

Управление образования и молодежной политики администрации
Павловского муниципального округа Нижегородской области

Муниципальное автономное общеобразовательное учреждение
средняя школа №9 с углубленным изучением отдельных предметов
г.Павлово

Рассмотрена и принята
на педагогическом совете
Протокол №1 от 30.08.2024 г.

УТВЕРЖДАЮ
Директор школы
_____ Д.С. Соколов
« 30 » августа 2024 г.

Дополнительная общеобразовательная
общеразвивающая программа
технической направленности
«Начала программирования»

Возраст обучающихся: 11-12 лет
Срок реализации: 1 год

Автор-составитель: Кучинова Любовь
Юрьевна, педагог дополнительного
образования

г. Павлово, 2024 г.

Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Начала программирования» разработана в соответствии с Федеральным Законом от 29.12.2012 г. №273 – ФЗ «Об образовании в Российской Федерации» и Приказом Министерства просвещения Российской Федерации от 09.11.2018 г. № 196. «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

Современное общество требует от специалистов разных сфер высокого уровня знаний и умений в области компьютерных и информационных технологий, эффективных способов коммуникации и обработки информации. Это означает, что большой необходимостью является формирование у школьников определенной информационно-компьютерной культуры и соответствующего мировоззрения. Кроме этого важно, чтобы человек не только умел работать за компьютером, но и понимал, как устроены программы, с помощью которых он работает на нём.

Для изучения основных принципов программирования была выбрана визуальная событийно-ориентированная среда программирования Scratch. Scratch - это новая среда программирования, которая позволяет детям создавать собственные анимированные и интерактивные истории, презентации, модели, игры и другие произведения.

Scratch базируется на традициях языка Лого и Лего-Лого. В среде Scratch используется метафора кирпичиков Лего, из которых даже самые маленькие дети могут собрать простейшие конструкции. Но, начав с малого, можно дальше развивать и расширять свое умение строить и программировать. Скретч создавался специально для того, чтобы подростки 8 – 16 лет использовали его самостоятельно. Эта новая технологическая среда позволяет им выразить себя в компьютерном творчестве. Кроме этого, дети получают возможность строить алгоритмы и программировать действия различных героев из мультфильмов и сказок в программной среде Scratch. Начав с малого, они смогут и дальше расширять и развивать свое умение

строить и программировать.

Актуальность программы состоит в том, что изучение основ программирования связано с развитием целого ряда таких умений и навыков, которые носят общеинтеллектуальный характер и формирование которых – одна из приоритетных задач современной школы. Изучение программирования развивает мышление школьников, способствует формированию у них многих приемов умственной деятельности. Здесь роль информатики сродни роли математики в школьном образовании. Поэтому не использовать действительно большие возможности программирования для развития мышления школьников, формирования многих общеучебных, общеинтеллектуальных умений и навыков было бы, наверное, неправильно.

Педагогическая значимость данной программы заключается также в том, что в ней большое внимание уделяется исследовательской деятельности, так как многие практические задачи, встающие перед учащимися, требуют грамотного и эффективного умения распределять свою работу по времени, искать свои ошибки и анализировать работу в целом. К тому же, занятия посещают и одаренные дети и дети из семей социального риска, что положительно скажется на их воспитании, развитии, саморазвитии и самореализации.

Программа направлена на реализацию технического направления.

Основные принципы реализации модульность и практическая направленность, что обеспечит вариативность обучения.

Содержание учебных модулей направлено:

- на детальное изучение алгоритмизации; –
- на реализацию межпредметных связей;
- на организацию проектной и исследовательской деятельности обучающихся.

Адресат программы. Программа предназначена для работы с детьми среднего школьного возраста (11 -12 лет). Набор учащихся в объединение свободный, независимо от национальной и половой принадлежности, социального статуса родителей (или законных представителей).

Рекомендуемое количество детей в группе – 15 человек.

Цель программы: Интеллектуальное и творческое развитие учащихся на базе повышенного познавательного интереса к информатике, основанного на поиске новых познавательных ориентиров с последующей профессиональной деятельностью в различных предметных областях.

Задачи программы:

Предметные:

- изучение методов программирования на языке программирования Scratch;
- приобретение навыков разработки эффективных алгоритмов и программ на основе изучения языка программирования Scratch;
- формирование навыков использования языка методов программирования в учебной и последующей профессиональной деятельности в различных предметных областях;
- научить учащихся ориентироваться и продуктивно действовать в информационном Интернет-пространстве, используя для достижения своих целей создаваемые веб-ресурсы;
- формирование знаний о способах научно-технического мышления и деятельности, направленных на самостоятельное творческое познание и исследование информационной части сетевого пространства;
- формирование и развитие навыков алгоритмического и логического мышления, грамотной разработки программ;
- знакомство с принципами и методами функционального программирования;
- формирование знаний о принципах и методах объектно-ориентированного программирования;

Метапредметные:

- формирование у обучаемых творческого мышления, способности к самостоятельному и инициативному решению проблем;
- реализовать коммуникативные, технические и эвристические

способности учащихся в ходе проектирования и конструирования сайтов.

- приобретение навыков поиска информации в сети Интернет, анализ выбранной информации на соответствие запросу, использование информации при решении задач;
- систематизации знаний по математическим основам информатики. – развитие у обучающихся интереса к программированию.

Личностные:

- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;
- формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе, приобретённой благодаря иллюстрированной среде программирования мотивации к обучению и познанию;
- формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;
- формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;
- формирование ценности здорового и безопасного образа жизни; усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.
- способны осуществлять рефлексивную деятельность, оценивать свои результаты, корректировать дальнейшую деятельность.

Объем и срок реализации программы, режим занятий. Программа рассчитана на 1 год обучения учащихся в возрасте 11-12 лет (5-6 классы), но в случае необходимости может быть реализована в течение более длительного срока. Время, отведенное на обучение, составляет 34 часа в год, причем практические занятия составляют большую часть программы.

Занятия проходят 1 раз в неделю, продолжительность занятия 40 минут

Учебный год начинается 2 сентября и заканчивается 27 мая.

70% содержания планирования направлено на практическую деятельность.

Формы организации образовательного процесса

Основной тип занятий - комбинированный, сочетающий в себе элементы теории и практики. Большинство заданий курса выполняется самостоятельно с помощью персонального компьютера и необходимых программных средств.

Единицей учебного процесса является блок уроков (модуль). Каждый такой блок охватывает отдельную информационную технологию или её часть. Внутри блоков разбивка по времени изучения производится учителем самостоятельно, но с учётом рекомендованного календарно-тематического плана. С учётом регулярного повторения ранее изученных тем темп изучения отдельных разделов блока определяется субъективными и объективными факторами.

Каждая тема курса начинается с постановки задачи — характеристики предметной области или конкретной программы на языке Scratch, которую предстоит изучить. С этой целью учитель проводит демонстрацию презентации или показывает саму программу, а также готовые работы, выполненные в ней. Закрепление знаний проводится с помощью практики отработки умений самостоятельно решать поставленные задачи, соответствующих минимальному уровню планируемых результатов обучения.

Основные задания являются обязательными для выполнения всеми обучающимися в группе. Задания выполняются на компьютере с использованием установленной среды Scratch на компьютере или онлайн интегрированной среды на сайте <https://scratch.mit.edu/>. При этом ученики не только формируют новые теоретические и практические знания, но и приобретают новые технологические навыки.

В каждую тему курса интегрированы вопросы, связанные с теоретическими основами изучаемого раздела. Это позволяет учащимся

получить более глубокое представление о принципах работы с информацией и способах автоматизации этой деятельности.

Методика обучения ориентирована на индивидуальный подход. Для того чтобы каждый ученик получил наилучший результат обучения, программой предусмотрены дополнительные задания для самостоятельной работы на домашнем компьютере. Такая форма организации обучения стимулирует интерес ученика к предмету, активность и самостоятельность учащихся, способствует объективному контролю глубины и широты знаний, повышению качества усвоения материала обучающимися, позволяет педагогу получить объективную оценку выбранной им тактики и стратегии работы, методики индивидуального обучения и обучения в группе, выбора предметного содержания.

Для самостоятельной работы используются разные по уровню сложности задания, которые носят репродуктивный и творческий характер. Количество таких заданий в работе может варьироваться.

Выполнение контрольных заданий способствует активизации учебно-познавательной деятельности и ведёт к закреплению знаний, а также служит индикатором успешности образовательного процесса.

Методы организации учебного процесса

При организации занятий по программе «Начала программирования» используются формы проведения занятий с активными методами обучения:

- занятие в форме проблемно-поисковой деятельности;
- занятие с использованием межпредметных связей;
- занятие в форме мозгового штурма;
- занятие в форме частично-поисковой деятельности.

Планируемые результаты освоения программы курса «Начала программирования»

В результате реализации программы у обучающихся будут сформированы личностные, регулятивные, познавательные и коммуникативные универсальные учебные действия.

Предметные результаты:

Учащиеся должны знать:

- правила выполнения алгоритма при заданных исходных данных, узнавать изученные алгоритмы
- знать о математических объектах информатики и об их свойствах, представлять информацию в структурированном виде;
- знать основы предметных понятий «информация», «алгоритм», «исполнитель», «модель» и их свойства;
- правила разработки программ в выбранной среде программирования Scratch, включая тестирование и отладку программ;
- основы несложных программ анализа данных, читать и понимать несложные программы, написанные на алгоритмическом языке;

Учащиеся должны уметь:

- составлять и записывать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами — линейной, условной и циклической;
- выполнять пошагово (с использованием компьютера или вручную) несложные алгоритмы;
- использовать основные управляющие конструкции и библиотеки прикладных программ, выполнять созданные программы;
- разрабатывать и использовать компьютерные модели, анализировать готовые модели на предмет соответствия реальному объекту или процессу.

Метапредметные результаты:

Учащиеся должны уметь:

- самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- планировать пути решения поставленной проблемы для получения эффективного результата; понимать, что в программировании длинная программа не всегда лучшая;
- критически оценивать правильность решения учебно-

- исследовательской задачи;
- корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;
 - владеть основами самоконтроля, способность к принятию решений;
 - создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;
 - формирование и развитие компетентности в области использования информационно-коммуникационных технологий (ИКТ-компетенция);
 - организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.

Личностные результаты:

- критическое отношение к информации и избирательность её восприятия;
- уважение к информации о частной жизни и информационным результатам других людей;
- осмысление мотивов своих действий при выполнении заданий с жизненными ситуациями;
- начало профессионального самоопределения, ознакомление с миром профессий, связанных с информационными и коммуникационными технологиями;
- сформированы ценности здорового и безопасного образа жизни; усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Формы контроля и аттестации

Для полноценной реализации данной программы используются текущий и промежуточный виды контроля.

Формы текущего контроля теоретической подготовки: устный опрос, практические работы, участие в проектной деятельности.

Формы текущего контроля практической подготовки: участие в научно-практических конференциях, олимпиадах, конкурсах.

Требования к результатам выполнения исследовательского проекта:

- умение планировать и осуществлять проектную и исследовательскую деятельность;
- способность презентовать достигнутые результаты, включая умение определять приоритеты целей с учетом ценностей и жизненных планов;
- самостоятельно реализовывать, контролировать и осуществлять коррекцию своей деятельности на основе предварительного планирования;
- способность использовать доступные ресурсы для достижения целей;
- осуществлять выбор конструктивных стратегий в трудных ситуациях;
- способность создавать продукты своей деятельности, востребованные обществом, обладающие выраженными потребительскими свойствами;
- сформированность умений использовать все необходимое многообразие информации и полученных в результате обучения знаний, умений и компетенций для целеполагания, планирования и выполнения индивидуального проекта.

Формы и разделы текущего контроля по курсу «Начала программирования»

№	Разделы	Форма контроля
1.	Рассмотрение циклов. Создание простых мультфильмов.	<i>Практические работы</i>
2.	Работа с костюмами. Первая игра на Scratch.	
3.	Взаимодействие героев. Игра с двумя и более уровнями.	
4.	Смена локаций. Добавление новых и удаление старых персонажей.	
5.	Использование математики. Подсчет очков. Использование условий.	
6.	Изучение операций дублирования и клонирования. Добавление эффектов.	

7.	Рисование. Создание собственных предметов и персонажей.	
8.	Создание игры «викторина». Создание игры от первого лица.	
9.	Промежуточная аттестация.	<i>Тестовая работа</i>

Результаты текущего контроля анализируются педагогом дополнительного образования по следующим уровням:

- высокий уровень;
- средний уровень; –
- базовый уровень.

Формы и порядок проведения промежуточной аттестации.

Для определения уровня обученности учащихся по дополнительной общеразвивающей программе используется система оценивания теоретической знаний и практической подготовки учащихся.

Промежуточная аттестация проводится самостоятельно педагогом дополнительного образования. Во время проведения промежуточной аттестации может присутствовать администрация школы. Промежуточная аттестация учащихся проводится по итогам учебного года (май).

Оценка, оформление и анализ результатов промежуточной аттестации.

Для определения уровня обученности учащихся по дополнительной общеразвивающей программе используется система оценивания подготовки учащихся. Теоретическая подготовка проверяется через выполнение **практических работ** (приложение), практическая подготовка – **защита проекта.**

По итогам года выполняется творческий проект, в рамках промежуточного контроля с 15 мая, защита которого может осуществляется по нескольким уровням.

- муниципальный – высокий уровень,
- школьный уровень - средний уровень,
- внутригрупповой уровень - базовый уровень,

Результаты участия учащихся в мероприятиях областного и выше уровней в течение учебного года оценивается педагогом как предварительная сдача промежуточной аттестации.

Оценка, оформление и анализ результатов промежуточной аттестации.

Основным контрольно-измерительным материалом является итоговый протокол, в котором фиксируется в суммарное значение теоретической и практической части прохождения промежуточной аттестации учащихся.

**Протокол
результатов промежуточной аттестации, обучающихся по
дополнительным общеобразовательным общеразвивающим
программам МБОУ СШ №9 с углубленным изучением отдельных
предметов г.Павлово**

20___/20___учебный год

Объединение: _____

Год обучения _____

ФИО педагога _____

Дата проведения аттестации _____

Форма проведения _____

Форма оценки результатов аттестации: уровень (высокий, средний, базовый)

№ п/п	Ф.И.О. учащегося	Теоретическая	Практическая подготовка	Результат
		подготовка		Уровень аттестации
1.	Иванов И.	Высокий	Высокий	высокий

Учебный план программы

Всего часов	В том числе теория	В том числе практика	Форма текущего контроля
34	9	25	<p><u>Теоретическая часть:</u> устный опрос, практические работы, участие в проектной деятельности.</p> <p><u>Практическая часть:</u> участие в научно-практических конференциях, олимпиадах, конкурсах, защита проекта.</p>

**Календарный учебный график
дополнительной общеобразовательной общеразвивающей программы «Начала программирования»
на 2024-2025 учебный год**

Комплектование групп проводится с 2 по 15 сентября 2024 года.

Продолжительность учебного года составляет 34 учебных недель. Учебные занятия в МАОУ СШ №9 с углубленным изучением отдельных предметов г. Павлово начинаются с 2 сентября 2024 г. и заканчиваются 27 мая 2025 г.


Учебные занятия проводятся во 2-ю смену (в соответствии с расписанием). Продолжительность занятий составляет 40 мин.

Каникулы: зимние каникулы с 30.12.2024 г. по 12.01.2025 г.; летние каникулы с 01.06.2025 г. по 31.08.2025 г.

В каникулярное время занятия в объединениях не проводятся. Во время каникул учащиеся могут принимать участие в мероприятиях в соответствии с планами воспитательной работы школы и классного руководителя.

Год обучения	Сентябрь					Октябрь				Ноябрь				Декабрь				Январь				Февраль			Март			Апрель				Май			Июнь	Июль	Август	Всего учебных недель/ часов		
	01.09	08.09	15.09	22.09	29.09	06.10	13.10	20.10	27.10	09.11	16.11	23.11	30.11	07.12	14.12	21.12	28.12	05.01	12.01	19.01	26.01	02.02	09.02	16.02	02.03	09.03	16.03	23.03	06.04	13.04	20.04	27.04	04.05	11.05	18.05	25.05	01.06 – 30.06		01.07 – 31.07	01.08 – 31.08
1 год	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	К	К	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	К	К	К	34/34

Условные обозначения:

Промежуточная аттестация – 

Ведение занятий по расписанию – 

Каникулярный период – 

Рабочая программа курса

№ п/п	Название раздела, темы	Количество учебных часов			Форма контроля
		<i>Всего</i>	<i>Теория</i>	<i>Практика</i>	
1	Знакомство со средой. Написание первой программы.	1	1		опрос
2	Рассмотрение циклов. Создание простых мультфильмов.	2	1	1	<i>Практическая работа</i>
3	Работа с костюмами. Первая игра на Scratch.	2	1	1	<i>Практическая работа</i>
4	Взаимодействие героев. Игра с двумя и более уровнями.	4	1	4	<i>Практическая работа</i>
5	Смена локаций. Добавление новых и удаление старых персонажей.	6	1	5	<i>Практическая работа</i>
6	Использование математики. Подсчет очков. Использование условий.	6	1	5	<i>Практическая работа</i>
7	Изучение операций дублирования и клонирования. Добавление эффектов.	4	1	3	<i>Практическая работа</i>
8	Рисование. Создание собственных предметов и персонажей.	4		4	<i>Практическая работа</i>
9	Создание игры «викторина». Создание игры от первого лица.	3		3	<i>Практическая работа</i>
10	Промежуточная аттестация. Презентация итоговых проектов.	2	2		<i>Тестовая работа</i> <i>Защита проекта</i>
	итого	34	9	25	

Содержание программы курса

Тема 1 «Знакомство со средой. Написание первой программы» (2 часа) Знакомство со средой. Изучения понятий «программа», «проект», «подпрограмма». Изучение особенностей языка. Объекты, операторы, функции. Создание программы, сохранение.

Тема 2 «Рассмотрение циклов. Создание простых мультфильмов» (3 часа). Рассмотрение различных способов передвижения героя. Оси координат. Местоположение героя. Рассмотрение правила хода

мультфильма. Создание простейшего мультфильма.

Тема 3 «Работа с костюмами. Первая игра на Scratch» (2 часа)

Знакомство с понятием «костюмы». Рассмотрение различных эффектов. Выведение правил игры, составление сценария, создание игры.

Тема 4 «Взаимодействие героев. Игра с двумя и более уровнями» (7 часов). Изучение правил взаимодействия двух героев. Введение нового персонажа. Задание определенных действий. Построение взаимодействия между ними.

Тема 5 «Смена локаций. Добавление новых и удаление старых персонажей» (6 часов). Изучение правил взаимодействия двух героев. Введение нового персонажа. Задание определенных действий. Построение взаимодействия между ними.

Тема 6 «Использование математики. Подсчет очков. Использование условий» (6 часов). Рассмотрение роли математики в подсчете очков в играх. Введение в игру счета. Использование условий.

Тема 7 «Изучение операций дублирования и клонирования. Добавление эффектов» (4 часа). Изучение правил взаимодействия двух героев. Введение нового персонажа. Задание определенных действий. Построение взаимодействия между ними.

Тема 8 «Рисование. Создание собственных предметов и персонажей» (4 часа). Создание собственноручно нарисованных персонажей. Добавление их в программную среду и в приложение. Действия с ними.

Тема 9 «Создание игры «викторина». Создание игры от первого лица» (3 часа). Подробная запись правил игры. Составление списка вопросов, составление списка вариантов ответов. Тестирование игры.

Тема 10 «Презентация итоговых проектов. Проведение открытого занятия» (2 часа). Анализ пройденного материала и сделанных приложений.

Условия реализации программы курса

«Начала программирования»

В основу программы положен принцип воздействия, производимый педагогом на воспитанника (обучающегося), представляющий ему выработать потребность найти свое место в жизни, развить свои способности и таланты.

Важнейшим условием реализации программы является создание развивающей, здоровьесберегающей образовательной среды как комплекса комфортных, психолого-педагогических и социальных условий, необходимых для развития творческих интересов и способностей детей.

Программа курса разработана с учетом индивидуальных возрастных, психологических и физиологических особенностей обучающихся и определяет систему требований:

- **Целеполагание.** Перед обучающимися ставятся конкретные, достижимые, понятные, диагностируемые цели. Целеполагание осуществляется совместно с обучающимися исходя из сформулированной проблемы. Обучающиеся должны знать, какие конкретно знания, умения и способы деятельности они освоят в процессе деятельности.
- **Мотивация.** Формирование интереса как к процессу учебной деятельности, так и к достижению конечного результата. Эффективными мотивами являются решение актуальной проблемы, практическая направленность содержания.
- **Практическая значимость знаний и способов деятельности.** Педагог должен показать обучающимся возможности применения осваиваемых знаний и умений в их практической деятельности.
- **Отбор содержания.** Планируемые результаты, определенные программой, отрабатываются на практике.

Материально-технические условия реализации программы:

- компьютерный класс с 15 персональными компьютерами;
- ноутбук,
- локальная сеть,
- глобальная сеть,
- мультимедиа проектор,
- интерактивная доска.
- принтер,
- сканер,
- среда программирования Scratch.

Учебно-методическое и информационное обеспечение программы

Список литературы:

Литература для педагога:

- Ю. В. Пашковская «Творческие проекты в среде Scratch» для 5–6 классов
- http://lbz.ru/metodist/authors/informatika/3/files/5kl_Scratch_lessons_1-10_2509_2015_Sorokina.pdf

Литература для учащихся:

- http://dvboyarkin.ru/wp-content/uploads/2015/05/Scratch_2014_1.pdf
- http://губернеф.пф/wp-content/uploads/2012/02/Grebnev_scratch_2012.pdf

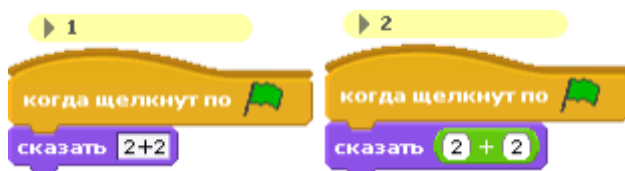
Список рекомендуемых Интернет-ресурсов:

- <https://brainbasket.org/ru/besplatnyie-uroki-scratch/>
- <https://stepik.org/lesson/92383/step/15>
- <http://scratch.aelit.net/category/scratch-games-lessons/>
- <https://sites.google.com/site/pishemkody/vvedenie>

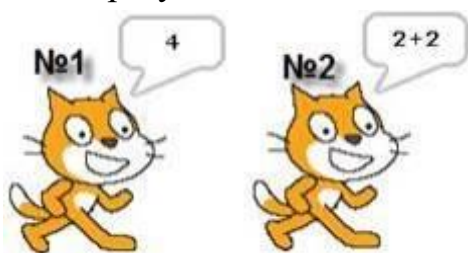
**Оценочные и методические
материалы для промежуточной аттестации
учащихся и текущего контроля курса
«Начала программирования»**

Пример итоговых вопросов по теме «Типы данных»

1. В каком примере скрипт работает с числовыми, а в каком со строковыми данными:



2. Где результат вывода на экран работы первого скрипта, а где второго?



3. Определите, чему равно значение оператора И: $6 > 2$ и $2 < 4$

А) истина; Б) ложь.

4. Определите, чему равно значение оператора И: $6 > 2$ и $2 < 1$

А) истина; Б) ложь.

5. Определите, чему равно значение оператора ИЛИ: $6 > 2$ или $2 < 4$

А) истина; Б) ложь.

6. Определите, чему равно значение оператора ИЛИ: $6 > 2$ или $2 < 1$

А) истина; Б) ложь.

7. Определите, чему равно значение оператора НЕ: не $6 > 2$ или $2 < 4$

А) истина; Б) ложь.

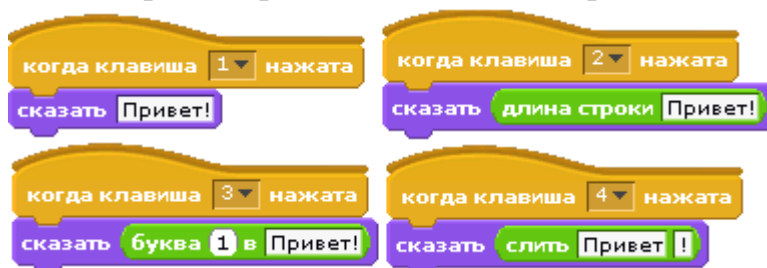
8. Определите, чему равно значение оператора НЕ: не $6 > 2$ и $2 < 4$

А) истина; Б) ложь.

9. Установите, сколько математических функций содержит репортер:



10. Соберите скрипты по данным образцам.



Пример вопросов для обсуждения перед началом занятий:

1. Сколько циклов в программе? Назовите их.
2. Какой цикл является вложенным, а какой внешним?
3. Какие команды содержит цикл повтори...?
4. Какие команды содержит цикл всегда?
5. Сколько всего шагов делает кот, прежде чем останавливается на 3 секунды?

Промежуточная аттестация проходит в форме теста, тесты разрабатываются по каждому разделу дисциплины.

1. Точный порядок действий объекта для достижения успеха.
 - A. алгоритм
 - B. программа
 - C. циклический алгоритм
 - D. ветвящийся алгоритм
 - E. линейный алгоритм
2. Все, что существует в природе: люди, животные, ветер, снег, дерево, солнце, буквы и т. д. В Скретче их называют спрайтами (англ. Sprite - вымышленный герой, эльф).
 - A. алгоритм
 - B. программа
 - C. циклический алгоритм
 - D. пространство
 - E. объекты
3. Повторение определенной последовательности действий заданное количество раз.
 - A. алгоритм
 - B. программа
 - C. циклический алгоритм
 - D. ветвящийся алгоритм
 - E. линейный алгоритм
4. Действия выполняются последовательно друг за другом, как бы по одной линии - от начала до конца.
 - A. алгоритм
 - B. программа
 - C. циклический алгоритм

- D. ветвящийся алгоритм
- E. линейный алгоритм

5. Включает две или более ветви решения задачи, выбор ветви зависит от логического условия в точке ветвления.

- A. алгоритм
- B. программа
- C. циклический алгоритм
- D. ветвящийся алгоритм
- E. линейный алгоритм

6. Алгоритм должен дробиться на отдельные минимальные шаги, операции. Каждый шаг алгоритма должен исполняться только после того, как закончилось исполнение

- A. дробность
- B. программа
- C. циклический алгоритм
- D. определенность
- E. конечность

7. Каждая команда должна точно определять действие объекта.

- A. дробность
- B. программа
- C. циклический алгоритм
- D. определенность
- E. конечность

8. Выполнение алгоритма должно заканчиваться успехом за конечное число шагов.

- A. дробность
- B. программа
- C. циклический алгоритм
- D. определенность
- E. конечность

9. Набор команд, которые может выполнять исполнитель алгоритма.

- A. алгоритм
- B. программа
- C. система команд исполнителя
- D. поведение
- E. скрипт

10. Это действия объекта в ответ на события и сообщения.

- A. алгоритм
- B. программа

- C. система команд исполнителя
- D. поведение
- E. скрипт

11. Это все окна и кнопки, которые вы видите на экране запущенной программы.

- A. интерфейс
- B. программа
- C. система команд исполнителя
- D. объекты
- E. скрипт

12. Верно ли, что в графическом редакторе Скретч нельзя сделать рисунок фона сцены?

- A. верно
- B. нет

13. Верно ли, что существует 7 способов запуска редактора изображений в Скретче?

- A. верно
- B. нет

14. Можно ли изменить имя фона?

- A. можно
- B. нельзя
- C. иногда можно

15. Можно ли изменить имя спрайта?

- A. можно
- B. нельзя
- C. иногда можно

16. Можно ли изменить имя костюма?

- A. можно
- B. нельзя
- C. иногда можно

17. Можно ли изменять порядок фонов в списке?

- A. можно
- B. нельзя
- C. иногда можно

18. Можно ли с помощью инструмента Ножницы удалять скрипты, костюмы и спрайты?

- A. можно

- В. нельзя
- С. иногда можно

19. Можно ли с помощью инструмента Штамп создавать копии скриптов, костюмов и спрайтов?

- А. можно
- В. нельзя
- С. иногда можно
- Д. только копии скриптов

20. Можно ли для сцены сделать несколько скриптов?

- А. можно
- В. нельзя
- С. иногда можно
- Д. можно не более 7

21. Сколько костюмов может иметь спрайт?

- А. 1
- В. любое количество
- С. 2
- Д. можно не более 7

22. Сколько осей имеется в прямоугольной системе координат?

- А. 2
- В. 4
- С. 1
- Д. 6

23. Сколько видов графических эффектов в Скретче?

- А. 2
- В. 4
- С. 1
- Д. 6
- Е. 7

24. Чему равна ширина сцены?

- А. 360 точек
- В. 480 точек
- С. 640 точек
- Д. 260 точек
- Е. может меняться

25. Чему равна координата X в крайней левой точки сцены?

- А. 480
- В. -180

- C. 180
- D. 240
- E. -240

26. Минимальная смысловая единица (команда) языка Скретч, которая служит для создания скриптов (сценариев)?

- A. скрипт
- B. блок
- C. алгоритм
- D. действие
- E. стек

27. Это алгоритм (или сценарий), составленный из блоков языка Скретч для какого-либо объекта.

- A. скрипт
- B. блок
- C. алгоритм
- D. действие
- E. стек

28. Это подвижный графический объект, который действует на сцене проекта, выполняя разнообразные алгоритмы (сценарии). Исполнитель алгоритмов, которому доступны ВСЕ команды языка Скретч.

- A. скрипт
- B. спрайт
- C. сцена
- D. интерфейс
- E. стек

29. Форма блоков, которые служат для остановки выполнения одного скрипта или всех скриптов проекта. В Скретче всего 2 блока этого вида.

- A. стек
- B. заголовок
- C. заглушка
- D. логический
- E. С-блок

30. Самая многочисленная форма блоков, из которых создаются стопки (предложения) - последовательности блоков в скриптах.

- A. стек
- B. заголовок
- C. заглушка
- D. логический
- E. С-блок

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

- А. скрипт
- В. спрайт
- С. сцена
- Д. интерфейс
- Е. стек

32. Форма блоков, которые содержат логическое значение (истина-ложь), вставляются в шестиугольные и прямоугольные окна других блоков, позволяют создавать логические условия для ветвлений и циклов.

- А. стек
- В. репортер
- С. заглушка
- Д. логический
- Е. С-блок

33. Форма блоков, которые содержат данные и вставляются в прямоугольные и овальные окна других блоков.

- А. стек
- В. репортер
- С. заглушка
- Д. логический
- Е. С-блок

34. Форма блоков, которые всегда стоят в начале скриптов. Указывают на способ, событие запуска скрипта. Блоки этого вида размещены в категории контроля.

- А. стек
- В. заголовок
- С. заглушка
- Д. логический
- Е. С-блок

35. Форма блоков, которые служат для создания циклов и ветвлений. Блоки этого вида размещены в категории контроля.

- А. стек
- В. заголовок
- С. заглушка
- Д. логический
- Е. С-блок

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

- А. интерфейс
- В. язык программирования
- С. скрипт
- Д. блок-схемы
- Е. сценарий

37. Алгоритм, выраженный на языке программирования.

- А. интерфейс
- В. язык программирования
- С. программа
- Д. блок-схемы
- Е. сценарий