

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Контроль знаний, умений и навыков

Формой подведения итогов реализации дополнительной образовательной программы выступает текущая, промежуточная и итоговая аттестация.

Образовательный процесс осуществляется на основании учебного плана и регламентируется расписанием занятий для каждой учебной группы.

В целях оценки показателей знаний, умений и навыков обучающихся по дополнительной образовательной общеразвивающей программе «Основы программирования в Майнкрафте» проводится текущий и промежуточный контроль знаний, а также итоговая аттестация.

Виды текущего контроля:

- устный ответ на поставленный вопрос;
- проверка результатов выполнения практических заданий.

Виды промежуточного контроля:

- тестирование устное/письменное/с помощью электронных форм
- проверка результатов выполнения практических работ/проектов по итогам блока занятий.

Тестирование — это форма измерения знаний обучающихся, основанная на применении тестов. Материалы для промежуточного и итогового тестирования предоставляются вместе с комплектом учебно-методических материалов к программе.

Итоговая аттестация

Итоговая аттестация проводится с целью установления уровня знаний обучающихся с учетом прогнозируемых результатов обучения и требований к результатам освоения образовательной программы.

Итоговая аттестация обучающихся осуществляется в форме защиты проекта с демонстрацией результатов — своя цифровая вселенная внутри мира Minecraft по выбранной тематике, которая сопровождается презентацией. Презентация – это краткий устный рассказ о процессе создания игрового мира по предложенному плану.

Выдача обучающимся документов о дополнительном образовании (сертификат о прохождении курса) осуществляется при условии успешного прохождения итоговой аттестации.

Индивидуальный предприниматель Попов А.В.

УТВЕРЖДАЮ

Индивидуальный предприниматель

_____/Попов А.В.//

**Дополнительная общеобразовательная общеразвивающая программа
технической направленности
«Основы программирования в Майнкрафте»
(Программа реализуется в объеме 40 академических часов)**

Оглавление

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	2
1.1. Общая характеристика программы	2
1.2. Цели и задачи программы	3
1.3. Планируемые результаты обучения	4
2. УЧЕБНЫЙ ПЛАН ПРОГРАММЫ	5
3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК	7
4. РАБОЧАЯ ПРОГРАММА	10
5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ	16
5.1. Контроль знаний, умений и навыков	16
5.2. Критерии оценивания освоения программы при проведении различных форм контроля:	17
6. УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ (УЧЕБНИКИ, РАЗДАТОЧНЫЕ МАТЕРИАЛЫ, МЕТОДИЧЕСКИЕ УКАЗАНИЯ, ПЛАКАТЫ, СЛАЙДЫ, ИНТЕРНЕТ-РЕСУРСЫ)	18
6.1. Информационные и учебно-методические условия реализации программы	18
6.2. Рекомендованная литература для обучающихся	18
6.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», рекомендованных для освоения программы	19
7. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ...	20
7.1. Материально-техническая и ресурсная база	20
7.2. Кадровое обеспечение программы	20

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1.1. Общая характеристика программы

Данный документ описывает комплекс основных характеристик образования (объем, содержание, планируемые результаты) и организационно-педагогических условий, который представлен в виде учебного плана, календарного учебного графика, рабочих программ учебных предметов, иных компонентов, а также оценочных и методических материалов общеобразовательной общеразвивающей программы «Основы программирования в Майнкрафте» для дистанционного обучения.

В ходе дополнительной общеобразовательной общеразвивающей программы «Основы программирования в Майнкрафте» обучающиеся научатся программировать в Майнкрафте, создавать конструкции в мире Майнкрафт, работать с использованием алгоритмов, циклов, событий и координат, работать с числовыми и строковыми переменными, функциями, массивами, использовать подходы искусственного интеллекта, понимать базовый синтаксис JavaScript.

Выдача обучающимся документов о дополнительном образовании (сертификат установленного образца) осуществляется при условии успешного прохождения итоговой аттестации.

Программа разработана на основе следующих **нормативных документов**:

- Федеральный закон от 29 декабря 2012 г. №273-ФЗ «Об образовании в Российской Федерации»;
- Приказ Министерства просвещения РФ от 9 ноября 2018 г. N 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»
- Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи» (вместе с «СП 2.4.3648-20. Санитарные правила...»)» (Зарегистрировано в Минюсте России 18.12.2020 № 61573).

Направленность (профиль) программы: техническая.

Актуальность программы, соответствие государственному, социальному заказу/запросам:

В современном мире умение программировать является **ценным и востребованным навыком**, который все больше и больше ценится на рынке, особенно сегодня, в условиях ограниченных ресурсов и повышенных расходов. В результате прохождения обучения слушатель сформирует знания в области программирования в игре Майнкрафте и алгоритмов работы искусственного интеллекта, активизирует навыки использования полученных знаний и умений в практической деятельности, разовьет логическое мышление, проявит интерес к программированию, раскроет свои способности в сфере IT-технологий. Обучающимся будут созданы оптимальные условия для всестороннего удовлетворения потребностей и развития их индивидуальных склонностей и способностей, появится мотивация личности к познанию и творчеству. Обучение по данной программе будет полезно для тех, кто интересуется

развитием в области научно-технической направленности, будущих программистов, аналитиков, разработчиков.

Отличительные особенности программы: по окончании обучения учащиеся с помощью полученных на курсе знаний и навыков создадут свою цифровую вселенную внутри игры, с учетом своих интересов. На занятиях ребенок погружается в мир IT, осваивает азы программирования через игру Майнкрафт и изучает алгоритмы работы искусственного интеллекта. Обучение проходит на платформе Minecraft Education, где ребята напишут свою первую программу из визуальных блоков для собственного робота-помощника, который будет создавать постройки.

Срок обучения: программа реализуется в объеме 40 академических часов, 40 недель (10 месяцев).

Режим занятий: 1 академических час в неделю

Продолжительность академического часа – 45 минут.

Занятия проводятся для детей возрастом от 7 до 15 лет

Продолжительность занятий в учебные дни - не более 2-х академических часов в день, в выходные и каникулярные дни - не более 2 академических часов в день.

Форма обучения: Дистанционный формат

Адресат программы и примерный портрет слушателя курсов: программа разработана для учащихся от 7 до 15 лет, которые хотят сделать первые шаги в программировании и научиться создавать свои игры. К освоению дополнительной общеобразовательной программы – дополнительной общеразвивающей программы допускаются лица без предъявления требований к уровню образования.

По завершении реализации программы, как правило, проводится анкетирование обучающихся с целью изучения мнения по вопросу эффективности и информативности проведенного обучения, уровню организации учебного процесса, удовлетворенности учебно-методическим материалом, работниками образовательной организации проводится анализ высказанных предложений и пожеланий.

1.2. Цели и задачи программы

1. **Цель программы** – получение обучающимися знаний и навыков по разработке программного обеспечения, получение визуального представления о программировании и подходах искусственного интеллекта, знакомство с фундаментальными структурами языков программирования, развитие навыков проектной и командной работы, логического и творческого мышления.

Задачи программы:

1. Сформировать системные знания в области компьютерных технологий и разработки программ.
2. Предоставить специализированные практические навыки от основ программирования до построения своей игровой вселенной в Minecraft и применения подходов искусственного интеллекта.
3. Сформировать интерес к увлечению программированием и раскрытию своих способностей в сфере IT-технологий.

4. Научить создавать свои программы в среде графического программирования MakeCode, конструкции в Minecraft и программы на JavaScript.
5. Сформировать представление о профессии программиста.
6. Сформировать навыки совместной работы над проектами, развивая лидерские качества, творческий подход, а также навыки тайм-менеджмента, решения задач и принятия решений.
7. Предоставить возможность программировать проект в команде с использованием всех полученных знаний и наработанных на курсе навыков.
8. Научить создавать проект в виде своей цифровой вселенной внутри игры и презентовать его.

1.3. Планируемые результаты обучения

По итогам освоения дополнительной общеобразовательной программы - дополнительной общеразвивающей программы «Основы программирования в Minecraft» обучающиеся должны будут овладеть следующими знаниями, умениями и навыками:

Знать:

- основы программирования в Minecraft
- основы работы в графической среде программирования MakeCode
- подходы искусственного интеллекта
- фундаментальные структуры языков программирования и языка JavaScript

Уметь:

- работать с числовыми и строковыми переменными, функциями, массивами
- программировать с использованием алгоритмов, циклов, событий и системы координат Minecraft
- программировать и создавать конструкции в Minecraft
- программировать робота, используя понятия искусственного интеллекта
- выявлять ошибки при создании программных продуктов
- писать простые программы на языке JavaScript
- применять полученные знания на практике, создавая игровые объекты и миры в среде Minecraft
- создавать свои игры в среде Minecraft, включая подготовку, сборку и тестирование
- применять творческий подход к решению задач
- работать в команде над созданием группового проекта
- создавать и презентовать свои проекты.

Владеть навыками в области:

- разработки игр в среде Minecraft
- блочного программирования в среде Minecraft
- планирования, алгоритмического и пространственного мышления
- тайм-менеджмента, постановки и решения задач и принятия решений
- поиска информации, анализа и структурирования проблем
- коммуникации и работы в команде над проектом
- подготовки презентаций проектов и защиты проектов/

2. УЧЕБНЫЙ ПЛАН ПРОГРАММЫ

В процессе преподавания курса «Основы программирования в Майнкрафте» используются как классические методы обучения (лекции), так и различные виды практической работы обучающихся по заданию преподавателя, которые направлены на развитие навыков логического и алгоритмического мышления, программирования и на поощрение интеллектуальных инициатив учащихся.

Формы организации образовательного процесса (индивидуальные, групповые и т.д.) и другие виды занятий по программе определяются содержанием программы. Образовательная деятельность обучающихся предусматривает следующие виды учебных занятий и учебных работ: лекции, практические и семинарские занятия, мастер-классы, тренинги, проектные занятия, консультации и другие виды учебных занятий и методической работы, определенные учебным планом.

№	Наименование дисциплины/раздела/темы	Количество академических часов				Форма аттестации /контроля
		Всего	Дистанционное обучение		СРС	
			теория	практич. занятия		
1.	Введение в программирование. Основные понятия.	1	0,5	0,5	0	Тестирование
2.	Практика создания алгоритмов	1	0,5	0,5	0	Тестирование
3.	Виды циклов	1	0,5	0,5	0	Тестирование
4.	Бесконечный цикл	1	0,5	0,5	0	Тестирование
5.	Изучение конструкции цикла в JavaScript.	1	0,5	0,5	0	Тестирование
6.	Старт работы с консолью JavaScript Часть 1	1	0,5	0,5	0	Тестирование
7.	Старт работы с консолью JavaScript Часть 2	1	0,5	0,5	0	Тестирование
8.	Старт работы с консолью JavaScript Часть 3	1	0,5	0,5	0	Тестирование
9.	Проектный урок	1	0	1	0	Тестирование
10.	События в программировании. 1 часть.	1	0,5	0,5	0	Тестирование

11.	События в программировании. 2 часть.	1	0,5	0,5	0	Тестирование
12.	Изучение понятия «условие» Часть 1	1	0,5	0,5	0	Тестирование
13.	Изучение понятия «условие» Часть 2	1	0,5	0,5	0	Тестирование
14.	Пуск команды при помощи Javascript.	1	0,5	0,5	0	Тестирование
15.	Изучение Breakpoint	1	0,5	0,5	0	Тестирование
16.	Координаты. Введение	1	0,5	0,5	0	Тестирование
17.	Оси координат	1	0,5	0,5	0	Тестирование
18.	Мировые и относительные координаты.	1	0,5	0,5	0	Тестирование
19.	Проектный урок	1	0	1	0	Тестирование
20.	Условные конструкции и их применение.	1	0,5	0,5	0	Тестирование
21.	Множественные условия.	1	0,5	0,5	0	Тестирование
22.	Логические операторы.	1	0,5	0,5	0	Тестирование
23.	Условия в Javascript.	1	0,5	0,5	0	Тестирование
24.	Понятие переменной.	1	0,5	0,5	0	Тестирование
25.	Типы данных.	1	0,5	0,5	0	Тестирование
26.	Функция. Часть 1	1	0,5	0,5	0	Тестирование

27	Функция. Часть 2	1	0,5	0,5	0	Тестирование
28	Аргументы функций.	1	0,5	0,5	0	Тестирование
29	Возвращаемое значение функции.	1	0,5	0,5	0	Тестирование
30	Проектный урок	1	0	1	0	Тестирование
31	Массивы	1	0,5	0,5	0	Тестирование
32	Индексы элементов массива	1	0,5	0,5	0	Тестирование
33	Добавление и удаление элементов массива	1	0,5	0,5	0	Тестирование
34	Двумерный массив	1	0,5	0,5	0	Тестирование
35	Работа с конструктором	1	0,5	0,5	0	Тестирование
36	Конкатенация строк. Связь текстовых констант и переменных	1	0,5	0,5	0	Тестирование
37	Сложение координат	1	0,5	0,5	0	Тестирование
38	Вложенные функции	1	0,5	0,5	0	Тестирование
39	Введение в искусственный интеллект	1	0,5	0,5	0	Тестирование
40	Защита проекта. Подготовка к демонстрации. Итоговая аттестация-презентация	1	0,5	0,5	0	Тестирование Защита проекта
ИТОГО		40	18,5	21,5	0	

3. СОДЕРЖАНИЕ ПРОГРАММЫ

1. Введение в программирование.
 - Понятие языка программирования
 - «Программирование» друг друга на решение бытовых задач
 - Изучение интерфейса code builder

Практическое задание: телепортация агента к разным объектам в мире Minecraft

2. Алгоритм.
 - Понятие алгоритма в повседневной жизни
 - Роль алгоритма в программировании

Практическое задание: разработка простых алгоритмов для агента

3. Движение агента.
 - Управление агентом
 - Изучение направлений движения агента
 - Постройка различных фигур по эскизу

Практическое задание: написание программы для перемещения агента по заданной траектории

4. Смена ячеек у агента.
 - Понятие ячейки
 - Размещение блоков
 - Выбор момента замены блоков у агента в программе

Практическое задание: построение объёмной радуги в Minecraft.

5. Старт работы с консолью JavaScript
 - Написание первой программы
 - Программирование команд управления агентом на JavaScript
 - Изучение направлений движения на JavaScript
 - Процесс смены блоков агента

Практическое задание: построение фигуры животного с помощью агента.

6. Циклы.
 - Понятие цикла
 - Виды циклов
 - Применение циклов в программах

Практическое задание: дождь из панд в Minecraft

7. Бесконечный цикл.
 - Применение бесконечного цикла в программах
 - Изучение блока «постоянно» в CodeBuilder

Практическое задание: программирование бесконечного танца для агента.

8. Изучение конструкции циклов в JavaScript.
 - Циклы с условием в JavaScript
 - Циклы с параметром в JavaScript
 - Бесконечный цикл в JavaScript

Практическое задание: построение моста через реку

9. Разработка программы для постройки своего дома в Minecraft.
 - Применение вложенных циклов на практике

- Понятие вложенности

Практическое задание: построение многоэтажного дома с окнами и крышей с помощью команд на JavaScript.

10. Итоговый проект 1 модуля.

- Повторение алгоритмов, циклов
- Закрепление практических навыков

Практическое задание: создание учениками самостоятельного проекта на основе ранее изученных тем

11. События в программировании. 1 часть.

- Понятие события в жизни
- Виды событий при разработке программ
- Обработчики событий в CodeBuilder

Практическое задание: программирование появления кроликов при взмахе морковки

12. События в программировании. 2 часть.

- Применение событий в создании игр
- Изучение команд настройки мира на JavaScript

Практическое задание: смена погоды и времени суток при различных условиях

13. Изучение понятия «Условие».

- Что такое условие
- Понятие выбора в играх и программировании

Практическое задание: написание игры «Охота на лисиц»

14. Пуск команды при помощи Javascript.

- Изучении команды `player.onDied`
- Применение блока `player.onItemInteracted`
- Вывод информации в чат пользователя

Практическое задание: написание программы, которая выводит на экран информацию о выпущенных стрелах игрока

15. Breakpoint.

- Точки останова в программировании
- Команда пауза и выход из цикла

Практическое задание: программирование диалога игрока

16. Координаты. Введение.

- Понятие координаты
- Изучение координат игрока в Minecraft

Практическое задание: телепортация игрока с помощью координат заданного объекта

17. Оси координат.

- Координатная плоскость
- Изучение направлений сторон света
- XYZ оси и их направление в мире Minecraft

Практическое задание: спавн животных на различном расстоянии от игрока

18. Мировые и относительные координаты.

- Запись различных видов координат в JavaScript

- Понятие абсолютной координаты в пространстве
- Понятие относительных координат игрока в Minecraft

Практическое задание: построение снеговика

19. Заполнение блоками выбранной области.

- Изучение функции `blocks.fill` и настройка её параметров
- Различие `pos` и `world` координат

Практическое задание: создание горы из динамита

20. Итоговый проект 2 модуля.

- Повторение событий, координат, условий
- Закрепление практических навыков

Практическое задание: создание учениками самостоятельного проекта на основе ранее изученных тем

21. Условные конструкции и их применение.

- Понятие условной конструкции
- Понятие ветвления

Практическое задание: программирование агента на поиск алмазов

22. Множественные условия.

- Изучение конструкции «если – то – иначе»
- Использование условного оператора

Практическое задание: создание почасового баффа

23. Логические операторы.

- Логическое «и»
- Логическое «или»
- Инверсия

Практическое задание: программа для выбора одежды и оружия игрока по погоде и времени суток

24. Условия в Javascript.

- Конструкция `if – then – else`
- Изучение команды `block.test`

Практическое задание: создание команды для поиска ресурсов

25. Переменная.

- Понятие переменной
- Объявление переменной в коде
- Нэйминг переменных
- Ввод значения переменной с клавиатуры

Практическое задание: создание заданного числа животных

26. Типы данных.

- Целочисленный тип `integer`
- Строковый тип `string`
- Логический тип `bool`
- Применение переменных в программах.

Практическое задание: написание программы, которая выводит на экран информацию об

ученике: имя, возраст, наличие домашнего животного

27. Функция. Начало.

- Определение функции
- Возвращаемое значение функции

Практическое задание: написание простого калькулятора

28. Аргументы функций.

- Освоение понятия параметра функций
- Различные типы параметров

Практическое задание: построение модели олимпийских колец

29. Функции в JavaScript.

- Применение функций на практике
- Блок функции в коде JavaScript
- Вызов функции

Практическое задание: создание собственного игрового режима

30. Итоговый проект 3 модуля.

- Повторение условных конструкций, переменных, функций
- Закрепление практических навыков

Практическое задание: создание учениками самостоятельного проекта на основе ранее изученных тем

31. Массивы.

- Массив как хранилище большого объема данных
- Создание массива
- Объявление массива в JavaScript

Практическое задание: создание сада из цветов

32. Индексы элементов массива.

- Индекс первого элемента
- Индекс последнего элемента
- Длина массива

Практическое задание: определение направления уничтожения агентом блока

33. Добавление и удаление элементов массива.

- Удаление значения по индексу
- Удаление первого элемента
- Добавление элемента в конец массива

Практическое задание: делаем из слитка броню для игрока

34. Двумерный массив.

- Понятие матрицы
- Проход двумерного массива циклом for

Практическое задание: создание зоопарка из животных

35. Работа с конструктором.

- Управление конструктором
- Размещение меток
- Ориентация конструктора

- Прокладывание пути по заданной траектории

Практическое задание: создание дороги между двумя объектами

36. Конкатенация строк. Связь текстовых констант и переменных.

- Понятие конкатенации
- Вывод на экран значений переменных

Практическое задание: определяем высоту горы

37. Сложение координат

- Относительные координаты
- Установка для переменной значения координат
- Тип данных Position

Практическое задание: создание светофора в Minecraft

38. Вложенные функции

- Вызов одной функции внутри другой
- Связь нескольких функций

Практическое задание: написание игры «Камень, ножницы, бумага»

39. Введение в искусственный интеллект.

- Понятие искусственного интеллекта
- Обучение искусственного интеллекта

Практическое задание: агент самостоятельно выбирается из пещеры

40. Итоговый проект курса.

- Повторение всех изученных ранее разделов
- Закрепление теоретических сведений
- Написание итогового теста

Практическое задание: постройка аквапарка

4. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Календарный учебный график формируется при осуществлении обучения в течение всего календарного года. По мере набора групп слушателей по программе составляется календарный график, учитывающий объемы лекций, практики, самоподготовки, другие формы организации занятий.

Темы / недели	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	Итого ч			
1. Введение в программирование. Основные понятия.	1																																										1	
2. Практика создания алгоритмов		1																																										1
3. Виды циклов			1																																									1
4. Бесконечный цикл				1																																								1
5. Изучение конструкции цикла в JavaScript.					1																																							1
6. Старт работы с консолью JavaScript Часть 1						1																																						1
7. Старт работы с консолью JavaScript Часть 2							1																																					1
8. Старт работы с консолью JavaScript Часть 3								1																																				1
9. Проектный									1																																			1

[illegible]

[illegible]

[illegible]

5.2. Критерии оценивания освоения программы при проведении различных форм контроля:

Тестирование (Приложение 1. Примерные вопросы для промежуточного тестирования).
Процент результативности (правильных ответов при выполнении тестовых заданий):

Выполнение теста	Итоговая оценка
65% и более правильных ответов	«Зачтено»
Менее 65% правильных ответов	«Не зачтено»

Проверка выполнения практических работ (Приложение 2. Примерные задания для проверки усвоения качества учебного материала). Система оценивания:

«Зачтено» – необходимый уровень выполнения задания достигнут, обучающийся демонстрирует хорошее знание теоретической и практической части материала занятия/блока занятий, достигнуты промежуточные и/или итоговые результаты работы над заданием.

«Не зачтено» - необходимый результат/уровень освоения не достигнут, обучающийся не усвоил теоретические основы и/или изученные практические приемы программирования и подходы искусственного интеллекта, не достиг промежуточных и итоговых результатов при выполнении задания.

Проверка результатов создания проекта на итоговой аттестации

Критерии оценки проекта в Minecraft	БАЛЛЫ
Обоснование выбора темы проекта, ее актуальность и своей роли в групповом проекте	0-2 балла
В проекте применено не менее двух фундаментальных структур языков программирования	0-3 балла
Применена система координат Minecraft.	0-2 балла
Применено не менее трех изученных методов и алгоритмов	0-3 балла
Выступающий может продемонстрировать не менее двух мини-игр в Minecraft, созданных на курсе	0-3 балла
Применено не менее четырех изученных алгоритмов и конструкций на языке JavaScript в Minecraft	0-3 балла
Показана самостоятельность работы над своей частью проекта	0-3 балла
Компетентность докладчика (ответы на вопросы)	0-3 балла
Итоговая оценка: «Не зачтено» «Зачтено»	0-11 баллов 12-22 баллов

6. УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ (УЧЕБНИКИ, РАЗДАТОЧНЫЕ МАТЕРИАЛЫ, МЕТОДИЧЕСКИЕ УКАЗАНИЯ, ПЛАКАТЫ, СЛАЙДЫ, ИНТЕРНЕТ-РЕСУРСЫ)

6.1. Информационные и учебно-методические условия реализации программы

Учебно-методический комплект

Для реализации целей и задач обучения по общеобразовательной общеразвивающей программе «Основы программирования в Minecraft» используется комплект материалов преподавателя, который включает:

1. Текстовое методическое пособие с описанием целей, результатов каждого занятия, теоретического материала и практических работ.
2. Видеоурок для преподавателя с методическими указаниями и порядком объяснения учебного материала.
3. Раздаточный материал для учащихся - описание дополнительной самостоятельной работы учащихся по каждому занятию с примерами и рекомендациями по выполнению.
4. Описание мероприятий по контролю знаний – тестовые вопросы, практические задания.
5. Рекомендации по проведению итоговой аттестации и защиты проектов.
6. Дополнительные материалы – презентации по тематике занятий, материалы по работе с дополнительными источниками.
7. Дополнительные материалы – инструкции по установке необходимого программного обеспечения, описание технических требований к компьютерному оборудованию.

Материалы преподавателя размещаются на учебном портале преподавателей, размещенном на сервере информационно-телекоммуникационной сети «Интернет», и доступны по ссылке для всех преподавателей курса. Материалы обучающихся раздаются в печатном виде или рассылаются преподавателем индивидуально каждому обучающемуся.

6.2. Рекомендованная литература для обучающихся

Основная:

1. Гарланд, Я. Изучаем программирование на примере Minecraft / Я. Гарланд. - Бомбара, 2022 г. - 160 с.
2. Крейг, Р. Програмируем с Minecraft / Ричардсон Крейг. – М.: МИФ, 2017 – 369 с.
3. Битно, Л. IT-тренажер для детей / Л.Г.Битно. – Ростов-на-Дону: Феникс, 2023. – 30 с.
4. Морган, Н. JavaScript для детей. Самоучитель по программированию / Ник Морган; пер. с англ. Станислава Ломакина; [науч. ред. Д. Абрамова]. — М. : Манн, Иванов и Фербер, 2016. — 288 с.

Дополнительная:

6. Миллер, М. Все секреты Minecraft. Строительство/ Меган Миллер. – М.: Эксмо, 2020 – 128 с. : ил.
7. Миллер, М. Все секреты Minecraft. Красный камень/ Меган Миллер. – М.:

Эксмо, 2014 – 128 с.

8. Michael Milford, The Complete Guide to Artificial Intelligence for Kids, 2020 – 50 с
9. Dale Lane, Machine Learning for Kids: A Project-Based Introduction to Artificial Intelligence, 2021 – 392 с.

6.3.Перечень ресурсов информационно-телекоммуникационной сети «Интернет», рекомендованных для освоения программы

- <https://www.minecraft.net/ru-ru> - портал Microsoft по Minecraft
- <https://education.minecraft.net/ru-ru/resources/math-subject-kit> - образовательный портал Microsoft по Minesraft
- <https://habr.com/ru/post/673268/> - образовательная статья о генерации миров в Minecraft
- <https://minecraft.makecode.com/javascript> - краткое введение в JavaScript на портале Microsoft
- http://window.edu.ru/catalog/resources?p_rubr=2.1.6 - единое окно доступа к образовательным ресурсам, раздел "Информатика и ИКТ"

7. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

7.1. Материально-техническая и ресурсная база

Для реализации программы предполагается использование учебных аудитории для проведения занятий лекционного типа, занятий семинарского типа, выполнения проектных работ, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации:

- Рабочее место, обеспеченное доступом в Интернет, ноутбуком и иными средствами визуализации учебного материала, микрофоном.
- Электронный информационно-образовательный портал, размещенный на сервере в информационно-телекоммуникационной сети «Интернет».
- Специальное программное обеспечение для веб-разработки, необходимое для реализации образовательных задач курса.
- Стандартное программное обеспечение для работы над разработкой учебно-методических материалов.
- Мастерские и аудитории для проведения открытых занятий.

Специальных помещений, предполагающих наличие какого-либо специального оборудования для реализации данной программы, не предусматривается.

7.2. Кадровое обеспечение программы

Образовательный процесс по программе осуществляется педагогом дополнительного образования с профильным высшим или средним профессиональным образованием.

К занятию педагогической деятельностью по дополнительной общеобразовательной программе также допускаются лица, обучающиеся по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направленности дополнительных общеобразовательных программ, и успешно прошедшие промежуточную аттестацию не менее чем за два года обучения.

Реализация дополнительной общеобразовательной (общеразвивающей) программы обеспечивается руководящими и педагогическими работниками организации, а также лицами, привлекаемыми к реализации программы на условиях гражданско-правового договора.

У педагогического работника, реализующего дополнительную общеобразовательную программу, должны быть сформированы основные компетенции, необходимые для обеспечения успешного достижения обучающимися планируемых результатов освоения программы, в том числе умения:

- обеспечивать условия для успешной деятельности, позитивной мотивации, а также самомотивирования обучающихся;
- осуществлять самостоятельный поиск и анализ информации с помощью современных информационно-поисковых технологий;
- разрабатывать программы учебных предметов, выбирать учебники и учебно-методическую литературу, рекомендовать обучающимся дополнительные источники информации, в том числе Интернет-ресурсы;
- реализовывать педагогическое оценивание деятельности обучающихся;
- работать с текстовыми редакторами, электронными таблицами, электронной почтой и браузерами, мультимедийным оборудованием.

Приложение 1. Примерные вопросы для
промежуточного тестирования

1. Какое направление нам нужно использовать, чтобы при ходьбе блоки размещались под ногами игрока?
 - X (отрицательный)
 - Y (отрицательный)
 - Y (положительный)
 - Z (положительный)

2. Какая получится длина полоски, если в команде заполнения указать такие значения: из 5 в -1?
 - 7 блоков
 - 6 блоков
 - 4 блока

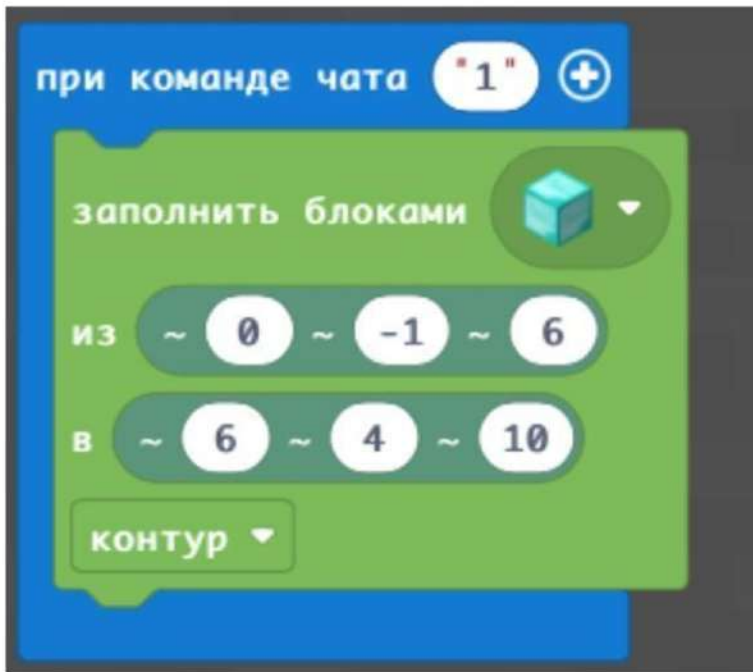
3. Какие обработчики ЕСТЬ в MakeCode? (выбери один или несколько вариантов)
 - При условии, что агент стоит
 - При телепортации игрока
 - При условии создания моба
 - При условии, что блок сломан

4. Абсолютные координаты - это ...
 - Координаты относительно абсолютного начала мира
 - Координаты агента относительно игрока
 - Координаты одного объекта относительно другого

5. Что нужно указать в открывающемся меню команды «заполнить блоками», чтобы создать пустую коробку?
 - заменить
 - оставить
 - контур

6. Относительные координаты – это...
 - Координаты относительно агента
 - Координаты игрока относительно точки абсолютного начала
 - Координаты одного объекта относительно другого

7. Какая ширина, высота и длина коробки получится, если запустить эту программу?



- ширина 6, высота 5, длина 4
- ширина 7, высота 5, длина 5
- **ширина 7, высота 6, длина 4**

•

8. Координата Y отвечает за ...:

- ширину и длину
- ширину
- длину
- **высоту**

9. Что НЕ является примером цикла (выберите один или несколько вариантов)?

- **выбор продуктов в магазине**
- смена времен года
- дни недели
- **разовое посещение дополнительного занятия**

10. Выберите наиболее точное определение. «Алгоритм - это...?»

- Несколько действий, которые начинаются одновременно
- Повторяющиеся команды
- Набор команд
- **Последовательность действий, приводящая к заданному результату**

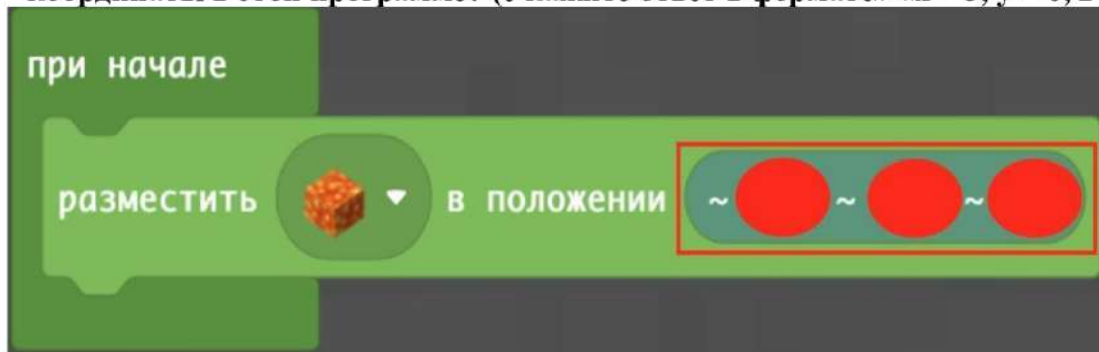
11. Ваня написал программу, но забыл, что же она делает. Внимательно посмотри на программу и отметь правильный ответ:

12.



- агент движется вперед 16 блоков
- **агент движется по квадрату 4x4**
- агент постоянно поворачивается на месте
- агент движется по квадрату 3x3

13. Чтобы блок разместился под ногами игрока, как должны быть отмечены координаты в этой программе? (Укажите ответ в формате» «x = 5, y = 0, z = 0»)

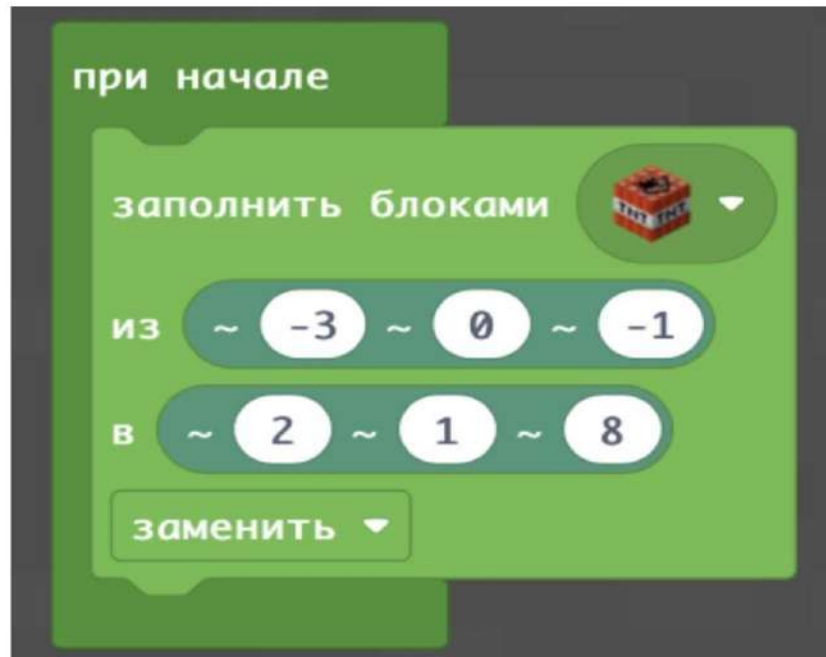


Ответ: _____ (x = 0, y = -1, z = 0)

14. Какое название переменной является корректным? (выберите один или несколько вариантов):

- 1Bb
- **mobs12**
- *arrows
- **Abcd**
- /zombie

15. Укажи правильные параметры платформы, которую создает эта программа:
(например, ширина 10, высота 10, длина 10)



Ответ: _____ (ширина 6, высота 2, длина 10)

16. Что используется для хранения каких-либо данных в программах?

- переменная
- условный оператор
- алгоритм
- цикл «пока»

17. В каком случае агент переместится вверх?



- если блок будет справа
- если блок будет слева
- только если блоки будут и слева и справа

18. Что делает условный оператор в программе?

- повторяет действие несколько раз
- **проверяет истинность заданного условия**
- повторяет действие, пока выполняется условие
- хранит данные

19. Оператор «иначе» в условной конструкции срабатывает ...

- всегда
- только когда условия выше выполняется
- **только когда условия выше НЕ выполняются**

20. Фрагмент кода, к которому можно обратиться из любого места программы называется ... ?

- массив
- **функция**
- переменная
- алгоритм

21. Порядковый номер ячеек в массиве называется ...?

- **индексом**
- параметром
- элементом

22. Выберите правильный вариант цикла for на 5 повторений на языке JavaScript:

- for let i = 0; i < 5; i++
- **for (let i = 0; i < 5; i++) { }**
- for 5
- for (let i = 0; i < 5; i++)

23. Выбери строку для правильного создания функции на языке JavaScript:

- `function () { }`
- `let stena ()`
- `function stena = 0`
- **`function stena () { }`**

24. В каких случаях желательно использовать массив (выбери один или несколько вариантов)?

- чтобы запомнить, какой блок перед игроком
- **чтобы запомнить несколько точек на карте**
- чтобы объединить несколько команд для выполнения какой-либо задачи
- **чтобы сохранить все виды блоков, которые игрок встречал на пути**

25. Чтобы активировать функцию в программе вы используете команду...

- установить
- активировать
- **вызвать**

26. Что может быть примером массива? (выберите один или несколько вариантов)

- полка с книгами
- смена времен года
- **фотоальбом**
- коллекция монет

27. Для чего используются параметры функции?

- **для передачи значений в функцию**
- для хранения какой-либо информации внутри функции
- для вызова функции

Приложение 2. Примерные задания для оценки качества освоения учебного материала

1. Написать программу строительства железной дороги.

Написать программу, в которой конструктор проложит железную дорогу. Размеры и сложность дороги можно выбрать самостоятельно.

2. Напишите программу решения математического примера.

Написать программу для решения арифметического примера, например $5+5=?$.

Протестировать работу программы на разных числах. Проиллюстрировать работу программы с помощью постройки блоков.

3. Создать свою мини-игру

Создать свой вариант мини-игры в Майнкрафте, обязательно прописав условие победы и поражения. Варианты игры - сражение, выполнение какого-либо задания по таймеру (например, найти нужное количество ресурсов, или пройти полосу препятствий за отведенное время и т. д.).

4. Напишите программу постройки сооружения.

Напишите программу строительства арены по следующей картинке. При создании программы используйте функции.



5. Написать программу с использованием условной конструкции.

Придумать вопрос и написать программу для проверки истинности ответа. При решении нужно использовать конструкцию из следующего примера (чтобы получилась программа по типу игры «ГОРЯЧО-ХОЛОДНО»):

```
let year = 0

player.say('В каком году официально была опубликована
игра Майнкрафт?')

if (year < 2009) {

    player.say("Это слишком рано...")

} else if (year > 2009) {

    player.say("Это поздновато")

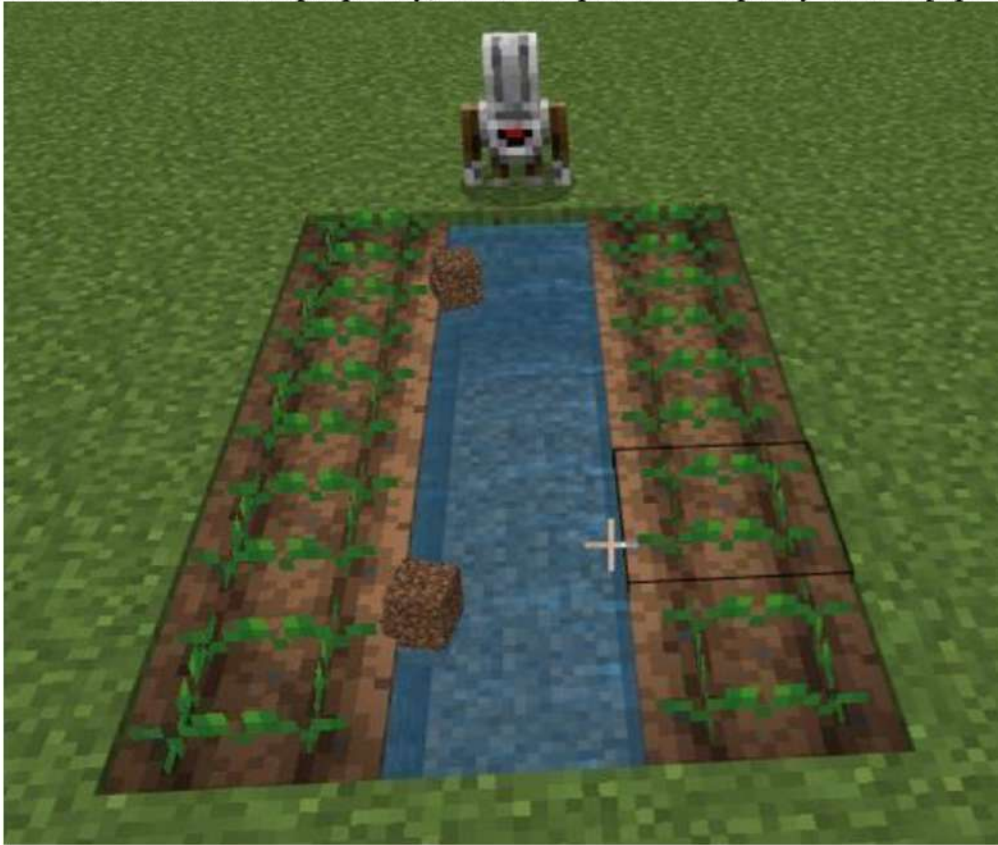
} else {

    player.say("Верно")

}
```

6. Написать программу для агента с использованием циклов.

Написать для агента программу, чтобы он разместил грядку такого формата:



7. Создать и запрограммировать блоки для игры.

Разместить 4 блока (как кнопки) разных цветов. Запрограммировать каждый блок на какое-либо событие. Например: один блок запрограммируем так: при условии, что он будет сломан, будет давать игроку какие-либо ценные предметы. А другой блок, запрограммировать на какое-либо вредоносное для игрока событие.

8. Напишите программу создания постройки.

Написать программу создания бункера из двух слоев. Материалы для слоев выбрать самостоятельно, например: бедрок (или коренная порода) наружный слой, обсидиан внутренний слой. Внутри воздух. Параметры бункера: ширина 10, высота 6, длина 10 блоков.

9. Написать программу с использованием учета времени.

Напишите программу, которая позволит отражать в чате, какое количество времени игрок находится в игре. При создании программы используйте следующий алгоритм 1. В первую очередь нам понадобится блок «при начале» из вкладки ЦИКЛЫ 2. Далее создайте переменную, например, «секунды» и вставьте команду «установить» в переменную значение 0» в «при начале» 3. После этого используйте цикл «пока истина» 4. В цикле поместите команду «изменить секунды на 1» 5. Пауза 1000 мс 6. Отразите содержимое переменной в чате.