»
- Создание программы вертикального движения вверх и вниз (управление клавишами) любого персонажа
- Создание программы «Взлёт ракеты»

Тема 4.2. Случайные числа

Теория (1 час):

- Знакомство с понятием «случайное число»

Практика (3 часа):

- Создание программы с использованием выбора случайного числа из указанного диапазона

Тема 4.3. Кот математик

Теория (2 часа):

- Знакомство с понятиями «типы данных», «переменная», «константа». Блок Данные, кнопка Создать переменную
- Операторы, их применимость к различным типам данных. Изменение переменных.
- Процедуры с параметрами.

Практика (4 часа):

- Упражнения по созданию переменных, определение их имён и типов
- Создание программы «Кот умеет складывать числа»: ввод данных с клавиатуры
- Создание программы «Кот умеет умножать»
- Создание программы «Кот умеет считать деньги»

#### **Раздел 5. Команды управления в среде программирования Scratch**

Тема 5.1. Ветвление. Реализация алгоритма ветвления в среде программирования Scratch

Теория (1 час):

- Знакомство с понятием «ветвление», полная развилка, структура обхода
- Блок Движение: применение команды Если на краю, оттолкнуться
- Знакомство с числом Гугол
- Изучение параметра команды Идти ... шагов
- Блок Стилль вращения влево-вправо
- Блок Внешность

Практика (1 час):

- Усложнение скрипта «Бегущий котёнок»
- Импорт в программу речи, изменение текста

Тема 5.2. Цикл. Реализация циклического алгоритма в среде программирования Scratch

Теория (1 час):

- Знакомство с понятием «цикл», виды циклических алгоритмов, особенности и отличия.
- Желтые блоки - команды группы «Управление». Паузы. Простые циклы. Цикл «Повторить».
- Запись цикла со счетчиком в виде блок – схемы.
- Понятия: «Заголовок цикла», «Тело цикла».
- Циклы и эффект цвета.
- Циклы и эффект призрака.
- Вращение.
- Бесконечный цикл.
- Автоматическая печать.

Практика (1 час):

- Учим спрайты совершать повторяющиеся действия. Создаём «Вечный двигатель».
- Усовершенствуем ранее созданные программы в Scratch. Создаём новые: «Гонки», «Полет самолета», «Разноцветный экран».
- Решение задачи на поворот персонажа на  $15^0$
- Решение задачи на поворот персонажа на  $180^0$  с использованием структуры Повторить (расчет количества повторений, вывод формулы)
- Изменение параметров команды для уменьшения/увеличения скорости вращения, понятие «максимальной скорости поворота»
- Анализ программы и получение вывода: есть ли смысл скрипта с 1 повторением.
- Повышенный уровень сложности: написать программу движения котёнка от края монитора до края с остановкой у краёв по 1 секунде (Блок Всегда)
  - Движение нескольких персонажей. Игра «Догонялки» со звуками

Тема 5.3. Реализация вложенных циклов в среде программирования Scratch

Теория (1 час):

- Знакомство с понятием «вложенный цикл», внешний и внутренний циклы, особенности реализации
- Вложенные циклы. Работа с алгоритмами.

Практика (1 час):



- Задания: «Вращающиеся квадраты», «Танцующий динозавр», «Снежинка», «Цветок», «Разные орнаменты», «Создаем рисунки из многоугольников».

## **Раздел 6. Создание мультфильмов в среде программирования Scratch**

### Тема 6.1. Этапы работы над проектом

#### Теория (1 час):

- Понятие «фон», фоны проектов из библиотеки
- Понятие «сценарий»: расположение персонажей, направление движения
- Выбор костюмов персонажам: вкладка Костюмы
- Графический редактор: прорисовка костюмов

#### Практика (1 час):

- Создание скриптов каждому персонажу
- Упорядочивание скриптов

### Тема 6.2. Мультфильм «Глубокое синее море» по технологической карте

#### Практика (2 часа):

- Разработка сценария: определение местоположения персонажей, прорисовка направлений движения
- Определение фона мультфильма
- Прорисовка костюмов персонажам (не менее 2-х каждому)
- Написание и отладка скриптов каждого персонажа с использованием текста, звуков

### Тема 6.3. Мультфильм «Кот и летучая мышь» (частичное использование технологической карты)

#### Практика (2 часа):

- Разработка сценария: определение местоположения персонажей, прорисовка направлений движения
- Определение фона мультфильма
- Прорисовка костюмов персонажам (не менее 2-х каждому)
- Написание и отладка скриптов каждого персонажа с использованием текста, звуков

### Тема 6.4. Мультфильм «Лабиринт»

#### Практика (2 часа):

- Разработка сценария: определение местоположения персонажей, прорисовка направлений движения: 1 Гига– движется по лабиринту, 2 Нано– объект для поиска
- Определение фона мультфильма
- Прорисовка костюмов персонажам (у каждого по 4 костюма: 3 весёлых и 1 грустный)
- Рисование лабиринта инструментом Линия
- Написание и отладка скриптов каждого персонажа с использованием текста, звуков

### Тема 6.5. Мультфильм «Гига и приведение»

#### Практика (2 часа):

- Разработка сценария: определение местоположения персонажей, прорисовка направлений движения, копирование персонажей

- Определение фона мультфильма
- Прорисовка костюмов персонажам (Гига – испуганный костюм, обычный костюм)
- Рисование лабиринта инструментом Линия
- Написание и отладка скриптов каждого персонажа с использованием текста, звуков

### **Раздел 8. Мой первый проект в среде программирования Scratch**

Теория (1 час):

- Определение продукта проекта – мультфильм
- Написание сценария, определение действующих персонажей, проработка костюмов

Практика (5 часов):

- Создание/добавление персонажей
- Прорисовка костюмов
- Написание скриптов
- Создание продукта проекта.
- Обсуждение, отладка
- Защита продукта

## **Учебный план второй год обучения**

№ п/п	Название раздела, темы	Количество часов			Формы аттестации (контроля)
		Всего	Теория	Практика	
<b>Раздел 1. Введение</b>					
1.1	Первичный инструктаж. Знакомство с курсом обучения.	1	1	-	Текущий контроль
1.2	Введение в среду программирования Pascal ABC.	2	1	1	Текущий контроль
1.3	Входное тестирование	1	-	1	Текущий контроль
<b>ИТОГО</b>		<b>4</b>	<b>2</b>	<b>2</b>	
<b>Раздел 2. Алгоритмизация. Основы программирования Pascal ABC.</b>					
2.1	Алгоритмы. Виды и способы описания	4	1	3	Текущий контроль
2.2	Этапы работы над программой. Среда Pascal ABC	4	1	3	Текущий контроль
2.3	Средства языка. Структура программы.	4	1	3	Текущий контроль
2.4	Типы данных, процедуры ввода- вывода. Линейные алгоритмы.	6	2	4	Текущий контроль
2.5	Стандартные функции	6	2	4	Текущий контроль

2.6	Логический тип данных	4	1	3	Текущий контроль
2.7	Символьный тип данных	4	1	3	Текущий контроль
2.8	Контрольная работа № 1 по теме: «Линейные алгоритмы и их программная реализация»	2	1	1	Контрольная работа
2.9	Разветвляющиеся алгоритмы и их программная реализация средствами языка Pascal ABC	6	1	5	Текущий контроль
2.10	Операторы перехода и выбора	4	1	3	Текущий контроль
2.11	Контрольная работа № 2 по теме: «Разветвляющиеся алгоритмы и их программная реализация»	2	1	1	Контрольная работа
2.12	Циклические алгоритмы. Операторы циклов.	20	4	16	Текущий контроль
2.13	Вложенные циклы	14	4	10	Текущий контроль
2.14	Контрольная работа №3 «Циклические алгоритмы и их программная реализация»	2	1	1	Контрольная работа
<b>ИТОГО</b>		<b>82</b>	<b>22</b>	<b>60</b>	
<b>Раздел 3. Алгоритм для конкретного исполнителя с фиксированным набором команд</b>					
3.1	Исполнитель. Система команд исполнителя.	2	1	1	Текущий контроль
3.2	Среда КУМИР для программирования исполнителя	4	1	3	Текущий контроль
3.3	Исполнитель Чертёжник	6	2	4	Текущий контроль
3.4	Исполнитель Черепашка	6	2	4	Текущий контроль
3.5	Исполнитель Робот	6	2	4	Текущий контроль
<b>ИТОГО</b>		<b>24</b>	<b>8</b>	<b>16</b>	
<b>Раздел 4. Программирование в среде Pascal ABC.</b>					
4.1	Массивы, способы описания. Одномерные массивы	8	2	6	Текущий контроль
4.2	Двумерные массивы	8	2	6	Текущий контроль
4.3	Метод «пузырька», как один из алгоритмов упорядочения массива	8	2	6	Текущий контроль
<b>ИТОГО</b>		<b>24</b>	<b>6</b>	<b>18</b>	
<b>Раздел 5. Решение олимпиадных задач</b>					
5.1	Решение задач повышенного	10	2	8	наблюдени

уровня сложности					е
<b>ИТОГО</b>	<b>10</b>	<b>2</b>	<b>8</b>		
<b>ВСЕГО ЗА ГОД ОБУЧЕНИЯ</b>	144	40	104		

## Содержание программы второй год обучения.

### Раздел 1. Введение

Тема 1.1. Первичный инструктаж. Знакомство с курсом обучения.

Теория (1 час):

- Режим труда и отдыха при работе на ПК, санитарные нормы времени.
- Защита от вредного воздействия ПК на состояние психики человека и его физическое состояние, меры профилактики.
- Рациональная организация труда и отдыха.
- Действие тока на организм; факторы, влияющие на исход поражения; мероприятия по защите от поражения электрическим током.
- Понятие о пожаре, горении; причины пожаров; средства пожаротушения
- Знакомство с курсом обучения.

Тема 1.2. Введение в среду программирования Pascal ABC

Теория (1 час):

- Место языка программирования Pascal среди языков программирования высокого уровня

Практика (1 час):

- Установка среды программирования Pascal ABC

Тема 1.3. Входное тестирование.

Практика (1 час)

### Раздел 2. Алгоритмизация. Основы программирования Pascal ABC

Тема 1.1. Алгоритмы. Виды и способы описания.

Теория (1 час):

- Понятие «Алгоритм», свойства алгоритма, назначение и область применения.
- Виды алгоритмов, способы их описания.
- Блок-схема алгоритма, основные геометрические фигуры, их параметры. Требования ГОСТа к построению алгоритмов.
- Словесное описание алгоритмов

Практика (3 часа):

- Словесное описание простейших алгоритмов

Тема 2.2. Этапы работы над программой. Среда Pascal ABC.

Теория (1 час):

- Системы программирования. Качество программ.
- Роль языка Паскаль среди языков программирования.
- Система программирования Паскаль: история развития, работа со средой.

Практика (3 часа):

Этапы работы над программой.

- Знакомство со средой программирования Pascal ABC

Тема 2.3. Средства языка. Структура программы.

Теория (1 час):

- Структура языка и программы на данном языке программирования (структура языка, алфавит языка, классификация данных, классификация операторов)
- Структура программы: разделы, служебные слова. Идентификаторы, правила их записи.

Практика (3 часа):

- Работа со структурой программы: игра пазлы

Тема 2.4. Типы данных, процедуры ввода-вывода. Линейные алгоритмы.

Теория (2 часа):

- Классификация типов данных.
- Процедуры ввода-вывода, их форматы.
- Структура алгоритма. Правила составления.

Практика (4 часа):

- Составление линейных алгоритмов и программ для их реализации.
- Решение задач с линейной алгоритмической структурой.

Тема 2.5 Стандартные функции.

Теория (2 часа):

- Стандартные функции языка Паскаль, их назначение, форматы.
- Тип аргумента, тип результата.

Практика (4 часа):

- Решение задач линейной структуры с использованием различных стандартных функций.

Тема 2.6 Логический тип данных.

Теория (1 час):

- Переменные и константы логического типа, служебные слова, функции для работы с ними
- Тип аргумента, тип результата.

Практика (3 часа):

- Решение задач линейной структуры с использованием логического типа данных и стандартных функций этого типа.

Тема 2.7. Символьный тип данных.

Теория (1 час):

- Переменные и константы символьного типа, служебные слова, функции для работы с ними.
- Тип аргумента, тип результата.

Практика (3 часа):

- Решение задач линейной структуры с использованием символьного типа данных и стандартных функций этого типа.

Тема 2.8. Контрольная работа № 1 по теме: «Линейные алгоритмы и их программная реализация»

Теория (1 час)

Практика (1 час)

## Тема 2.9. Разветвляющиеся алгоритмы и их программная реализация средствами языка Pascal ABC

### Теория (1 час):

- Назначение, структура, виды разветвляющихся алгоритмов, блок-схема; словесное описание, область их применения.
- Алгоритмы ветвления и их программная реализация.
- Оператор If <условие> THEN <действие> ELSE<действие> (полная развилка);
- Оператор If <условие> THEN <действие> (структура обхода);

### Практика (5 часов):

- Составление алгоритмов с полной структурой ветвления
- Составление алгоритмов со структурой обхода
- Решение задач, содержащих различные структуры ветвления

## Тема 2.10. Операторы перехода и выбора.

### Теория (1 час):

- Знакомство с понятием «переход» в программе, «прерывание».
- Оператор выбора CASE ...OF и его практическое применение. Программная реализация
- Оператор безусловного перехода GOTO. Метки. Программная реализация

### Практика (3 часа):

- Решение задач, содержащих структуру ветвления с использованием условного и безусловного перехода к частям программы.
- Решение задач, содержащих структуру ветвления с использованием оператора выбора. Оценка эффективности составления программы при большом количестве условий.

## Тема 2.11. Контрольная работа № 2 по теме: «Разветвляющиеся алгоритмы и их программная реализация»

### Теория (1 час)

- Назначение, структура, виды разветвляющихся алгоритмов

### Практика (1 час)

- Программная реализация разветвляющихся алгоритмов

## Тема 2.12. Циклические алгоритмы. Операторы циклов.

### Теория (4 часа):

- Виды циклических структур, их программная реализация.
- Цикл с параметром: особенность работы цикла, блок схема, словесное описание. Программная реализация данной циклической структуры средствами языка Pascal ABC
- Цикл с предусловием: особенность работы цикла, блок схема, словесное описание. Программная реализация данной циклической структуры средствами языка Pascal ABC
- Цикл с постусловием: особенность работы цикла, блок схема, словесное описание. Программная реализация данной циклической структуры средствами языка Pascal ABC

### Практика (16 часов):

- Решение задач, содержащих циклические структуры (цикл с параметром, цикл с предусловием)

- Повышенный уровень сложности – цикл с постусловием

Тема 2.13. Вложенные циклы.

Теория (4 часа):

- Назначение, виды структур вложения.
- Знакомство с понятиями «внешний», «внутренний» цикл. Особенности их реализации

Практика (10 часов):

- Решение задач с использованием вложенных циклических структур.

- Данный раздел содержит задачи повышенного уровня сложности, например, числа палиндромы.

Тема 2.14. Контрольная работа №3 «Циклические алгоритмы и их программная реализация»

Теория (1 час)

- Назначение, структура, виды циклических алгоритмов

Практика (1 час)

- Программная реализация циклических алгоритмов

### **Раздел 3. Алгоритм для конкретного исполнителя с фиксированным набором команд**

Тема 3.1. Исполнитель. Система команд исполнителя.

Теория (1 час):

- Знакомство с понятиями «исполнитель», «система команд исполнителя», виды исполнителя, особенности команд, которые понимает конкретный исполнитель.

Практика (1 час):

- Словесное описание алгоритмов для конкретных исполнителей.

Тема 3.2. Среда КУМИР для программирования исполнителя

Теория (1 час):

- Знакомство с интерфейсом среды КУМИР, правила создания новых полей
- Особенности реализации линейных алгоритмов в среде КУМИР
- Особенности реализации разветвляющихся алгоритмов в среде КУМИР
- Особенности реализации циклических алгоритмов в среде КУМИР

Практика (3 часа):

- Создание новых полей для различных исполнителей
- Словесное описание линейных алгоритмов действий
- Словесное описание разветвляющихся алгоритмов действий
- Словесное описание циклических алгоритмов действий

Тема 3.3. Исполнитель Чертёжник

Теория (2 часа):

- Знакомство с исполнителем Чертёжник, система команд исполнителя Чертёжник

Практика (4 часа):

- Решение задач с помощью данного исполнителя в среде КУМИР

### Тема 3.4. Исполнитель Черепашка

Теория (2 часа):

- Знакомство с исполнителем Черепашка, система команд исполнителя Черепашка

Практика (4 часа):

- Решение задач с помощью данного исполнителя в среде КУМИР

### Тема 3.5. Исполнитель Робот

Теория (2 часа):

- Знакомство с исполнителем Робот, система команд исполнителя Робот

Практика (4 часа):

- Решение задач с помощью данного исполнителя в среде КУМИР

## **Раздел 4. Программирование в среде Pascal ABC**

### Тема 4.1. Массивы, способы описания. Одномерные массивы

Теория (2 часа):

- Знакомство с понятием «массив», как примером упорядочения данных.
- Способы описания массива в среде программирования Pascal ABC
- Понятие «одномерный (линейный) массив», описание одномерного массива

Практика (6 часов):

- Отображение одномерного массива при описании блок-схемой
- Решение задач на обработку элементов одномерного массива

### Тема 4.2. Двумерные массивы

Теория (2 часа):

- Понятие «двумерный (табличный) массив», описание двумерного массива

Практика (6 часов):

- Отображение двумерного массива при описании блок-схемой (вложение циклов)
- Решение задач на обработку элементов двумерного массива

### Тема 4.3. Метод «пузырька», как один из алгоритмов упорядочения массива

Теория (2 часа):

- Знакомство с понятием «сортировка», как примером упорядочения данных по заданному условию.
- Методы сортировки: метод «пузырька» (особенность данного метода, простота и эффективность)

Практика (6 часов):

- Решение задач на упорядочение элементов одномерного массива по заданным условиям
- Задачи повышенного уровня сложности – упорядочение элементов двумерного массива
- Другие методы сортировки

## **Раздел 5. Решение олимпиадных задач**

### Тема 5.1. Решение задач повышенного уровня сложности

Теория (2 часа)



Практика (8 часов)

**Комплекс организационно-педагогических условий  
Календарный учебный график (Приложение 1).**

**Материально-техническое обеспечение педагогического процесса**

Для реализации дополнительной общеобразовательной программы «Школа программирования» имеются в наличии:

- помещение для занятий с достаточным освещением (не менее 300-500лк);
- столы, стулья;
- интерактивная доска;
- экран;
- мультимедийный проектор;
- маркерная доска;
- принтер;
- доступ в сеть Интернет;
- персональные компьютеры/ноутбуки.

Рекомендуемое оборудование на учебную группу из 14 обучающихся

<b>Основное оборудование и материалы</b>	<b>Кол-во</b>	<b>Ед. изм.</b>
Ноутбук с предустановленным ПО (ОС Windows, среды программирования Кумир, Pascal ABC, Scratch, пакетом прикладных программ MS Office), компьютерная мышь	4	шт.
Персональный компьютер с предустановленным ПО (ОС Windows, среды программирования Кумир, Pascal ABC, Scratch, пакетом прикладных программ MS Office), компьютерная мышь	12	шт.

**Методическое обеспечение программы (формы и методы организации учебной деятельности):**

Для реализации программы используются следующие

– формы организации учебной деятельности - фронтальная, индивидуальная, групповая

– формы организации учебного процесса - теоретическое занятие, практикум, работа со специальной литературой (инструкционными картами), мини-конференция, фестивали, творческие выставки, соревнования.

– методы организации учебной деятельности:

1. методы формирования сознания и личностных смыслов:

– словесные (объяснение, лекция, беседа, диалог, учебная дискуссия, диспут).

– работа с информацией: с учебной книгой, с дополнительной научной и популярной литературой, Интернет;

2. методы организации познавательной деятельности и опыта общественного поведения:

– методы организации учебной работы: инструктаж, иллюстрация, демонстрация, наблюдение, упражнение, приучение, создание ситуации, самостоятельная работа (индивидуальная, групповая, в парах), взаимообучение;

– методы познавательной деятельности: репродуктивные (действия по образцу, по алгоритму), проблемно-поисковые (анализ проблемной ситуации, выдвижение гипотез, догадка, мозговая атака),

– проблемно-исследовательские (моделирование, теоретический анализ);

– методы, отражающие логический путь познания: эмпирические (опора на субъектный опыт), теоретические (опора на теоретические закономерности); анализ, синтез, сравнение, аналогия, обобщение, индуктивные, дедуктивные;

– методы, отражающие степень субъектности ученика: активные, интерактивные, пассивные;

– методы управления учебно-познавательной деятельностью: указание, предъявление требований, направляющие вопросы, алгоритмические предписания, индивидуальная поддержка, самоуправление.

3. методы стимулирования и мотивации учебно- познавательной деятельности и поведения:

– методы эмоционального воздействия: создание ситуаций эмоционально-нравственного, эстетического переживания, занимательности, новизны, парадоксальности, ситуации успеха, увлеченности поиском неизвестного, положительные подкрепления, поощрения;

– стимулирование личностной значимости учения: убеждение, опора на жизненный опыт, имитационное моделирование жизненных и профессиональных ситуаций, познавательные игры.

4. методы контроля эффективности образовательного процесса:

– наблюдение;

– опросы: устный, письменный; индивидуальный, групповой, фронтальный, уплотненный, компьютерный;

– педагогическая диагностика: проверочные и контрольные работы, тестирование; самопроверка, взаимопроверка, проверка педагогом, независимыми экспертами, компьютерная;

– методы оценивания: критериальный.

Педагогические технологии, которые применяются при работе с учащимися

Название	Цель
Технология личностно-ориентированного обучения	Развитие индивидуальных технических способностей на пути профессионального самоопределения учащихся.
Технология развивающего обучения	Развитие личности и ее способностей через вовлечение в различные виды деятельности.
Технология проблемного обучения	Развитие познавательной активности, самостоятельности учащихся.
Технология дифференцированного обучения	Создание оптимальных условий для выявления задатков, развития интересов и способностей, используя методы индивидуального обучения.
Технологии здоровье сберегающие	Создание оптимальных условий для сохранения здоровья учащихся.
Игровые технологии	Активизация умственной деятельности обучающихся, развитие внимания и

	интереса к изучению языков программирования, минимизация пассивности и социализация обучающихся в коллективе.
--	---

## **Диагностика результативности образовательного процесса – Приложение 2 к программе.**

Основные методы контроля: наблюдение, опросы, педагогическая диагностика.

Система мониторинга разработана по видам контроля

Предварительный – имеет диагностические задачи и осуществляется в начале учебного года.

Цель предварительного контроля – зафиксировать начальный уровень подготовки учащихся, имеющиеся знания, умения и навыки, связанные с предстоящей деятельностью:

- умение пользоваться ПК,
- умение составлять алгоритм (например, словесный) для описания события/действия/явления
- умение содержать в порядке рабочее место,
- умение доводить работу до конца.

Текущий – предполагает систематическую проверку и оценку знаний, умений и навыков по конкретным темам в течение учебного года

Промежуточный – осуществляется в середине учебного года с целью оценки теоретических знаний, а также практических умений и навыков по итогам полугодия:

- умение следовать правилам поведения, соблюдать технику безопасности,
- знание основных особенностей алгоритмов и их реализация,
- умение программировать алгоритмы, используя различные среды программирования,
- умение содержать в порядке рабочее место,
- умение доводить работу до конца.

Итоговый – проводится в конце каждого года обучения и предполагает оценку теоретических знаний, практических умений и навыков.

Результаты заносятся в диагностическую карту.

**Контрольно–измерительные материалы (Приложение 3).**

### **Список литературы для педагога**

1. Абрамов С. А., Зима Е. В. Начала программирования на языке Паскаль. - М.: Наука. Гл. ред. физ.-мат. лит., 1987. - 112 с.
2. Введение в программирование: Учеб. пособие для уч. сред. и ст. шк. возраста / Авт.-сост. В. А. Гольденберг. - Минск: ООО «Харвест», 1997.- 528 с. (Библиотека школьника).
3. Великович Л., Цветкова М. Программирование для начинающих. – М.: Бином, 2007
4. Григорьев С. А. Программирование на языке Паскаль для математиков: Учебное пособие / Изд. 2-е, испр. и доп.; Калинингр. ун-т.— Калининград, 2000. ISBN 5-88874-189-2.
5. Голиков Д. В. Scratch для юных программистов. — СПб.: БХВ-Петербург, 2017. - 192 с.: ил.
6. Окулов С. М. Основы программирования. - М.: Лаборатория Базовых Знаний, 2002.- 424 с.
7. Патаракин Е.П. Учимся готовить в среде Скретч. Версия 2.0
8. Полат Е.С. Новые педагогические и информационные технологии в системе образования. – М. Академия. – 2006.
9. Рындак, В.Г. «Проектная деятельность школьника в среде программирования Scratch»: учебно-методическое пособие / В.Г. Рындак, В. О. Дженжер, Л. В. Денисова. - Оренбург: Оренб. гос. ин-т. менеджмента, 2009 -116 с.
10. Румянцев Д. Г., Монастырский Л. Ф. Путь программиста: Опыт создания личности программиста.- М.: ИНФРА-М, 2000.— 864 с., ил. ISBN 5-16-000507-2.
11. Шпынева С. М. Методическое пособие Технологии Scratch. – Тамбов, 2014. – 29с: ил.
12. Щетинская А. И. Совершенствование процесса дополнительного образования в современных условиях: научно-методическое пособие/ А.И. Щетинская. – Оренбург: ОЦДЮТ, 1997 – 100 с.
- 13.

### **Список литературы для обучающихся и родителей**

1. Патаракин, Е.Д. «Учимся готовить в среде Скретч»: Учеб. Метод.пособие/ Е.Д. Патаракин. - М: 2007.-189 с.
2. Голиков, Д.В. Книга юных программистов на Scratch: учебное пособие/ Д.В. Голиков, А.Д. Голиков. – М.: ВHV, 2017 – 192 с.

### **Сетевые ресурсы:**

1. <http://scratch.mit.edu> – сетевое сообщество для обмена проектами, сайт, где можно скачать свободно распространяемую программу Scratch
2. Олейникова, О. Н. Социальное партнерство в профессиональном образовании [Электронный ресурс] / О. Н. Олейникова. - Режим доступа: <http://www.cvets.ru/SPart/SocPartnership.pdf>
3. <http://scratch.mit.edu> - официальный сайт проекта Scratch

## Приложение 1

### Календарный учебный график Дополнительная общеобразовательная общеразвивающая программа технической направленности «Школа программирования»

Год обучения - первый  
Количество часов -72.

№ п/п	Месяц	Число	Время	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля/ аттестации
1.	сентябрь			Комбинированное занятие	2	Первичный инструктаж. Знакомство с курсом обучения. История языков программирования. Введение в среду программирования Scratch		Наблюдение
2.	сентябрь			Комбинированное занятие	2	Введение в среду программирования Scratch. Входное тестирование		Наблюдение
3.	сентябрь			Комбинированное занятие	2	Интерфейс среды программирования Scratch. Линейный алгоритм		Наблюдение
4.	сентябрь			Комбинированное занятие	2	Спрайты. Библиотеки костюмов и сцен		Наблюдение
5.	сентябрь			Практикум.	2	Графический редактор Scratch		Наблюдение, тест
6.	октябрь			Практикум	2	Добавление нового спрайта		Опрос/тест
7.	октябрь			Практикум	2	Учимся управлять спрайтами		Опрос/тест
8.	октябрь			Комбинированное занятие.	2	Создание спецэффектов - команды «Внешность»		Опрос/тест
9.	октябрь			Практикум	2	Знакомство с пером		Опрос/тест
10.	октябрь			Практикум.	2	Работа со звуком		Опрос/тест
11.	ноябрь			Контрольная работа	2	Контрольная работа №1 «Знакомство со средой Scratch»		Контрольная работа
12.	ноябрь			Комбинированное занятие.	2	Цветовой эффект		Опрос/тест
13.	ноябрь			Комбинированное занятие.	2	Эффект рыбьего глаза		Опрос/тест
14.	ноябрь			Комбинированное занятие.	2	Эффект завихрения		Опрос/тест
15.	ноябрь			Комбинированное занятие.	2	Эффект укрупнения пикселей		Опрос/тест
16.	ноябрь			Комбинированное занятие.	2	Эффекты мозаики и яркости.		Опрос/тест
17.	ноябрь			Комбинированное	2	Эффект призрака		Опрос/тест

				занятие				
18.	ноябрь			Контрольная работа	2	Практическая контрольная работа №2 «Эффекты среды программирования Scratch»		Контрольная работа
19.	декабрь			Комбинированное занятие	2	Знакомство с координатами x и y		Опрос/тест
20.	декабрь			Практикум.	2	Случайные числа		Опрос/тест
21.	декабрь			Комбинированное занятие	2	Кот математик		Опрос/тест
22.	декабрь			Практикум.	2	Кот математик		Опрос/тест
23.	декабрь			Комбинированное занятие	2	Кот математик		Промежуточный контроль
24.	январь			Комбинированное занятие	2	Ветвление. Реализация алгоритма ветвления в среде программирования Scratch		Опрос/тест
25.	январь			Практикум.	2	Реализация алгоритма ветвления в среде программирования Scratch		Опрос/тест
26.	январь			Комбинированное занятие	2	Цикл. Реализация циклического алгоритма в среде программирования Scratch		Опрос/тест
27.	февраль			Практикум.	2	Цикл. Реализация циклического алгоритма в среде программирования Scratch		Опрос/тест
28.	февраль			Практикум.	2	Цикл. Реализация циклического алгоритма в среде программирования Scratch		Опрос/тест
29.	февраль			Комбинированное занятие	2	Реализация вложенных циклов в среде программирования Scratch		Опрос/тест
30.	февраль			Проектная деятельность	2	Этапы работы над проектом		Наблюдение
31.	март			Проектная деятельность	2	Мультфильм «Глубокое синее море» по технологической карте		Защита проекта
32.	март			Проектная деятельность	2	Мультфильм «Кот и летучая мышь» (частичное использование технологической карты)		Защита проекта
33.	март			Проектная деятельность	2	Мультфильм «Лабиринт»		Защита проекта
34.	март			Комбинированное занятие	2	Мультфильм «Гига и приведение»		Защита проекта
35.	май			Проектная деятельность	2	Мой первый проект в среде программирования Scratch. Создание мультфильма/компьютерной игры		Наблюдение
36.	май			Проектная деятельность	2	Мой первый проект в среде программирования Scratch. Создание мультфильма/компьютерной игры		Итоговый контроль (защита проекта)

**Календарный учебный график**  
**Дополнительная общеобразовательная общеразвивающая программа технической направленности**  
**«Школа программирования»**

Год обучения - второй

Количество часов -144.

№ п/п	Месяц	Число	Время	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля/ аттестации
1.	сентябрь			Комбинированное занятие	2	Первичный инструктаж. Знакомство с курсом обучения. Введение в среду программирования Pascal ABC.		Наблюдение
2.	сентябрь			Комбинированное занятие	2	Введение в среду программирования Pascal ABC. Входное тестирование		Наблюдение
3.	сентябрь			Комбинированное занятие	2	Алгоритмы. Виды и способы описания		Наблюдение
4.	сентябрь			Практикум.	2	Алгоритмы. Виды и способы описания		Наблюдение, тест
5.	сентябрь			Комбинированное занятие	2	Этапы работы над программой. Среда Pascal ABC		Наблюдение
6.	сентябрь			Практикум.	2	Этапы работы над программой. Среда Pascal ABC		Наблюдение, тест
7.	сентябрь			Комбинированное занятие	2	Средства языка. Структура программы.		Наблюдение
8.	сентябрь			Практикум.	2	Средства языка. Структура программы.		Наблюдение, тест
9.	сентябрь			Комбинированное занятие.	2	Типы данных, процедуры ввода-вывода. Линейные алгоритмы.		Наблюдение
10.	октябрь			Практикум	2	Типы данных, процедуры ввода-вывода. Линейные алгоритмы.		Опрос/тест
11.	октябрь			Комбинированное занятие.	2	Линейные алгоритмы.		Опрос/тест
12.	октябрь			Комбинированное занятие.	2	Стандартные функции		Опрос/тест
13.	октябрь			Практикум.	2	Стандартные функции		Опрос/тест
14.	октябрь			Практикум.	2	Стандартные функции		Опрос/тест
15.	октябрь			Комбинированное занятие.	2	Логический тип данных		Опрос/тест
16.	октябрь			Практикум	2	Логический тип данных		Опрос/тест
17.	октябрь			Комбинированное занятие.	2	Символьный тип данных		Опрос/тест
18.	октябрь			Практикум.	2	Символьный тип данных		Опрос/тест

19.	ноябрь			Контрольная работа	2	Контрольная работа № 1 по теме: «Линейные алгоритмы и их программная реализация»		Контрольная работа
20.	ноябрь			Комбинированное занятие.	2	Разветвляющиеся алгоритмы и их программная реализация средствами языка Pascal ABC		Опрос/тест
21.	ноябрь			Практикум.	2	Разветвляющиеся алгоритмы и их программная реализация средствами языка Pascal ABC		Опрос/тест
22.	ноябрь			Практикум.	2	Разветвляющиеся алгоритмы и их программная реализация средствами языка Pascal ABC		Опрос/тест
23.	ноябрь			Комбинированное занятие.	2	Операторы перехода и выбора		Опрос/тест
24.	ноябрь			Практикум.	2	Операторы перехода и выбора		Опрос/тест
25.	ноябрь			Контрольная работа	2	Контрольная работа № 2 по теме: «Разветвляющиеся алгоритмы и их программная реализация»		Контрольная работа
26.	ноябрь			Комбинированное занятие	2	Циклические алгоритмы. Операторы циклов.		Опрос/тест
27.	декабрь			Комбинированное занятие	2	Циклические алгоритмы. Операторы циклов.		Опрос/тест
28.	декабрь			Практикум.	2	Циклические алгоритмы. Операторы циклов.		Опрос/тест
29.	декабрь			Комбинированное занятие	2	Циклические алгоритмы. Операторы циклов.		Опрос/тест
30.	декабрь			Практикум.	2	Циклические алгоритмы. Операторы циклов.		Опрос/тест
31.	декабрь			Комбинированное занятие	2	Циклические алгоритмы. Операторы циклов.		Опрос/тест
32.	декабрь			Практикум.	2	Циклические алгоритмы. Операторы циклов.		Опрос/тест
33.	декабрь			Практикум.	2	Циклические алгоритмы. Операторы циклов.		Промежуточный контроль (тест)
34.	декабрь			Комбинированное занятие	2	Циклические алгоритмы. Операторы циклов.		Промежуточный контроль (тест)
35.	январь			Практикум.	2	Циклические алгоритмы. Операторы циклов.		Опрос/тест
36.	январь			Комбинированное занятие	2	Вложенные циклы		Опрос/тест
37.	январь			Практикум.	2	Вложенные циклы		Опрос/тест
38.	январь			Практикум.	2	Вложенные циклы		Опрос/тест
39.	январь			Практикум.	2	Вложенные циклы		Опрос/тест
40.	январь			Практикум.	2	Вложенные циклы		Опрос/тест
41.	февраль			Практикум.	2	Вложенные циклы		Опрос/тест
42.	февраль			Практикум.	2	Вложенные циклы		Опрос/тест
43.	февраль			Контрольная работа	2	Контрольная работа №3 «Циклические алгоритмы и их программная реализация»		Контрольная работа
44.	февраль			Комбинированное	2	Исполнитель. Система команд исполнителя.		Опрос/тест



				занятие				
45.	февраль			Комбинированное занятие	2	Среда КУМИР для программирования исполнителя		Опрос/тест
46.	февраль			Практикум.	2	Среда КУМИР для программирования исполнителя		Опрос/тест
47.	февраль			Комбинированное занятие	2	Исполнитель Чертёжник		Опрос/тест
48.	февраль			Проектная деятельность	2	Исполнитель Чертёжник		Опрос/тест
49.	март			Проектная деятельность	2	Исполнитель Чертёжник		Соревнование
50.	март			Комбинированное занятие	2	Исполнитель Черепашка		Опрос/тест
51.	март			Проектная деятельность	2	Исполнитель Черепашка		Опрос/тест
52.	март			Проектная деятельность	2	Исполнитель Черепашка		Соревнование
53.	март			Комбинированное занятие	2	Исполнитель Робот		Опрос/тест
54.	март			Проектная деятельность	2	Исполнитель Робот		Опрос/тест
55.	март			Проектная деятельность	2	Исполнитель Робот		Соревнование
56.	март			Комбинированное занятие	2	Массивы, способы описания. Одномерные массивы		Опрос/тест
57.	март			Комбинированное занятие	2	Массивы, способы описания. Одномерные массивы		Опрос/тест
58.	апрель			Комбинированное занятие	2	Массивы, способы описания. Одномерные массивы		Опрос/тест
59.	апрель			Комбинированное занятие	2	Массивы, способы описания. Одномерные массивы		Опрос/тест
60.	апрель			Комбинированное занятие	2	Двумерные массивы		Опрос/тест
61.	апрель			Проектная деятельность	2	Двумерные массивы		Опрос/тест
62.	апрель			Комбинированное занятие	2	Двумерные массивы		Опрос/тест
63.	апрель			Комбинированное занятие	2	Двумерные массивы		Опрос/тест
64.	апрель			Комбинированное занятие.	2	Метод «пузырька», как один из алгоритмов упорядочения массива		Опрос/тест
65.	апрель			Комбинированное	2	Метод «пузырька», как один из алгоритмов		Опрос/тест

				занятие.		упорядочения массива	
66.	апрель			Комбинированное занятие.	2	Метод «пузырька», как один из алгоритмов упорядочения массива	Опрос/тест
67.	май			Комбинированное занятие.	2	Метод «пузырька», как один из алгоритмов упорядочения массива	Опрос/тест
68.	май			Комбинированное занятие	2	Решение задач повышенного уровня сложности	Наблюдение
69.	май			Практикум	2	Решение задач повышенного уровня сложности	Наблюдение
70.	май			Практикум	2	Решение задач повышенного уровня сложности	Наблюдение
71.	май			Практикум	2	Решение задач повышенного уровня сложности	Наблюдение
72.	май			Практикум	2	Решение задач повышенного уровня сложности	Итоговый контроль (олимпиада)

## Диагностика результативности образовательного процесса

Виды контроля

<b>Виды контроля</b>	<b>Содержание</b>	<b>Формы контроля</b>	<b>Сроки контроля</b>
Предварительный	Начальный уровень подготовки учащихся, имеющиеся знания, умения и навыки, связанные с предстоящей деятельностью	Наблюдение, тест	Сентябрь
Текущий	Освоение учебного материала по темам.	Опрос, тест, проектная деятельность	Октябрь-апрель
Промежуточный	Освоение учебного материала за полугодие	Контрольная работа	Декабрь-январь
Итоговый	Освоение учебного материала за год	Итоговая работа, защита проекта	Май



## Диагностическая форма учёта результатов итоговой аттестации

### ДИАГНОСТИЧЕСКАЯ КАРТА

Детское объединение:

Бюджет/платный сертификат/внебюджет (подчеркнуть)

Дата проведения:

Форма проведения:

Аттестация: промежуточная/итоговая

Срок реализации программы: \_\_ года

Год обучения: \_\_ Группа \_\_

№	Ф.И.О. ребенка	Теоретич. задания		Практическая подготовка			Уровень развития и воспитанности	Уровень знаний (Высокий, Средний, Низкий)
		Знание интерфейса среды программирования Pascal ABC, КУМИР	Знание принципов работы среды программирования Pascal ABC, КУМИР: создание программы/среды исполнителя, компиляция, трассировочные таблицы для проверки результатов работы	Умение использовать программные средства среды программирования Pascal ABC для реализации различных видов алгоритмов (линейных, разветвляющихся,	Умение использовать среду программирования Pascal ABC, КУМИР для решения задач различного уровня сложности	Умение оценивать эффективность использования алгоритма для решения задач		
							Культура организации самостоятельной деятельности, аккуратность и ответственность при работе, взаимодействие в коллективе	

			программы	циклических, вложенных циклических структур) базового и повышенного уровня сложности				
1								
2								
3								

### Оценка уровней освоения программы

Уровни / количество во %	Параметры	Общие критерии оценки результативности обучения	Показатели
Высокий уровень/ 80-100%	Теоретические знания.	Оценка уровня теоретических знаний по программным требованиям: широта кругозора, свобода восприятия теоретической информации, развитость практических навыков работы со специальной литературой, осмысленность и свобода использования специальной терминологии	Учащийся освоил материал в полном объеме. Знает и понимает значение терминов, самостоятельно ориентируется в содержании материала по темам. Учащийся заинтересован, проявляет устойчивое внимание к выполнению заданий.
	Практические умения и	Оценка уровня практической подготовки учащихся: соответствие развития уровня	Способен применять практические умения и навыки во время выполнения самостоятельных

	навыки.	практических умений и навыков программным требованиям, свобода владения специальным оборудованием и оснащением, качество выполнения практического задания, технологичность практической деятельности	заданий. Правильно и по назначению применяет инструменты. Работу аккуратно доводит до конца. Может оценить результаты выполнения своего задания и дать оценку работы своего товарища.
Средний уровень / 50%-79%	Теоретические знания.	Оценка уровня теоретических знаний по программным требованиям: широта кругозора, свобода восприятия теоретической информации, развитость практических навыков работы со специальной литературой, осмысленность и свобода использования специальной терминологии	Учащийся освоил базовые знания, ориентируется в содержании материала по темам, иногда обращается за помощью к педагогу. Учащийся заинтересован, но не всегда проявляет устойчивое внимание к выполнению задания.
	Практические умения и навыки.	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	Владеет базовыми навыками и умениями, но не всегда может выполнить самостоятельное задание, затрудняется и просит помощи педагога. В работе допускает небрежность, делает ошибки, но может устранить их после наводящих вопросов или самостоятельно. Оценить результаты своей деятельности может с подсказкой педагога.
Низкий уровень / Ниже 50%	Теоретические знания.	Оценка уровня теоретических знаний по программным требованиям: широта кругозора, свобода восприятия теоретической информации, развитость практических навыков работы со специальной литературой, осмысленность и свобода использования специальной	Владеет минимальными знаниями, ориентируется в содержании материала по темам только с помощью педагога.

		терминологии	
	Практические умения и навыки.	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	Владеет минимальными начальными навыками и умениями. Учащийся способен выполнять каждую операцию только с подсказкой педагога или товарищей. Не всегда правильно применяет необходимый инструмент или не использует вовсе. В работе допускает грубые ошибки, не может их найти их даже после указания. Не способен самостоятельно оценить результаты своей работы.



**Контрольно-измерительные материалы ДООП «Школа  
программирования»**

Входная диагностика  
Тестирование

Ф.И. \_\_\_\_\_

1. Что такое Scratch?
  - a. Сообщество, в котором собираются люди и обсуждают свои проблемы.
  - b. Язык программирования
  - c. Программа по созданию приложений на телефон.
  - d. Затрудняюсь ответить.
2. Что такое спрайт?
  - a. Фирма, логотип, работодатель.
  - b. Напиток.
  - c. Действующее лицо в программе Scratch.
  - d. Затрудняюсь ответить.
3. Для чего нужны языки программирования?
  - a. Для того, чтобы компьютер понимал и выполнял команды человека.
  - b. Для того, чтобы компьютер работал сам как ему нравится без участия человека.
  - c. Для того, чтобы у человека была работа.
  - d. Затрудняюсь ответить.
4. Кто такой программист?
  - a. Человек, который придумывает и собирает робота.
  - b. Человек, который следит за роботом.
  - c. Человек, который пишет алгоритм (шаги) для деятельности робота.
  - d. Затрудняюсь ответить.
5. Кто такой инженер?
  - a. Человек, который придумывает и собирает робота.
  - b. Человек, который следит за роботом.
  - c. Человек, который пишет алгоритм (шаги) для деятельности робота.
  - d. Затрудняюсь ответить.

Промежуточная диагностика  
Тестирование

Ф.И. \_\_\_\_\_

Вопрос 1. Как переводится с английского название программы Скретч?

1. Царапка
2. Котёнок
3. Лисёнок

Вопрос 2. Для чего предназначена программа Скретч?

1. Для программирования в режиме конструктора
2. Для рисования мультиков
3. Для написания сайтов

Вопрос 3. Каких блоков нет в программе (несколько вариантов ответа)?

1. Движение
2. Внешность
3. Фигуры
4. Контроль
5. Сенсоры
6. Картинки

Вопрос 4. Что такое спрайт?

1. Объект программы
2. Напиток
3. Загадочное существо

Вопрос 5. Что такое скрипт?

1. Звуки в программе
2. Программа, по которой действует герой
3. Отдельные действия спрайта

Вопрос 6. Можно ли вставить песню, скачанную через Интернет, в качестве звука в программу?

1. Нет
2. Да
3. Да, предварительно записав её через микрофон

Вопрос 7. Можно ли рисовать спрайт самим?

1. Да
2. Нет

Вопрос 8. Можно ли с помощью данной программы создавать игры?

1. Да
2. Нет

Вопрос 9. Есть ли в Скретч графический редактор?

1. Нет
2. Да

Вопрос 10. Зачем спрайту нужны костюмы?

1. Для красоты
2. Чтоб не замёрзнуть
3. Для создания анимации

Вопрос 11. Минимальная смысловая единица (команда) языка Скретч, которая служит для создания скриптов (сценариев) - это

1. скрипт
2. блок
3. алгоритм
4. действие
5. стек

Вопрос 12. Алгоритм (или сценарий), составленный из блоков языка Скретч для какого-либо объекта - это

1. скрипт
2. блок
3. алгоритм
4. действие
5. стек

Вопрос 13. Исполнитель алгоритмов, которому доступны ВСЕ команды языка Скретч - это

1. скрипт
2. спрайт
3. сцена
4. интерфейс
5. стек

Вопрос 14. Это неподвижный графический объект, который изображает место действия проекта. Является исполнителем алгоритмов, которому доступны почти все команды Скретч, кроме команд движения и рисования.

1. скрипт
2. спрайт
3. сцена
4. интерфейс
5. стек

Вопрос 15. Понятная компьютеру система символов для точной записи алгоритмов и их выполнения компьютером.

1. интерфейс
2. язык программирования
3. скрипт
4. блок-схемы
5. сценарий

Вопрос 16. Алгоритм, выраженный на языке программирования.

1. интерфейс
2. язык программирования
3. программа
4. блок-схемы
5. сценарий

Итоговая диагностика  
Тестирование.

Ф.И. \_\_\_\_\_

Вопрос 1

Какой скрипт проведёт спрайт по квадратной траектории?



Варианты ответов

- Первый
- Второй
- Третий

Вопрос 2

Какой скрипт проведёт спрайт по квадратной траектории?

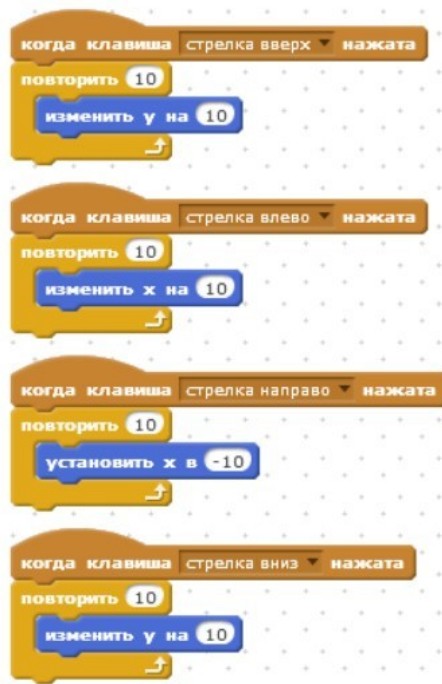


Варианты ответов

- Первый
- Второй
- Третий
- Четвертый

Вопрос 3

**Куда сместится спрайт при нажатии стрелки влево?**

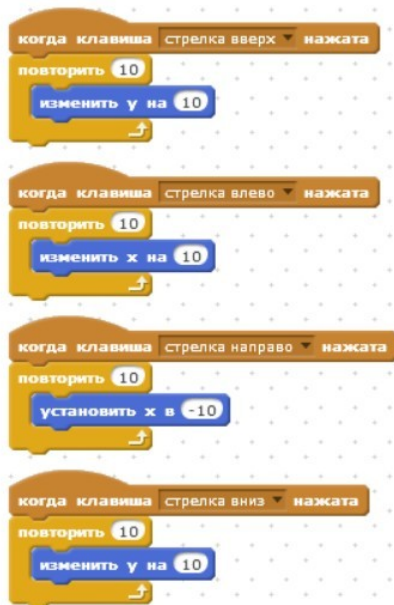


Варианты ответов

- вправо
- влево
- вверх
- вниз

Вопрос 4

**Куда сместится спрайт при нажатии стрелки вверх?**



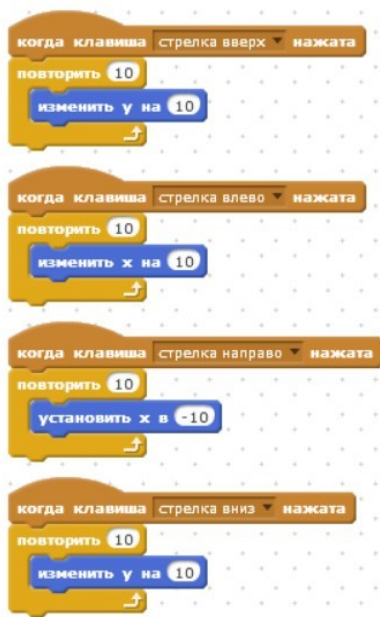
Варианты ответов

- вправо
- влево
- вверх

- вниз

Вопрос 5

Куда сместится спрайт при нажатии стрелки вниз.



Варианты ответов

- вправо
- влево
- вверх
- вниз

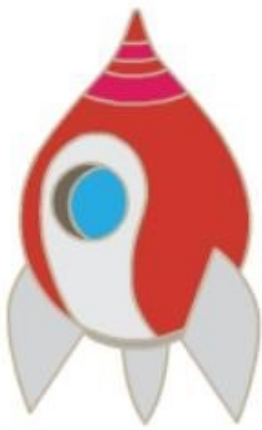
Вопрос 6. Я старался сделать так, чтобы персонаж не исчезал за краем сцены, а появлялся с другой стороны и продолжал двигаться, как ни в чём не бывало. В каких блоках ошибки?



Варианты ответов

- 1, 2, 3 и 4
- 1 и 2
- 2 и 4

Вопрос 7. Ракета расположена в точке  $X = 170$ ,  $Y = -100$ . В какой точке окажется ракета, после нажатия на пробел?



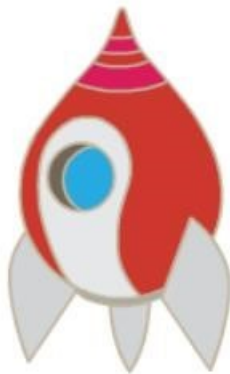
```
когда щелкнут по [флаг]
перейти в x: 170 y: -100
показать

когда клавиша пробел нажата
повторить 20
  изменить y на 10
скрыть
```

Варианты ответов

- X = 170, Y = -100
- X = -170, Y = 100 • X = 170, Y = 100

Вопрос 8 . Как ускорить полёт ракеты по сцене, чтобы она быстрее добралась до точки X = 170, Y = 100?



```
когда щелкнут по [флаг]
перейти в x: 170 y: -100
показать

когда клавиша пробел нажата
повторить 20
  изменить y на 10
скрыть
```

- В блоке "изменить Y на 10" заменить 10 на 20
- Вместо блока "повторить 20" использовать блок "всегда"
- Удалить блок "скрыть"

Вопрос 9. Что такое спрайт?

- Напиток
- Программа на Scratch
- Загадочное существо
- Объект программы

Вопрос 10. Что такое скрипт?

- Звуки в программе
- Звук двери
- Отдельные действия спрайта
- Программа, по которой действует герой

Вопрос 11. Какой блок позволяет изменить направление движения у края экрана?

1. "Если на краю, оттолкнуться";
2. "Если на краю, повернуться";
3. "Если на краю, отразиться".

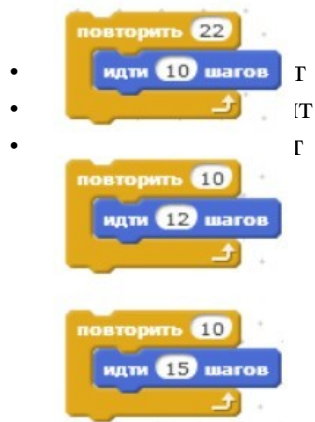
Вопрос 12. При выполнении какого скрипта Котёнок совершит один оборот вокруг своей оси?



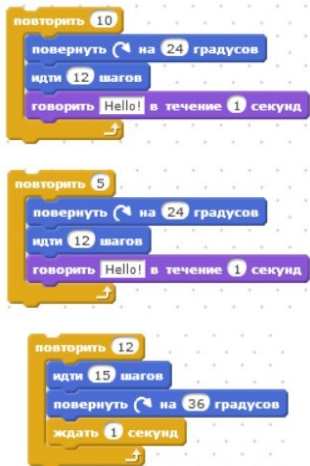
- Второй скрипт
- Первый скрипт
- Третий скрипт

Вопрос 13. Какой из этих блоков переместит Котёнка на 120 шагов?

Варианты ответов



Вопрос 14. Какой из этих скриптов будет выполняться дольше других?





- Первый скрипт
- Третий скрипт
- Второй скрипт

Вопрос 15. Какой из этих скриптов выполнится быстрее всех?

```

повторить 12
  идти 15 шагов
  повернуть на 36 градусов
  сказать Hello!
  играть звук meow
  
```

```

повторить 10
  повернуть на 24 градусов
  идти 12 шагов
  говорить Hello! в течение 1 секунд
  если на краю, оттолкнуться
  
```

```

повторить 5
  повернуть на 24 градусов
  идти 12 шагов
  говорить Hello! в течение 1 секунд
  
```

- Третий скрипт
- Первый скрипт
- Второй скрипт

Вопрос 16. Какой из скриптов Котёнка переместит его при нажатии на клавишу "1"?

```

когда клавиша пробел нажата
  идти 10 шагов
  
```

```

когда клавиша 1 нажата
  идти 10 шагов
  
```

```

когда клавиша 1 нажата
  повернуться к Sprite2
  
```

Варианты ответов

- Третий
- Второй
- Первый