

**Частное учреждение дополнительного образования детей и взрослых
«ТАЛАНТИУМ»**

<i>Рассмотрена на заседании педагогического совета №1, протокол № 1 от «10» апреля 2023г.</i>	<i>«Утверждаю»: Директор ЧУДО «Талантиум» Синицына Н.С. /_____/ «10» апреля 2023г.</i>
---	--

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«Программирование HeyGo»**

НАПРАВЛЕННОСТЬ: ТЕХНИЧЕСКАЯ

Уровень: ознакомительный
Возраст обучающихся: 7 – 12 лет
Язык преподавания -русский
Срок реализации: 3 года

**Автор-составитель:
Баранов Егор Павлович,
педагог дополнительного образования**

ОГЛАВЛЕНИЕ

<i>РАЗДЕЛЫ ПРОГРАММЫ</i>	<i>СТРАНИЦЫ</i>
1. Пояснительная записка	3
1.1. Актуальность и педагогическая целесообразность программы	3
1.2. Цель и задачи программы	4
1.3. Планируемые результаты	5
1.4. Формы организации образовательной деятельности	6
2. Содержание программы	6
<i>1 год обучения</i>	
2.1. Учебно-тематический план	6
2.2. Содержание учебно-тематического плана	8
<i>2 год обучения</i>	
2.3. Учебно-тематический план	14
2.4. Содержание учебно-тематического плана	15
<i>3 год обучения</i>	
2.3. Учебно-тематический план	23
2.4. Содержание учебно-тематического плана	24
3. Формы контроля и оценочные материалы	30
4. Организационно-педагогические условия реализации программы.	31

1. Пояснительная записка.

Дополнительная общеобразовательная общеразвивающая программа «**Программирование Neugo**» (далее – Программа) разработана на основании действующих федеральных, региональных нормативно-правовых документов, определяющих содержание дополнительного образования детей, а также с учётом социально-психологического статуса детей 7 - 12 лет.

Программа *технической* направленности, *ознакомительного* уровня.

В процессе реализации Программы происходит становление интереса детей к информационным технологиям, а также развитие технических и творческих способностей.

1.1. Актуальность программы.

Актуальность программы обусловлена возросшим интересом общества к мобильным телефонам, компьютерам и прочим цифровым устройствам. Количество пользователей ими растет с каждым днем, и эта тенденция просуществует не одно десятилетие.

Сегодня специалистами в области информационных технологий разрабатываются программы и приложения, которые позволяют решать огромное количество задач. Игры входят в число самых популярных приложений.

Современные школьники очень увлечены компьютерными играми, но они не задумываются над тем, где могут применить накопленный опыт. Реализация программы позволит обучающимся перейти из статуса игрока в статус разработчика и понять, что создание своей игры – увлекательный и познавательный процесс.

Обучаясь по данной программе, дети будут осваивать основы объектного программирования, а также мобильной разработки, создавая игры и иные приложения. Также обучающиеся закрепят некоторые темы школьного курса математики.

Новизна программы заключается в использовании современных средств разработки приложений: эмуляторов, облачных сред и мобильных телефонов на Android.

Отличительные особенности программы

. В процессе реализации данной программы, обучающиеся поэтапно осваивают разработку игр и других полезных программ при помощи современных средств с плавным приростом сложности, проходя 3 этапа:

- **Scratch** - среда визуального программирования с графическим интерфейсом, которая была создана медиалабораторией Массачусетского технологического института, чтобы сделать программирование простым, понятным и интересным именно для детей.

- **Mit App Inventor** - среда разработки поддерживающая концепцию no – code программирования и не требующая особых навыков от пользователя, кроме стандартных базовых навыков работы с компьютером: умение работать в графических и текстовых редакторах, умение работать в браузерах.

- **Python** – универсальный язык программирования с динамической типизацией, который позволяет разрабатывать программы в соответствии с разными задачами: процедурным программированием, объектно-ориентированным, параметрическим, функциональным. В данном курсе рассматриваются все основные возможности языка Python и их применение при разработке программ.

1.2. Цель и задачи программы

Цель : Создание благоприятных условий для развития научно-технического и творческого потенциала личности учащегося, а также первое знакомство с основными алгоритмическими конструкциями, используемыми в языках программирования и получение позитивного опыта отладки и написания первых завершённых программных продуктов.

Задачи:

Образовательные:

- Рассмотреть основные правила составления и написания алгоритмов и программ;
- Научить современным методам разработки программ: блочному и строчному программированию;
- Научить основным принципам работы в различных средах (Scratch, App inventor, Python);
- Научить ребят грамотно выражать свою идею, выделять персонажей и их функции и действия, реализовать идею в виде законченного мультфильма или игры;

Развивающие:

- Развивать у ребенка умения работать по предложенным стандартам программирования;
- Развивать креативное мышление и пространственное воображение, умение излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений;
- Формировать навыки проектного мышления, работы в команде, эффективно распределять обязанности;

Воспитательные:

- Повышать мотивацию учащихся к изобретательству и созданию собственных законченных произведений;
- Воспитывать у учащихся стремление к получению качественного законченного

результата;

1.3 Планируемые результаты

Предметные результаты:

- развитие логических способностей и алгоритмического мышления, умений составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами — линейной, условной и циклической;
- развитие представлений о числах, числовых системах;
- овладение символьным языком математики, умение составлять и использовать сложные алгебраические выражения для моделирования учебных проектов, моделировать реальные ситуации на языке математики;
- осознание значения математики и информатики в повседневной жизни человека;
- формирование представлений об основных предметных понятиях — «информация», «алгоритм», «модель» и их свойствах;
- формирование информационной и алгоритмической культуры, развитие основных навыков использования компьютерных устройств и программ;
- развитие пространственных представлений, навыков геометрических построений и моделирования таких процессов, развитие изобразительных умений с помощью средств ИКТ;

Метапредметные результаты:

- умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;
- умение оценивать правильность решения учебно-исследовательской задачи;
- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата, понимая, что в длинная программа не гарантирует лучший результат;
- владение основами самоконтроля, принятия решений;
- умение сотрудничества и совместной деятельности со сверстниками в процессе проектной и учебно-исследовательской деятельности.

Личностные результаты:

- формирование ответственного отношения к учению, способности к саморазвитию;
- развитие осознанного и ответственного отношения к собственным поступкам, повышение уровня самооценки, благодаря реализованным проектам;

- формирование коммуникабельности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности;

1.4. Формы организации образовательной деятельности

Занятия проводятся 2 раза в неделю, по 90 минут с обязательным 10-ти минутным перерывом через 45 минут занятия.

Группы формируются по возрастному принципу.

1 год обучения – 7-9 лет,

2 год обучения - 8-10 лет.

3 год обучения – 9-12 лет.

Срок реализации программы – 3 года.

Общее количество учебных часов, запланированных на весь период обучения: 126 часов.

Количество учебных часов в год: 42 часов.

Зачисление на обучение по программе осуществляется в течение всего учебного года по заявлению родителей (законных представителей) без предварительных творческих испытаний.

2. Содержание программы

1-й год обучения

2.1. Учебно-тематический план

№	Тема занятия	Количество часов			Практическое задание
		Всего	Теоретических	Практических	
Модуль 1. Знакомство с миром алгоритмов и программ.					
1	Понятие алгоритмов. Линейные алгоритмы	1,5	0,5	1	Прибытие на базу альянса
2	Команды движения, координатная плоскость, работа с костюмами	1,5	0,5	1	Исследование подземелий
3	Координаты, растровая графика и векторная графика, движение	1,5	0,5	1	Курс управления космолетом
4	Линейные алгоритмы, координаты, спрайты и их свойства	1,5	0,5	1	Контрольное задание для рейнджеров
Модуль 2. Основы логики нелинейных алгоритмов и программ.					
5	Использование циклов с фиксированным количеством	1,5	0,5	1	Важный след

	повторений				
6	Закрепление оператора цикла. Циклы с неограниченным числом повторений и их особенности.	1,5	0.5	1	Пробить путь
7	Алгоритм ветвление. Оператор условия, формы оператора условия	1,5	0.5	1	Прилет на планету Азир
8	Ветвление, неполное ветвление и случаи их применения	1,5	0.5	1	телепорты
9	Закрепление знаний о изученных операторах	1,5	0.5	1	Погоня пиратов
Модуль 3 . Движение объекта и управление им.					
10	Способы задания движения объектов на плоскости. Мгновенное и плавное перемещение	1,5	0.5	1	Ремонтные работы
11	Вращение объектов, определение и изменение углов поворота в градусах	1,5	0.5	1	
12	Создание программы управления объектом на плоскости	1,5	0.5	1	
13	Сенсоры касания объектов и цветов	1,5	0.5	1	Защитные меры
14	Циклы с условием и варианты их применения	1,5	0.5	1	Астероидные дожди
15	Создание множества спрайтов по их модели. Вложенные циклы и их особенности	1,5	0.5	1	Клоны
16	Блоки-активаторы программ по различным событиям	1,5	0.5	1	Сопровождение каравана
17	Запуск и остановка программ по получению сообщений	1,5	0.5	1	Караван
Модуль 4. Математика игр и приложений					
18	Логические переменные и операторы сравнения чисел	1,5	0.5	1	Космическая трасса
19	Логические операторы “И”, “ИЛИ” , “НЕ”	1,5	0.5	1	Снова порталы

20	Математические, логические операторы и их совместное применение	1,5	0.5	1	Стражи порядка
21	Переменные. Создание шифров и их расшифровка	1,5	0.5	1	Шифрование данных
22	Закрепление знаний по теме “математические и логические операторы”	1,5	0.5	1	Устранение поломки
23	Функции. Создание специальных блоков (функций)	1,5	0.5	1	На краю альянса
24	Расширения. Разнообразие подключаемых наборов блоков. Генератор речи и переводчик	1,5	0.5	1	Пограничная охрана
Модуль 5. Создание проектов игр в scratch					
25	Обсуждение идей и подготовка набросков проектов	1,5	0.5	1	Вперед к подвигам
26	Оформление графической части будущих игр	1,5	0.5	1	
27	Создание алгоритма работы игры и соответствующих скриптов	1,5	0.5	1	
28	Тесты и проработка ошибок (багов)	1,5	0.5	1	
29	Презентация своих работ другим ученикам	1,5	0.5	1	
30	Подведение итогов проектной работы	1,5	0.5	1	
	Всего часов	45	15	30	

2.2.Содержание учебно-тематического плана

Модуль 1. Знакомство с миром алгоритмов и программ.

1.Понятие алгоритмов. Линейные алгоритмы

Теория: История Понятие исполнителя, алгоритма и программы, их назначение, виды и использование. Виды управления исполнителем. Способы записи алгоритма. Основные характеристики исполнителя. Система команд исполнителя Понятие проект, его структура и реализация в среде Scratch.

Практика: составление первых линейных алгоритмов движения .
(упражнение “Прибытие на базу альянса”)

2. Команды движения, координатная плоскость, работа с костюмами

Теория: Сцена. Ширина и высота сцены. Текущие координаты объекта. Редактирование текущего костюма. Плавное и мгновенное движение. Команды – идти; повернуться направо (налево); повернуть в направлении; повернуться к; изменить x (y) на; установить x (y) в.

Практика: составление продвинутых линейных алгоритмов с использованием дополнительных функций управляемых объектов.
(упражнение “Исследование подземелий”)

3. Координаты, растровая графика и векторная графика, движение

Теория: Стандартный объект. Спрайты. Список спрайтов. Редактор рисования для создания новых спрайтов. Инструменты рисования (кисточка, линия, текст, эллипс) и редактирования объекта (ластик, заливка, поворот, выбор, печать, пипетка). Центрирование костюма. Масштабирование и движение спрайта. Загрузка на сцену спрайтов из стандартной коллекции среды Scratch. Вставка спрайтов из файлов форматов JPG, BMP, PNG, GIF. Выбор случайного спрайта. Удаление спрайтов .

Практика: Создание сцены и прорисовка основных спрайтов для космической Scratch-истории.
(упражнение “Курс управления космолетом”)

4. Линейные алгоритмы, координаты, спрайты и их свойства

Теория: Команды – “если край”, “оттолкнуться”. Принципиальное различие действия команд “идти в” и “плыть в”. Назначение сенсоров положение x, положение y и направление.

Практика: создание программ для передвижения спрайтов по сцене.
(упражнение “Контрольное задание для рейнджеров”)

Модуль 2. Основы логики нелинейных алгоритмов и программ.

5. Использование циклов с фиксированным количеством повторений

Теория: функциональность работы циклов. Цикличность выполнения действий в зависимости от поставленных условий. Циклы с фиксированным числом повторений. Заголовок цикла. Тело цикла. Циклы с условным оператором. Заголовок цикла. Предусловие и постусловие.

Практика: создание программ с использованием циклов с фиксированным числом повторений. Создание программ с использованием циклов с предусловием и постусловием.
(упражнение “ Важный след ”)

6.Закрепление оператора цикла. Циклы с неограниченным числом повторений и их особенности.

Теория: работа циклов с неограниченным числом повторений. Цикличность выполнения действий в зависимости от поставленных условий. Заголовок цикла. Циклы с условным оператором. Заголовок цикла. Тело цикла. Предусловие и постусловие. Зацикливание.

Практика: создание программ с использованием циклов с неограниченным числом повторений.

(упражнение “ Пробить путь ”)

7.Алгоритм ветвление. Оператор условия, формы оператора условия

Теория: использование в программах условных операторов. Базовая конструкция ветвление, назначение, виды (полная и неполная форма). Понятие условия. Изменение порядка выполнения скриптов в зависимости от условия. Скрипты условных операторов. Использование полной формы ветвления в системе Scratch.

Практика: создание программ с изменением последовательного выполнения скриптов при наличии условий.

(упражнение “ Прилет на планету Азир”)

8.Ветвление, неполное ветвление и случаи их применения

Теория: конструкция неполное ветвление, назначение, формы. Изменение порядка выполнения скриптов в зависимости от условия. Разветвление листинга программы. Скрипты условных операторов. Использование неполной формы ветвления в системе Scratch.

Практика: создание программ с нелинейного выполнения скриптов при наличии условий неполного ветвления.

(упражнение “ Телепорты ”)

9.Закрепление знаний о изученных операторах

Теория: закрепление полученных знаний о циклах и ветвлениях. Составление различных комбинаций данных операторов для решения более сложных задач. Циклы с предусловием и постусловием.

Практика: создание программ с комбинацией циклов, а также условий полного и неполного ветвления.

(упражнение “ Погоня пиратов ”)

Модуль 3 . Движение объекта и управление им.

10 Способы задания движения объектов на плоскости. Мгновенное и плавное перемещение

Теория: Команды – идти; изменить x (y) на; установить x (y) в; если край,

оттолкнуться. Принципиальное различие действия команд идти в и плыть в. Назначение сенсоров положение x, положение y и направления.

Практика: Создание своей программы перемещения космического корабля по координатной плоскости.

(упражнение “ Ремонтные работы ” начало)

11 Вращение объектов, определение и изменение углов поворота в градусах.

Теория: Вращение объектов на плоскости на способы задания углов поворота.

Положительные и отрицательные углы поворота. Плавное и мгновенное вращение объекта. Применение вращение совместно с перемещением по координатной плоскости.

Практика: Создание программы вращения космического корабля совместно с существующей программой перемещения.

(упражнение “ Ремонтные работы ” продолжение)

12 Создание программы управления объектом на плоскости

Теория: Понятие сенсора. Правила применения и область действия команд касается.

Сенсоры мышка по x, мышка по y, мышка нажата? ,клавиша...нажата?, расстояние до, перезапустить таймер. Сенсоры, значение которых можно выводить на экран – ответ, таймер, громкость, громко? ...значение сенсора и сенсор.

Практика: Создание программы управления с использованием значений сенсоров и команды спросить.

(упражнение “ Ремонтные работы ” завершение)

13 Сенсоры касания объектов и цветов

Теория: Правила применения и область действия команд касается, касается цвета и цвет касается. Необходимость ввода данных для их обработки в программе. Ввод данных с помощью команды спросить. Вывод конечного результата обработки с помощью команд говорить и сказать.

Практика: Создание программ посадки космического корабля с использованием значений сенсоров цвета.

(упражнение “ Защитные меры”)

14 Циклы с условием и варианты их применения

Теория : Понятие цикла с условием. Изменение порядка выполнения скриптов в зависимости от условия. Разветвление листинга программы. Скрипты условных операторов. Использование формы циклического ветвления в системе Scratch.

Практика: Создание программ управления с постоянной или циклической с постоянным промежутком времени проверкой условия.

(упражнение “ Астероидные дожди ”)

15 Создание множества спрайтов по их модели. Вложенные циклы и их особенности

Теория : Понятие вложенных циклов. Автоматическое создание копий объекта – особенности, преимущества, недостатки. Скрипты составленные для клона и для его прообраза. Использование формы циклического ветвления для программирования клонов в системе Scratch.

Практика: Создание программ клонирования космического корабля с различными видами управления.

(упражнение “ Клоны”)

16 Блоки-активаторы программ по различным событиям

Теория: События в проектах Scratch. Понятие событий, их активаторов и необходимость их использования в листинге программы. Дробление больших скриптов при помощи запуска по событию.

Практика: Разработка сценария Scratch-историй с несколькими событиями.

(упражнение “ Сопровождение каравана ”)

17 Запуск и остановка программ по получению сообщений

Теория: Сообщения в проектах Scratch. Понятие сообщений , их активаторов и необходимость их использования в листинге программы. Линейная, циклическая связь и связь с ветвлением между отдельными скриптами при помощи отправки сообщений.

Практика: Разработка сценария Scratch-историй с несколькими ветвлениями запускаемыми через сообщения.

(упражнение “ Караван ”)

Модуль 4. Математика игр и приложений

18 Логические переменные и операторы сравнения чисел

Теория: Числа. Строинги. Логические величины. Логические выражения. Арифметические операции. Логические операции. Операции сравнения. Использование арифметических и логических блоков в листинге программы. Использование арифметических и логических блоков вместе с блоками управления.

Практика: Создание программ с использованием операций сравнения данных. Создание программ с использованием арифметических данных и логических операций.

(упражнение “Космическая трасса ”)

19 Логические операторы “И”, “ИЛИ” , “НЕ”

Теория: . Логические и арифметические операции. Операции, выполняемые при

помощи блоков “И”, “ИЛИ” , “НЕ”. Использование данных операторов листинге программы. Комбинации логических операторов для получения сложного условия. Наиболее часто используемые логические конструкции.

Практика: Создание программ с использованием операций сравнения данных. Создание программ с использованием арифметических данных и логических операций.

(упражнение “ Снова порталы”)

20 Математические, логические операторы и их совместное применение

Теория: Комбинации математических и логических операторов и их наиболее популярные примеры. Текстовые переменные – строки (строки) . Управление текстовыми переменными и поиск необходимой информации в них – слияние нескольких строк , определение длины строки поиск буквы и ее порядкового номера. Команда “выдать случайное от...до” и ее использование с числами и строками.

Практика: Создание программ с использованием операций над двумя типами данных.

(упражнение “ Стражи порядка ”)

21 Переменные. Создание шифров и их расшифровка

Теория: глобальные и локальные переменные. Имя переменной и правила его формирования. Команды для переменных - поставить...в, изменить...на, показать переменную, спрятать переменную. Удаление переменных. Создание счетчиков с помощью переменных. Простейшие способы шифрования и их применение.

Практика: Создание программ шифрования и расшифровки кода с использование глобальных и локальных переменных.

(упражнение “Шифрование данных ”)

22 Закрепление знаний по теме “математические и логические операторы”

Теория: Различия между типами данных. Логические и арифметические операции. Использование арифметических и логических блоков в листинге программы. Использование арифметических и логических блоков вместе с блоками управления.

Практика: Создание программ с использованием операций над различными типами данных. Создание программ с использованием арифметических данных и логических операций.

(упражнение “ Устранение поломки ”)

23 Функции. Создание специальных блоков (функций)

Теория: Функции. Преимущества и недостатки их применения. Особенности правильного составления собственных блоков (функций) .

Практика: Создание программ с использованием самостоятельно составленных

блоков под конкретную задачу.
(упражнение “ На краю альянса ”)

24 Расширения. Разнообразие подключаемых наборов блоков. Генератор речи и переводчик

Теория: Библиотека расширений в scratch. Подключение и использование расширений “Генератор речи” и ”переводчик”. Автоматическое озвучивание игр различными голосами. Создание интерфейса игры на нескольких языках.

Практика: Создание программ с использованием блоков, загруженных из библиотеки расширений
(упражнение “ Пограничная охрана ”)

Модуль 5. Создание проектов игр в scratch

Общая Практика: поэтапное создание собственного проекта в задуманной тематике с проработкой всех аспектов с позиции будущего игрока и помощью педагога в решении задач повышенной сложности продолжительностью в 6 занятий.

25 Обсуждение идей и подготовка набросков проектов

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

26 Оформление графической части будущих игр

Теория: Разработка и создание основных спрайтов и их костюмов для будущей игры. Разработка скриптов для спрайтов и объектов.

27 Создание алгоритма работы игры и соответствующих скриптов

Теория: Логика создания персонажей для игры. Перевод алгоритма, написанного на естественном языке, в коды Scratch. Односторонний (без возможности вернуться назад) переход из одного пространства в другое. Понятие интерфейса. Элементы интерфейса. Основные принципы дизайна интерфейсов. Обратная связь. Необходимые элементы меню.

28 Тесты и проработка ошибок (багов)

Теория: Правила работы в сети. Интернет-сообщества. Сообщество Scratch. Регистрация на сайте. Использование заимствованных кодов и объектов. Авторские права. Публикация проектов Scratch.

29 Презентация своих работ другим ученикам

30 Подведение итогов проектной работы

2-ой год обучения

2.3.Учебно-тематический план

№	Тема занятия	Количество часов			Практическое задание
		Всего	Теоретических	Практических	
Модуль 1. Знакомство с App inventor. Логика работы приложений для Android.					
1	Знакомство с интерфейсом App inventor	1,5	0,5	1	-
2	Логика условий. Компонент “Кнопка” и его функции.	1,5	0,5	1	Алфавит
3	Математика в App inventor. Численные переменные.	1,5	0,5	1	Кликер
4	Случайные числа и их роль в создании игр.	1,5	0,5	1	Кости игральные
5	Расположение объектов при создании дизайна приложения. Компонент “Звук”	1,5	0,5	1	Приложение DJ
6	Интерфейс при создании игр на 2х и более игроков. Применение текстовых переменных	1,5	0,5	1	Battle
Модуль 2. Приложения с использованием 2D графики					
7	Действия с PNG изображениями. Компонент “Фото”	1,5	0,5	1	Инстаграмщик
8	Свойства объектов. Цвета. Задание цветов при помощи значений RGB.	1,5	0,5	1	Звездный творец
9	Компонент “Холст” и его функции	1,5	0,5	1	Рисовач
10	Применение компонентов вкладки “Рисование и анимация” для создания движущейся сцены	1,5	0,5	1	Самолетик
11	Приёмы разделения на более простые задачи и закрепление пройденного материала	1,5	0,5	1	Танчики
Модуль 3. Приложения с подключением сторонних ресурсов					

12	Компонент “Переводчик” и его функции	1,5	0,5	1	Переводчик
13	Шифрование данных различными графическими способами и их распознавание.	1,5	0,5	1	QR-сканнер
14	Использование компонентов из вкладки “Maps”	1,5	0,5	1	Геопозиция
15	Логика нейросетей и способы их использования	1,5	0,5	1	Искусственный интеллект
16	Массивы и базы данных. Преимущества их использования при большом количестве переменных данных.	1,5	0,5	1	База данных
17	Использование облачных баз данных. Обмен информацией между устройствами	1,5	0,5	1	Чат
Модуль 4. Создание своего приложения от идеи до презентации					
18	Развитие идеи, обсуждение возможных проблем на пути реализации	1,5	0,5	1	-
19	Создание дизайна приложения	1,5	0,5	1	-
20	Проработка кода приложения	1,5	0,5	1	-
21	1й запуск, тесты, поиск ошибок	1,5	0,5	1	-
22	Исправление программ с учетом опыта предыдущего занятия	1,5	0,5	1	-
23	Подготовка презентации своего приложения	1,5	0,5	1	-
24	Презентация проектов друг другу (другим группам)	1,5	0,5	1	-
25	Доработка своих проектов с учетом пожеланий других пользователей.	1,5	0,5	1	-

Модуль 5. 3D моделирование					
26	Знакомство с отраслью 3D моделирования.	1,5	0,5	1	Ваза
27	Построение продвинутой геометрии.	1,5	0,5	1	Предметы мебели
28	Построение модели для 3D печати	1,5	0,5	1	Модель значка для 3D печати
29	Продвинутые структуры и сборки.	1,5	0,5	1	Робот с подвижными элементами
	Всего часов	43,5	14,5	29	

2.4. Содержание учебно-тематического плана

Модуль 1. Знакомство с App inventor. Логика работы приложений для Android.

1 Знакомство с интерфейсом App inventor

Теория: Инструктаж о правилах поведения и особенностями деятельности на будущих занятиях. Организация рабочего места в компьютерном классе. Основные элементы интерфейса среды Mit App Inventor.

Практика: Создание приложения с выводом данных на экран «Привет, мир!»

2 Логика условий. Компонент “Кнопка” и его функции.

Теория: Основные элементы управления: кнопка, текст, надпись. Математические блоки. Основные элементы графического интерфейса Mit App Inventor: кнопка, текст, надпись. Обзор блоков раздела «Математика».

Практика : Создание приложения с использованием нескольких кнопок «Алфавит».

3 Математика в App inventor. Численные переменные.

Теория: Переменные. Математические операторы. Циклические алгоритмы.

Алгоритмы с условием. Построение блок – схем алгоритмов. Блоки управления для реализации циклов с предусловием и с параметром.

Практика: Создание приложения с использованием развитых условий и переменного подсчета очков «Кликер».

4 Случайные числа и их роль в создании игр.

Теория : Акселерометр. Механики использования акселерометра. Генератор случайных чисел. Особенности использования генератора случайных чисел. Игровая механика. Создание игрового приложений с использованием случайного числа.

Практика: Создание приложения, которое имитирует игральный кубик.

5 Расположение объектов при создании дизайна приложения. Компонент “Звук”

Теория: Компонент звук. Компоненты из категорий медиа и интерфейс пользователя. Программирование кнопок для экранной клавиатуры и их расположение.

Практика : создание экранов и написание скриптов для двух приложений, использующих звуки. (приложение “DJ”)

6 Интерфейс при создании игр на 2х и более игроков. Применение текстовых переменных

Теория: Использование одного устройства несколькими людьми. Разветвляющиеся алгоритмы. Построение блок - схем разветвляющихся алгоритмов. Блоки управления «если ...». Блоки для работы со строками. Обзор блоков раздела «Текст».

Практика. Создание приложения для неграфической игры на 2х и более игроков «Battle».

Модуль 2. Приложения с использованием 2D графики

7 Действия с PNG изображениями. Компонент “Фото”

Теория: Работа с камерой мобильного приложения; компоненты "Выборщик изображений", "Публикация"; работа с камерой в мобильном приложении, компоненты для работы с PNG фото, приложения для обмена и передачи фото.

Практика: создание приложения для создания, изменения и обмена фото “Инстаграмщик”

8 Свойства объектов. Цвета. Задание цветов при помощи значений RGB.

Теория: Цвета и их задание в программах. RGB спектр и его значение. Динамическое управление цветом. Базовые цвета RGB системы цветов. Блоки управления

свойствами объектов.

Практика: создание приложения управляющего параметрами звезды при помощи слайдеров (приложение “Звездный творец”)

9 Компонент “Холст” и его функции

Теория: создание графического редактора для рисования графическими примитивами. Способы реализации основного функционала графического редактора. Компоненты использующие холст.

Практика: Создание простого графического редактора для рисования графическими примитивами “Рисовач”

10 Применение компонентов вкладки “Рисование и анимация” для создания движущейся сцены

Теория: Анимация в АИ. Элементы изображения спрайтов, холст, шар и их методы и свойства. Движение объектов на холсте. Приемы при создании анимации.

Практика. Создание анимированных объектов с использованием элементов холст, с возможностью управления ими (приложение “самолетик”).

11 Приёмы разделения на более простые задачи и закрепление пройденного материала

Теория: Построение игровой механики, движение элемента, координаты. Загрузка спрайтов, использование таймера. Трудности решения больших задач. Упрощение и дробление больших программ.

Практика. Создание игры, в которой необходимо управлять персонажем , уничтожать противника и уклоняться от него, с фиксированием времени и количества попаданий.(приложение “Танчики”)

Модуль 3. Приложения с подключением сторонних ресурсов

12 Компонент “Переводчик” и его функции

Теория: Подключение к приложению внешних ресурсов из сети интернет. Web-приложения. Компонент переводчик – подключение, особенности использования и принцип работы. Элемент веб - просмотрщик и его свойства.

Практика: Создание приложения переводчик с возможностью перехода в веб-браузер.

13 Шифрование данных различными графическими способами и их распознавание.

Теория : Шифры. Графические шифры. Методы шифрования и расшифровки. Компонент QR-сканнер - подключение, особенности использования и принцип работы. Создание собственных QR-кодов для личного пользования.

Практика: создание приложения , сканирующего QR-коды .

14 Использование компонентов из вкладки “Maps”

Теория: Сенсор ориентации, сенсор местоположения и группа элементов Maps - подключение, особенности использования и принцип работы. Свойства используемых компонентов и их управление через блоки кода.

Практика: Создание приложения для ориентации и навигации на местности.

15 Логика нейросетей и способы их использования

Теория : Искусственный интеллект и его виды. Логика работы программных нейронов и нейросетей в целом. Машинное обучение и факторы влияющие на его быстроту и качество. Задачи для которых целесообразно применять нейросети.

Практика: Создание нейросети при помощи онлайн сервиса, способной различать предметы и лица на фото.

16 Массивы и базы данных. Преимущества их использования при большом количестве переменных данных.

Теория: Структуры данных: массив и Dictionary. Обзор блоков группы Dictionary. Структурирование больших объемов информации. Виды массивов и данных хранимых внутри.

Практика (3 часа). Создание приложения “база данных”, которое будет вести заметки пользователей через облачное хранилище .

17 Использование облачных баз данных. Обмен информацией между устройствами Чат

Теория: Хранение данных на сервере. Передача и прием информации по сети.

Особенности работы чата. Облачные хранилища – преимущества и недостатки перед локальным хранением данных.

Практика: Создание мессенджера для общения между учениками группы.

Модуль 4. Создание своего приложения от идеи до презентации

18 Развитие идеи, обсуждение возможных проблем на пути реализации

Теория: Введение в проектную деятельность. Цели, задачи проекта. Этапы работы над проектом. Правила эффективного планирования. Исследование предметной области проекта. Приемы поиска информации в сети интернет.

Практика: Выбор темы, определение целей и задач проекта. Составление плана работы над проектом. Подбор и изучение материалов о выбранной предметной области.

19 Создание дизайна приложения

Теория: Создание макетов приложения. Правила создания дружелюбного интерфейса. Правила гармоничного расположения элементов. Сочетания элементов расположения для точной установки компонентов дизайна.

Практика : Подготовка изображений, текстов и другого контента для проекта. Создание макетов приложения.

20 Проработка кода приложения

Практика: Программирование логики программ. Программирование функционала согласно задуманной идее.

21 1й запуск, тесты, поиск ошибок

Практика: Тестирование, поиск ошибок в работе приложения для итогового проекта. Проверка корректности работы задуманных механик.

22 Исправление программ с учетом опыта предыдущего занятия

Практика: Отладка ошибок работы проектного приложения отмеченных на предыдущем занятии. Корректировка баланса в случае если приложение является игрой.

23 Подготовка презентации своего приложения

Теория: Состав презентации проекта. Приемы грамотной подачи материала о своем приложении. Методы привлечения и удержания аудитории.

Практика: Подготовка речи и презентации для защиты итогового проекта.
Демонстрация программы.

24 Презентация проектов друг другу (другим группам)

Практика: выступление с готовым проектом перед аудиторией.

25 Доработка своих проектов с учетом пожеланий других пользователей.

Теория: подведение итогов проектной деятельности разбор ошибок и удачных решений.

Практика: доработка проекта на основании полученного опыта.

Модуль 5. 3D моделирование

26. Знакомство с отраслью 3D моделирования.

Теория: Виды 3D редакторов в зависимости от предполагаемой области применения. Параметрическое и полигональное моделирование. Трехмерные координаты. Эскизы. Тела линейного выдавливания и тела вращения

Практика: построение первой модели объекта несложной геометрии - модель вазы

27. Построение продвинутой геометрии.

Теория: Продвинутые элементы эскиза. Разложение сложных форм на примитивы. Булевы операции при наложении твердотельных объектов.

Практика: построение объекта несложной геометрии по натуральному образцу (в качестве образца берется предмет мебели с линейными формами)

28. Построение модели для 3D печати

Теория: принципы 3D печати. Особенности построения и подготовки моделей для 3D печати. Установка поддержек при печати сложных форм. Нарезка модели на слои, установка параметров печати и перевод в G-code, используемый принтером.

Практика: создание 3D моделей памятных значков с собственным дизайном и подготовка их к печати.

29. Продвинутые структуры и сборки.

Теория : Сборки из нескольких деталей. Подвижные сборки. Способы изменения взаимного расположения деталей в сборке. Использование готовых деталей из библиотеки стандартных изделий.

Практика: создание подвижной сборки : работа способного передвигать частями тела.

3-й год обучения

2.5.Учебно-тематический план

№	Тема занятия	Количество часов			Практическое задание
		Всего	Теоретических	Практических	
Модуль 1. Введение в Python. Знакомство с интерфейсом и логикой построения программ.					
1	Установка среды программирования и знакомство с ней. Ввод и вывод данных	1,5	0,5	1	Вывод текстовых сообщений
2	Функции. Математические функции. Установка и создание собственных модулей в Python.	1,5	0,5	1	Калькулятор
3	Условия и циклы в Python	1,5	0,5	1	Викторина
4	Случайные числа. Списки, словари и их использование в программах	1,5	0,5	1	Квест
Модуль 2. Создание 2D игр на Python					
5	Знакомство с модулем pygame и изучение его возможностей	1,5	0,5	1	Установка модуля и создание 1го окна игры
6	Настройка окна игры под различные условия. Создание фона и объектов в окне игры	1,5	0,5	1	Змейка
7	Способы задания управления объектом. Управление змейкой и ее длиной	1,5	0,5	1	
8	Задание условий появления бонусов, начисления очков, победы и поражения	1,5	0,5	1	
9	Закрепление материала по созданию окна игры и объектов	1,5	0,5	1	Арканойд

10	Физика движения упругого мяча в игре и его траектория	1,5	0,5	1	
11	Закрепление материала по созданию циклов и условий. Завершение игры Арканойд	1,5	0,5	1	
Модуль 3. Создание 3D игр на Python в Minecraft					
12	Знакомство со средой программирования IDLE, подключение IDLE к Minecraft	1,5	0,5	1	Создание своего сервера и его 1й запуск
13	Ориентация в 3D координатах. Определение и изменение координат.	1,5	0,5	1	определение координат главного героя и управление им
14	Создание и изменение положения блоков внутри игры при помощи кода.	1,5	0,5	1	Сцена, декорации и препятствия для игры лабиринт
15	применение пройденных логических конструкций и функций для задания условий игры	1,5	0,5	1	Лабиринт
16	Создание усложненных геометрических форм при помощи библиотек	1,5	0,5	1	Создать шар и многогранник из любимого материала
17	Искусственный интеллект (ИИ). Программа движения внутриигровых персонажей	1,5	0,5	1	Пес напарник с ИИ
18	Совместное использование ИИ и случайной генерации чисел	1,5	0,5	1	Игра прятки
Модуль 4. Создание виртуальных собеседников (чат-ботов) на Python					
19	Знакомство с виртуальными собеседниками. Создание и настройка бота.	1,5	0,5	1	Создание бота в Telegram. Написание первой программы для бота
20	Программирование ветвлений составляющих диалог пользователя с виртуальным собеседником (чат-ботом).	1,5	0,5	1	Telegram-бот. Собеседник: диалог с ботом
21	Оптимизация и доработка кода.	1,5	0,5	1	Telegram-бот. Кнопки
22	Подключение внешних ресурсов к программе. Использование данных со сторонних источников.	1,5	0,5	1	Telegram-бот. Программа для поиска в Wikipedia

Модуль 5. Создание Python - проекта					
23	Создание Python - проекта	1,5	0	1,5	Создание Python - проекта
24		1,5	0	1,5	
25		1,5	0	1,5	
	Всего часов	37,5	11	26,5	

2.6. Содержание учебно-тематического плана

Модуль 1. Введение в Python. Знакомство с интерфейсом и логикой построения программ.

1 Установка среды программирования и знакомство с ней. Ввод и вывод данных

Теория: Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python. Комментарии. Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран.

Практика: первая программа - вывод текстовых сообщений.

2 Функции. Математические функции. Установка и создание собственных модулей в Python.

Теория: Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова. Выражения. Операции. Порядок выполнения операций. Математические функции. Композиция. Создание функций. Параметры и аргументы. Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Примеры решения задач с использованием функций.

Практика: написание программы “Калькулятор”, производящей действия сложения, вычитания, умножения и деления.

3 Условия и циклы в Python

Теория: Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Примеры решения задач с условным оператором. Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов. Оператор цикла с параметром for. Операторы управления циклом.

Практика: написание программы “ Викторина”, задающей вопросы и считающей правильные ответы.

4 Случайные числа. Списки, словари и их использование в программах

Теория: Случайные числа. Модуль random. Списки. Тип список (list). Индексы. Обход списка. Проверка вхождения в список. Добавление в список. Суммирование или изменение списка. Операторы для списков. Срезы списков. Удаление списка. Клонирование списков. Списочные параметры. Функция range.. Генераторы списков в Python.

Практика: написание программы “Квест”- текстовой игры со непредсказуемым поведением ИИ .

Модуль 2. Создание 2D игр на Python

5 Знакомство с модулем pygame и изучение его возможностей

Теория: Игры с 2D графикой. Способы реализации двумерных динамических сцен. Импорт модуля pygame. Функционал, возможности и ограничения применения pygame. Управление инициализацией и работой окна 2D визуализации.

Практика: установка модуля и создание 1го окна игры.

6 Настройка окна игры под различные условия. Создание фона и объектов в окне игры

Теория: геометрические примитивы и анимация. Управление элементами. Загрузка изображений в игру. Настройка размеров области вывода данных. Функции обновления экрана update и flip.

Практика: проект игры “Змейка” , первый этап – установка необходимых параметров экрана вывода.

7 Способы задания управления объектом. Управление змейкой и ее длиной

Теория: Динамическая прорисовка геометрических фигур. Управление движением объектов через переменные изменения координат. Взаимодействие элементов в игре через условия. Клонирование элементов в игре.

Практика: проект игры “Змейка”, второй этап – создание управления движением змейки через клавиши клавиатуры и механик поедания еды.

8 Задание условий появления бонусов, начисления очков, победы и поражения

Теория: Применение логических и численных переменных, а также операторов при реализации игровых механик. Оператор If и его применение к координатам объектов. Баланс игрового процесса игр рассматриваемого формата.

Практика: проект игры “Змейка”, третий этап – создание системы бонусов и подсчета очков, а также реализация смерти по установленным правилам.

9 Закрепление материала по созданию окна игры и объектов

Теория (закрепление полученных знаний по темам): настройка размеров области вывода данных. Функции обновления экрана update и flip. Геометрические примитивы и их перемещение. Управление примитивами. Загрузка изображений в игру.

Практика: проект игры “Арканоид”, первый этап – создание экрана игры с необходимыми параметрами и объектов: шара, блоков, ракетки.

10 Физика движения упругого мяча в игре и его траектория

Теория: Физика движения тел в пространстве. Упругий и неупругий удар. Импульс и механическая энергия. Функции описывающие поведение упруго тела. Базовые элементы кинематики и динамики.

Практика: проект игры “Арканоид”, второй этап – составление программы движения шара без прямого управления им через устройства ввода.

11 Закрепление материала по созданию циклов и условий. Завершение игры Арканоид

Теория (закрепление полученных знаний по темам): виды переменных и их применение, математические и логические операторы при реализации игровых механик. Оператор If и его применение к координатам объектов.

Практика: проект игры “Арканоид”, третий этап – составление программы начисления очков, победы и проигрыша в игре.

Модуль 3. Создание 3D игр на Python в Minecraft

12 Знакомство со средой программирования IDLE, подключение IDLE к Minecraft

Теория: установка компонентов, необходимых для совместного использования языка программирования Python и игры Minecraft в качестве средства визуализации и графического редактора. Среда программирования IDLE и ее интерфейс. Сервера: назначение и способы создания.

Практика: создание своего сервера и его 1й запуск.

13 Ориентация в 3Д координатах. Определение и изменение координат.

Теория: трехмерная система координат и способы ориентации в ней. Декартовы, цилиндрические и сферические координаты. Команды определения координат главного героя и иных объектов. Команды задания и изменения координат.

Практика: определение координат главного героя и управление им через строки кода.

14 Создание и изменение положения блоков внутри игры при помощи кода

Теория: Основы создания трехмерного дизайна локаций для игр. Многоуровневые локации и их внутренние переходы. Создание и удаление больших объемов блоков строками кода. Формы записи команд для создания геометрических примитивов.

Практика: Создание сцены, декораций и препятствий для игры лабиринт.

15 применение пройденных логических конструкций и функций для задания условий игры

Теория: Гейм-дизайн как независимое направление деятельности для разработчиков игр. Приемы повышения и понижения сложности игр. Системы механик, создающих разнообразие игрового опыта. Система бонусов внутри игр.

Практика: Создание собственных правил и игровых механик для игры лабиринт на основании имеющихся знаний о циклах и ветвлениях.

16 Создание усложненных геометрических форм при помощи библиотек

Теория: подключение библиотек, помогающих создавать сложные нелинейные фигуры. Наиболее популярные трехмерные геометрические формы и способы их построения в Python + Minecraft. Математические функции, описывающие форму шара, цилиндра и конуса.

Практика: Создать шар и многогранник из любимого материала.

17 Искусственный интеллект (ИИ). Программа движения внутриигровых персонажей

Теория: Искусственный интеллект построенный на множестве ветвлений – преимущества и недостатки в сравнении и нейросетями. Команды реализующие управление персонажами помимо главного героя. Изменение предустановленного искусственного интеллекта.

Практика: Создать программу описывающую искусственный интеллект пса напарника для главного героя.

18 Совместное использование ИИ и случайной генерации чисел

Теория: Добавление генератора случайных чисел в логику принятия решений искусственного интеллекта построенного на множестве ветвлений. Команды, исключающие неприемлемое в данных условиях игры поведение искусственного интеллекта. Баланс сложности противника с ИИ.

Практика: Создать программу, реализующую игру в прятки с персонажами, использующими ИИ.

Модуль 4. Создание виртуальных собеседников (чат-ботов) на Python

19 Знакомство с виртуальными собеседниками. Создание и настройка бота. Создание бота в Telegram.

Теория : Использование мессенджеров программистами для решения индивидуальных задач. Подключение Telegram к программированию на Python. Библиотека telegrambotapi – возможности и простейшие команды.

Практика: Написание первой программы для бота, реализующей вывод сообщений.

20 Программирование ветвлений составляющих диалог пользователя с виртуальным собеседником (чат-ботом).

Теория : Условия в чат-боте. Функции и возврат функций в чат-боте.. Ветки чат-бота.

Практика: Написание программы “ Telegram-бот. Собеседник: диалог с ботом”

21 Оптимизация и доработка кода.

Теория : Способы оптимизации и доработки кода. inline url кнопки чат-бота. Обычные кнопки чат бота. Работа списков в чат-боте. Типы файлов используемых пользователями. Работа с файлами в чат-боте.

Практика: Доработка программы предыдущего занятия с добавлением “ Telegram-бот. Кнопки ”

22 Подключение внешних ресурсов к программе. Использование данных со сторонних источников.

Теория : Использование ссылок на web-ресурсы. Использование информации и обмен данными со внешними ресурсами. Многообразие библиотек для чат-бота.

Практика: Написание программы “ Telegram-бот. Программа для поиска в Wikipedia” или доработка программы предыдущего занятия с добавлением команд, соответствующих теме занятия.

Модуль 5. Создание Python - проекта

3. Формы контроля и оценочные материалы

Реализация Программы предусматривает входной, текущий (промежуточный) контроль и итоговую аттестацию обучающихся.

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

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

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

Параметры контроля

Контроль освоения обучающимися программы осуществляется путем оценивания промежуточных и итоговых проектов по следующим критериям (параметрам):

- Четкость формулировки целей и задач
- Технологическая сложность проекта
- Новизна и оригинальность решения
- Качество разработанного продукта
- Умение работать в команде

Результативность обучения дифференцируется по трем уровням: низкий, средний, высокий.

Высокий уровень – учащийся глубоко изучил учебный материал, последовательно и исчерпывающе отвечает на поставленные вопросы, задание выполняет правильно, уверенно и быстро; владеет логическими операциями, умеет выделять существенные признаки и выделяет самостоятельно закономерности; хорошо ориентируется в изученном материале, может самостоятельно найти нужный источник информации, умеет самостоятельно наблюдать и делать простые выводы; проявляет активный интерес к деятельности, стремится к самостоятельной творческой активности, самостоятельно занимается дома, помогает другим, активно участвует в конкурсах, проявляет доброжелательность.

Средний уровень – учащийся знает лишь основной материал, на заданные вопросы отвечает недостаточно четко и полно, при выполнении практической работы испытывает затруднения, устраняет отдельные неточности с помощью дополнительных вопросов педагога, может допускать ошибки, не влияющие на результат; владеет логическими операциями частично, группирует по

несущественным признакам; не всегда может определить круг своего незнания и найти нужную информацию в дополнительных источниках; понимает различные позиции других людей, но не всегда проявляет доброжелательность, дает обратную связь, когда уверен в своих знаниях, проявляет интерес к деятельности, настойчив в достижении цели, проявляет активность только при изучении определенных тем или на определенных этапах работы.

Низкий уровень – учащийся не может достаточно полно и правильно ответить на поставленные вопросы, имеет отдельные представления об изученном материале, при выполнении практической работы задание или не сделано, или допущены ошибки, влияющие на результат; логические операции не сформированы; самостоятельно не может определять круг своего незнания, не может делать самостоятельные выводы; редко понимает и принимает позицию других людей, считая свое мнение единственно верным, присутствует на занятиях, но не активен, выполняет задания только по четким инструкциям и указаниям педагога.

4. Организационно-педагогические условия реализации программы

Реализация Программы предполагает следующие *формы организации образовательной деятельности*:

- беседы
- игры
- практические занятия
- самостоятельная работа
- викторины
- проекты.

Использование метода проектов позволяет обеспечить условия для развития у ребят навыков самостоятельной постановки задач и выбора оптимального варианта их решения, самостоятельного достижения цели, анализа полученных результатов с точки зрения решения поставленной задачи. Программой предусмотрены методы обучения: объяснительно-иллюстративные, частично-поисковые (вариативные задания), творческие, практические. Личностные и метапредметные результаты освоения курса внеурочной деятельности.

При реализации программы используются следующие **методы и приемы обучения**:

- 4.1.1. наблюдение, демонстрация наглядных пособий, показ примеров программ и способов действий;
- 4.1.2. рассказ педагога, беседы, вопросы, пояснения, объяснения;
- 4.1.3. игровое действие, создание игровой ситуации;
- 4.1.4. моделирование, экспериментирование.

Методы, в основе которых лежит уровень деятельности детей: репродуктивные (ученики воспроизводят полученные знания, на примере

упражнений); поисковые (ученики самостоятельно ищут методы решения задач при выполнении проектного задания)

Условия организации образовательного процесса:

Компьютерный кабинет с персональным компьютером для каждого учащегося, с установленной операционной системой Windows и офлайн-редактором Scratch; проектор или иной мультимедийный экран, локальная сеть, доступ к сети Интернет.

Учебно-информационное обеспечение программы:

Рабочие тетради, интернет-источники, содержащиеся на сайтах, рекомендованных педагогам, реализующим программу

Литература:

1. Краля Н. А. Метод учебных проектов как средство активизации учебной деятельности учащихся: Учебно-методическое пособие / Под ред. Ю. П. Дубенского. Омск: Изд-во ОмГУ, 2005. – 9 с.

2. Матвеева Н. В. Информатика и ИКТ. 3 класс: методическое пособие / Н.В.Матвеева, Е. Н. Челак, Н. К. Конопатова, Л. П. Панкратова. М.: БИНОМ. Лаборатория знаний, 2009. – 420 с.

3. Матяш Н. В. Психология проектной деятельности школьников в условиях технологического образования / Под ред. В. В. Рубцова. Мозырь: РИФ «Белый ветер», 2000. – 285 с.

4. Пашковская Ю.В. Творческие задания в среде Scratch: рабочая тетрадь для 5-6 классов/ Ю.В. Пашковская. – М.: БИНОМ. Лаборатория знаний, 2014.

5. Патаракин Е. Д. Учимся готовить в среде Скретч (Учебно-методическое пособие). М: Интуит.ру, 2008. – 61 с.

6. Рындак В. Г., Дженжер В. О., Денисова Л. В. 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.: ил.

7. Гриффитс Дэвид, Гриффитс Дон. Head First. Программирование для Android. 2-е изд.— СПб, Питер, 2018.

8. Дейтел П., Дейтел Х., Уолд А. Android для разработчиков. 3-е издание (2016)

9. Харди Б., Филипс Б., Стюарт К., Марсикано К. Android. Программирование для профессионалов 2-е изд.— СПб: Питер, 2016.

Интернет-ресурсы:

1. developer.android.com
2. guides.codepath.com/android
3. scratch.mit.edu

Кадровое обеспечение программы

Для реализации программы требуется педагог, обладающий профессиональными знаниями в области математики и информатики

