

**Индивидуальный предприниматель Селендеева О.Н.**

**УТВЕРЖДАЮ**

**Индивидуальный предприниматель**

\_\_\_\_\_/Селендеева О.Н.//

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА  
ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА  
«Minecraft: введение в искусственный интеллект»**

**Москва, 2023**

## 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
.....	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 «Minecraft: введение в искусственный интеллект».

В ходе дополнительной общеобразовательной общеразвивающей программы «Minecraft: введение в искусственный интеллект» обучающиеся научатся программировать в Minecraft, создавать конструкции в мире Minecraft, 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 Minecraft 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 ИТ, осваивает азы программирования через игру Minecraft 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 84 академических часов, 32 недель (8 месяцев).

**Режим занятий:** 2-4 академических часа в неделю

Продолжительность академического часа – 45 минут.

Занятия начинаются не ранее 9.00 часов утра и заканчиваются не позднее 20.00 часов. Для обучающихся в возрасте 16-18 лет допускается окончание занятий в 21.00 часов.

Продолжительность занятий в учебные дни - не более 3-х академических часов в день, в выходные и каникулярные дни - не более 4 академических часов в день. После 30-45 минут теоретических занятий организуется перерыв длительностью не менее 10 мин.

**Адресат программы и примерный портрет слушателя курсов:** программа разработана для учащихся от 6,5 до 12 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 ИТ-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 «Minecraft: введение в искусственный интеллект» 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 «Minecraft: введение в искусственный интеллект» 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, проведение открытых занятий, консультации и другие виды учебных занятий и методической работы, определенные учебным планом.

№	Наименование дисциплины/раздела/темы	Количество академических часов			СРС	Форма аттестации /контроля
		Всего	в т.ч. аудиторных			
			теория	практич. занятия		
1	Знакомство с Minecraft. Алгоритм	2	1	1	0	Практическое задание
2	Практика создания алгоритмов. Циклы. Часть 1	3	1	1	1	Практическое задание
3	Виды циклов	4	1,5	1,5	1	Практическое задание
4	Проектный урок	3	0,5	1	1,5	Практическое задание, Контрольные вопросы
5	Циклы с условием. Вложенные циклы	2	0,5	0,5	1	Практическое задание
6	События в Minecraft	2	1	1	0	Практическое задание
7	Система координат в Minecraft	2	1	1	0	Практическое задание
8	Проектный урок	3	1	1	1	Практическое задание, Контрольные вопросы
9	Система координат. Часть 2	3	1	1	1	Практическое задание
10	Система координат. Часть 3	2	0,5	0,5	1	Практическое задание
11	Конструктор	2	0,5	0,5	1	Практическое задание
12	Проектный урок	4	1	1	2	Практическое задание, Контрольные вопросы.

13	Переменная	2	0,5	0,5	1	Практическое задание
14	Переменные. Условные операторы	2	1	1	0	Практическое задание
15	Условный оператор. Часть 1	2	1	1	0	Практическое задание
16	Проектный урок	3	1	1	1	Практическое задание, Контрольные вопросы
17	Условный оператор. Часть 2	3	1	1	1	Практическое задание
18	Условные конструкции. Часть 1	2	0,5	0,5	1	Практическое задание
19	Условные конструкции. Часть 2	2	0,5	0,5	1	Практическое задание
20	Проектный урок	4	1	1	2	Практическое задание, Контрольные вопросы.
21	Функции. Часть 1	2	0,5	0,5	1	Практическое задание
22	Функции. Часть 2	2	1	1	0	Практическое задание
23	Массив	2	1	1	0	Практическое задание
24	Проектный урок	4	1	1	2	Практическое задание, Контрольные вопросы
25	Параметры функции	3	1	1	1	Практическое задание
26	Введение в синтаксис языка JavaScript	2	0,5	0,5	1	Практическое задание
27	Циклы и условия на языке JavaScript	2	0,5	0,5	1	Практическое задание
28	Проектный урок	4	1	1	2	Практическое задание, Контрольные вопросы
29	Массивы и функции на языке JavaScript	3	1	1	1	Практическое задание
30	Условные конструкции на языке JavaScript	2	0,5	0,5	1	Практическое задание
31	Завершение проекта	2	0,5	0,5	1	Практическое задание
32	Защита проекта. Подготовка к демонстрации. Итоговая аттестация-презентация	4	1	1	2	Практическое задание, Контрольные вопросы. Защита проекта
	<b>ИТОГО</b>	<b>84</b>	<b>26,5</b>	<b>27</b>	<b>30,5</b>	

### 3. 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	Итого часов
1. Знакомство с Minecraft. Алгоритм	2																																2
2. Практика создания алгоритмов. Циклы. Часть 1		3																															3
3. Виды циклов			4																														4
4. Проектный урок				2,5																													2,5
Промежуточный контроль				0,5																													0,5
5. Циклы с условием. Вложенные циклы					2																												2
6. События в Minecraft						2																											2
7. Система координат в Minecraft							2																										2
8. Проектный урок								2,5																									2,5
Промежуточный контроль								0,5																									0,5
9. Система координат. Часть 2									3																								3
10. Система координат. Часть 3										2																							2
11. Конструктор											2																						2
12. Проектный урок												3																					3
Промежуточный контроль												1																					1
13. Переменная													2																				2
14. Переменные. Условные операторы														2																			2





## 4. РАБОЧАЯ ПРОГРАММА

### 1. Знакомство с Minecraft. Алгоритм

- Знакомство со средой Minecraft.
- Основные понятия в программировании.
- Создание простых команд.
- Выполнение заданий на создание алгоритмов.

**Практическое задание:** написание программ для работы агента с использованием изученных команд и алгоритмов.

### 2. Практика создания алгоритмов. Циклы

- Повторение темы алгоритмы и практика написания команд.
- Знакомство с понятием циклов.
- Написание программ на постройку различных объектов агентом.

**Практическое задание:** создание программ для строительства объектов агентом с использованием циклов.

### 3. Виды циклов

- Повторение понятия циклы.
- Виды циклов.
- Циклы с условием.
- Создание программ с применением различных видов циклов.

**Практическое задание:** написать не менее 5 программ для агента с использованием циклов.

### 4. Проектный урок

- Повторение изученных тем по основам программирования.
- Правила проектной работы, распределение ресурсов.
- Выбор идеи и работа над проектом.
- Демонстрация и обсуждение созданных на уроке проектов.

**Практическое задание:** выполнить тест на повторение теоретического материала, выбор проекта, строительство игрового объекта.

### 5. Циклы с условием. Вложенные циклы

- Циклы с условием - повторение, закрепление на практике.
- Вложенные циклы в программировании.
- Создание программ для агента с использованием изученных видов циклов.

**Практическое задание:** написание программ по добыче ресурсов и строительству дома агентом с применением циклов с условием и вложенных циклов.

### 6. События в Minecraft

- Понятие «событие» и «обработчик событий».
- Вкладки MakeCode для работы с событиями.
- Написание программ с использованием событий.
- Создание механизма из нескольких событий.

**Практическое задание:** создание 10 программ с использованием различных событий.

## 7. Система координат в Minecraft

- Координаты в мире Minecraft.
- Команда «заполнить блоками».
- Создание команд с использованием абсолютных и относительных координат.

**Практическое задание:** написание команд с использованием системы координат и команды «заполнить блоками».

## 8. Проектный урок.

- Повторение изученных тем по основам программирования.
- Правила проектной работы, распределение ресурсов.
- Работа над проектом.
- Демонстрация и обсуждение созданных на уроке проектов.

**Практическое задание:** выполнить тест на повторение теоретического материала, создание проекта «Деревня» в режиме выживания.

## 9. Система координат. Часть 2

- Вкладка «Фигуры», построение фигур.
- Понятие радиуса.
- Практика создания программ с использованием координат.
- Применение команды «выбрать случайное положение» в программах.

**Практическое задание:** написание программ - строительство аквариума, катка, заполнение воздухом, построение фигур.

## 10. Система координат. Часть 3

- Использование координат для клонирования объектов.
- Циклическое клонирование. Программа заполнения блоками.
- Переменные в коде программ.
- Улучшение программы клонирования при помощи переменных.

**Практическое задание:** создание программы с циклическим клонированием объекта.

## 11. Конструктор

- Знакомство с конструктором и изучение его работы.
- Создание программ по работе с конструктором.
- Практика программирования - создание игровых объектов при помощи конструктора и клонирования.

**Практическое задание:** выполнение заданий на работу с конструктором, написание программ для создания игровых объектов - острова, мосты, фонари, воздушные шары.

## 12. Проектный урок

- Повторение изученных тем по основам программирования.
- Правила проектной работы, распределение ресурсов.
- Создание проекта.
- Демонстрация и обсуждение созданных на уроке проектов.

**Практическое задание:** выполнить тест на повторение теоретического материала, создать проект «Парящая деревня».

### 13. Переменная

- Переменные в программировании.
- Как создать переменную и задать ей значение.
- Практика создания программ с использованием переменных.

**Практическое задание:** написание программ - строительство пирамиды, «сила прыжка», «калькулятор», «пещера летучих мышей».

### 14. Переменные. Часть 2

- Повторение темы «переменные».
- Алгоритмы с применением цикла «пока» и условного оператора.
- Практика программирования - создание программ с использованием переменных и условного оператора.

**Практическое задание:** создание программ - «агент-дровосек», «секундомер», «счетчик стрел».

### 15. Условный оператор. Часть 1

- Изучение понятия «условный оператор».
- Условные конструкции в программах.
- Оператор «иначе».

**Практическое задание:** создание мини-игры с условием победы и проигрыша, создание программы на проверку примеров.

### 16. Проектный урок

- Повторение изученных тем по основам программирования.
- Правила проектной работы, распределение ресурсов.
- Начало создания группового проекта.
- Демонстрация и обсуждение созданных на уроке работ.

**Практическое задание:** выполнить тест на повторение теоретического материала, выбор идеи группового проекта и своей роли в нем, работа над своей частью проекта.

### 17. Условный оператор. Часть 2

- Задача проекта «Дом-квестов».
- Практика программирования - программируем комнаты квеста.
- Тестирование проекта, обсуждение вопросов и результатов.

**Практическое задание:** программирование 5 комнат проекта «Дом квестов».

### 18. Условные конструкции. Часть 1

- Подготовка к решению задачи о проведении переписи населения.
- Написание программы проверки введенного значения.
- Написание программы для вывода полученных данных в чате.
- Практика программирования - создание программы анализа данных о переписи населения в подготовленной деревне.

**Практическое задание:** создание программы проверки введенного значения, проведения переписи населения и анализа результатов.

## 19. Условные конструкции. Часть 2

- Задача создания программы по строительству лабиринта.
- Применение функций в программах.
- Создание программы «танец агента».

**Практическое задание:** создание программы по строительству лабиринта, программы «танец агента».

## 20. Проектный урок

- Повторение изученных тем по основам программирования.
- Правила проектной работы, распределение ресурсов.
- Продолжение создания группового проекта.
- Демонстрация и обсуждение созданных на уроке работ.

**Практическое задание:** выполнить тест на повторение теоретического материала, работа над своей частью проекта.

## 21. Функции. Часть 1

- Функции в программировании. Параметры функций.
- Практика программирования - программа «Бургер».
- Практика программирования - функции с параметрами.
- Программа «строительство стен».

**Практическое задание:** написание программы по созданию Бургера и строительству стен.

## 22. Функции. Часть 2

- Подготовка к созданию игры.
- Создание функции «Арена».
- Написание функций для игровых настроек.
- Создание функции для размещения мобов.
- Создание программы с условием победы.

**Практическое задание:** подготовка функций для создания игры, сборка и тестирование игры.

## 23. Массив

- Понятия «массив», «элемент массива» и «индекс массива».
- Создание массива и получение значений из массива.
- Практика программирования - пишем программы с применением массивов.

**Практическое задание:** создание программ «Пояс-телепорт», «Радужный маяк», «Зоопарк».

## 24. Проектный урок

- Повторение изученных тем по основам программирования.
- Продолжение создания группового проекта.
- Демонстрация и обсуждение созданных на уроке работ.

**Практическое задание:** выполнить тест на повторение теоретического материала, работа над своей частью проекта.

## 25. Параметры функции

- Повторение понятий «массив», «функция».
- Обсуждение: разбор кода программы об истории мобов в Майнкрафт и предложения по улучшению.
- Строим Колизей - обсуждаем код программы.
- Практика программирования - пишем программы с применением массивов и функций.

**Практическое задание:** создание программ «История мобов», «Ходячая статуя», «Колизей».

## 26. Введение в синтаксис языка JavaScript

- Синтаксис команд языка JavaScript.
- Использование обработчиков событий на языке JavaScript.
- Практика программирования - создание команд для агента на языке JavaScript.
- Переменные и типы данных в программах на языке JavaScript.

**Практическое задание:** написание команд на JavaScript, создание программы «Агент проходит коридор», «Полоса из блоков».

## 27. Циклы и условия на языке JavaScript

- Цикл for и его использование в программах.
- Цикл while и его применение в программах.
- Оператор if и его применение в программировании.
- Практика программирования: решение задач на применение циклов и оператора if.

**Практическое задание:** создание программ на строительство башни, вывод чисел, «Таймер», «Секундомер».

## 28. Проектный урок

- Повторение изученных тем по основам программирования.
- Продолжение создания группового проекта.
- Демонстрация и обсуждение созданных на уроке работ.

**Практическое задание:** выполнить тест на повторение теоретического материала, работа над своей частью проекта.

## 29. Массивы и функции на языке JavaScript

- Объявление массива и работа с элементами массива в JavaScript.
- Объявление и вызов функции в JavaScript.
- Локальные и глобальные переменные. Параметры функции.
- Практика программирования - решаем задания на применение массивов и функций.

**Практическое задание:** написание строк кода с применением функций, массивов. Создание программ «Строим стену», «Радуга».

### **30. Условные конструкции на языке JavaScript**

- Повторение условных конструкций в программировании.
- Конструкции else, else if в JavaScript.
- Практика программирования - пишем программы с применением условных конструкций.

**Практическое задание:** создание программ «Проверка ответа на вопрос», «Ваша биография», программы на работу со случайными числами.

### **31. Завершение проекта**

- Завершение создания группового проекта.
- Практика программирования: написание программ на языке JavaScript по изученным темам.
- Демонстрация и обсуждение созданных на уроке работ.

**Практическое задание:** написание программ «Строим столб», «Дождь из куриц», «Разноцветный пол» на JavaScript с применением изученных конструкций программирования, работа над своей частью группового проекта.

### **32. Защита проекта**

- Выполнение итогового теста по изученному на курсе материалу.
- Подготовка к выступлению и защита проектов курса.
- Обсуждение результатов курса, рекомендации по дальнейшему обучению и развитию в сфере программирования.

**Практическое задание:** участие в защите проектов.

## **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 «Minecraft: введение в искусственный интеллект» 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.

## 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
<b>Итоговая оценка:</b> «Не зачтено» «Зачтено»	<b>0-11 баллов</b> <b>12-22 баллов</b>

## **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 «Minecraft: введение в искусственный интеллект» 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](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:

- Компьютерные классы, обеспечивающие доступ в Интернет, и оснащенные мультимедиа проектором или иными средствами визуализации учебного материала, магнитной доской или флипчартом.
- 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.

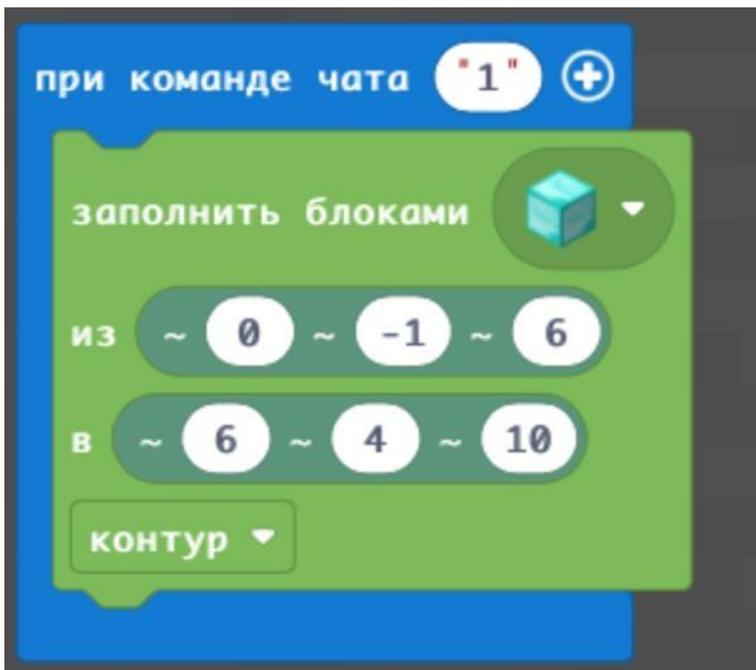
Реализация дополнительной общеобразовательной (общеразвивающей) программы обеспечивается руководящими и педагогическими работниками организации, а также лицами, привлекаемыми к реализации программы на условиях гражданско-правового договора.

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.

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)?

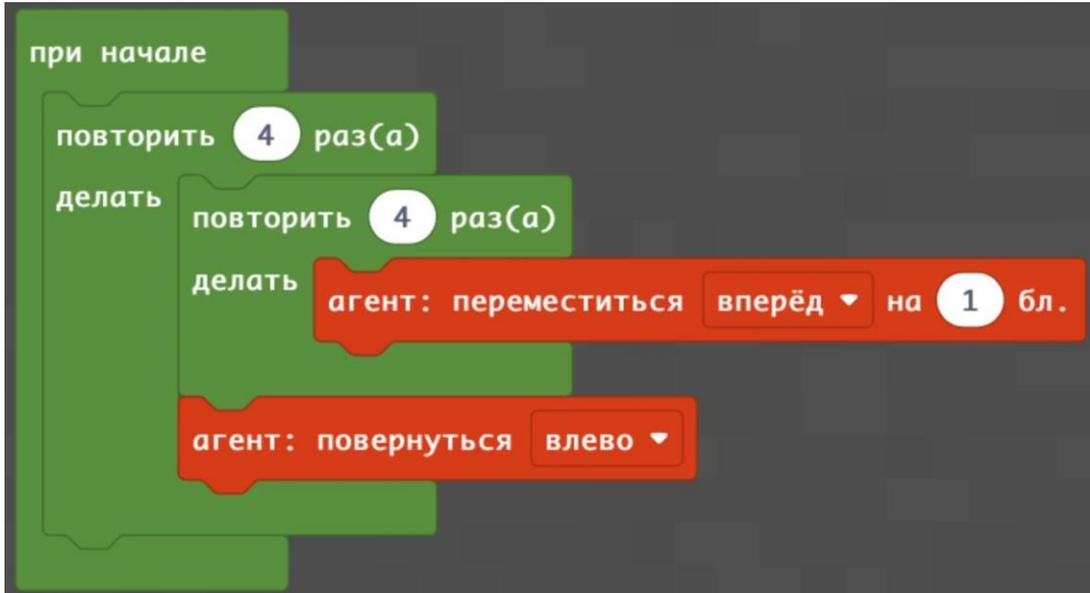
- **выбор продуктов в магазине**
- смена времен года
- дни недели
- **разовое посещение дополнительного занятия**

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

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

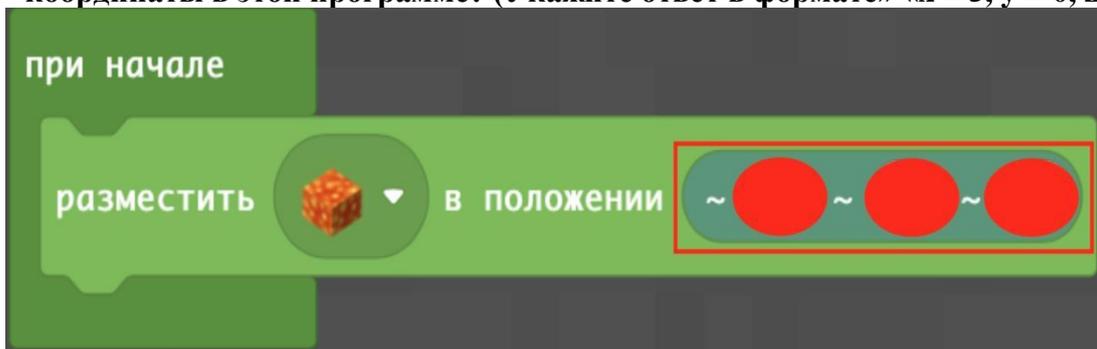
11. Ваня написал программу, но забыл, что же она делает. Внимательно посмотри на программу и отметь правильный ответ:

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»)

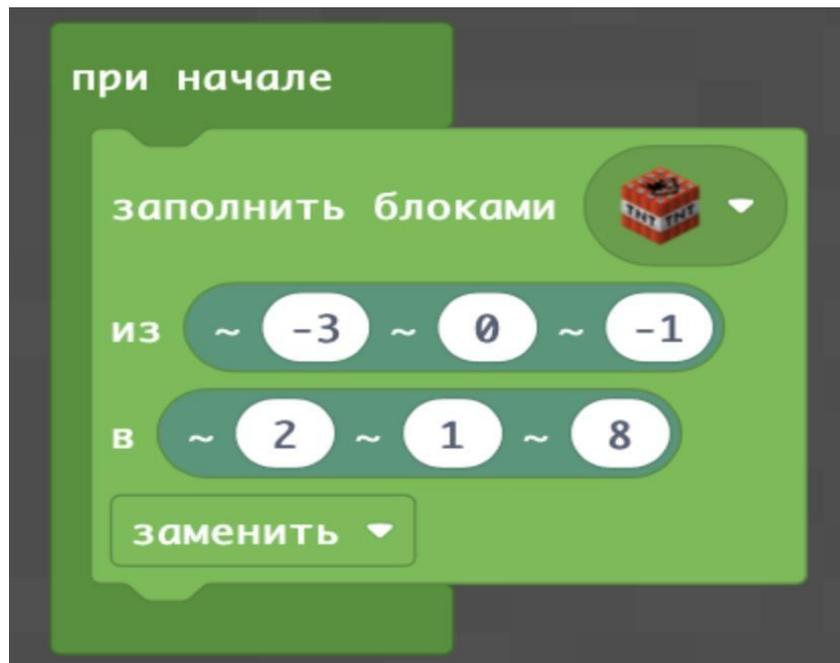


Ответ: \_\_\_\_\_ (x = 0, y = -1, z = 0)

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. Что делает эта программа?

```
1 while (agent.detect(AgentDetection.Block, DOWN)) {  
2     agent.move(FORWARD, 1)  
3 }
```

- перемещает агента на 1 блок вперёд
- **перемещает агента вперед, пока он чувствует блок внизу**
- перемещает агента на 1 блок вверх
- постоянно перемещает агента вперёд

23. 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++)

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

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

25. 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**

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

- установить
- активировать
- **вызвать**

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

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

28. Для чего используются параметры функции?

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

29. При каком условии агент пойдет вперед в этой программе?

```
1 if (agent.detect(AgentDetection.Block, DOWN)) {
2     agent.move(FORWARD, 1)
3 } else {
4     agent.turn(TurnDirection.Left)
5 }
```

- если обнаружит блок внизу
- если условие не выполнится
- если обнаружит блок вверх
- агент в любом случае переместится вперед

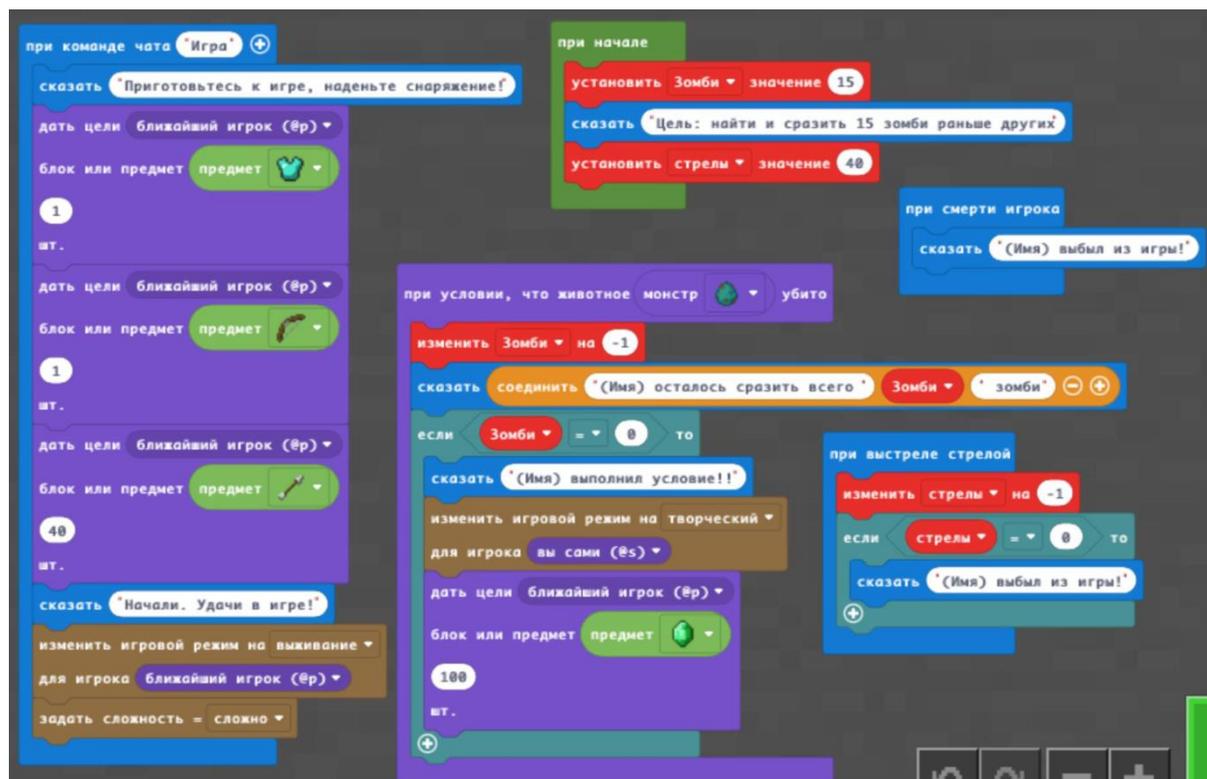
## Приложение 2. Примерные задания для оценки качества освоения учебного материала

### 1. Написать программу строительства железной дороги.

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

### 2. Реализовать игру по предложенному алгоритму.

Реализуйте игру по алгоритму, предложенному на картинке. Затем нужно протестировать ход игры, задавая различные параметры персонажей, игрового оружия и предметов.



### 3. Напишите программу решения математического примера.

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

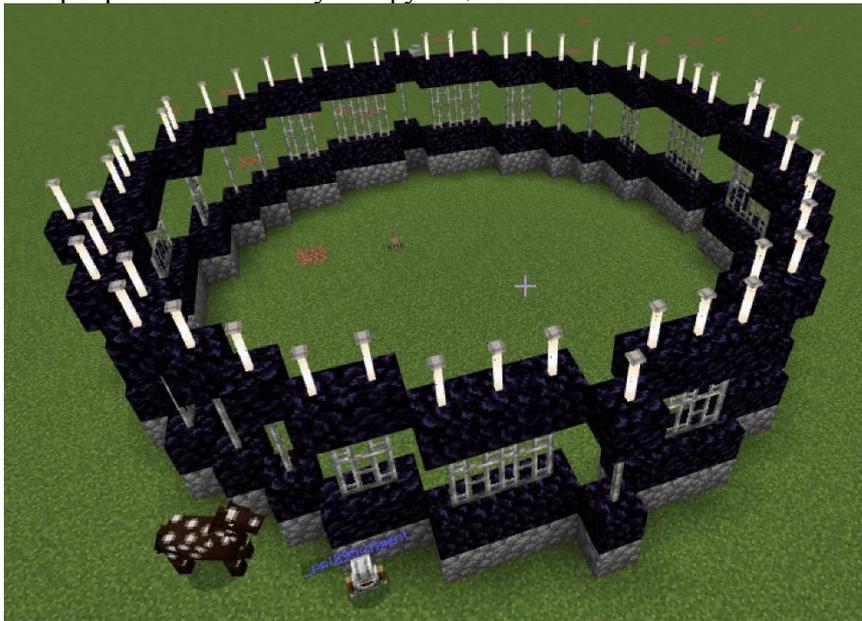
Протестировать работу программы на разных числах. Проиллюстрировать работу программы с помощью постройки блоков.

### 4. 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.).

### 5. Напишите программу постройки сооружения.

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



### 6. 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»):

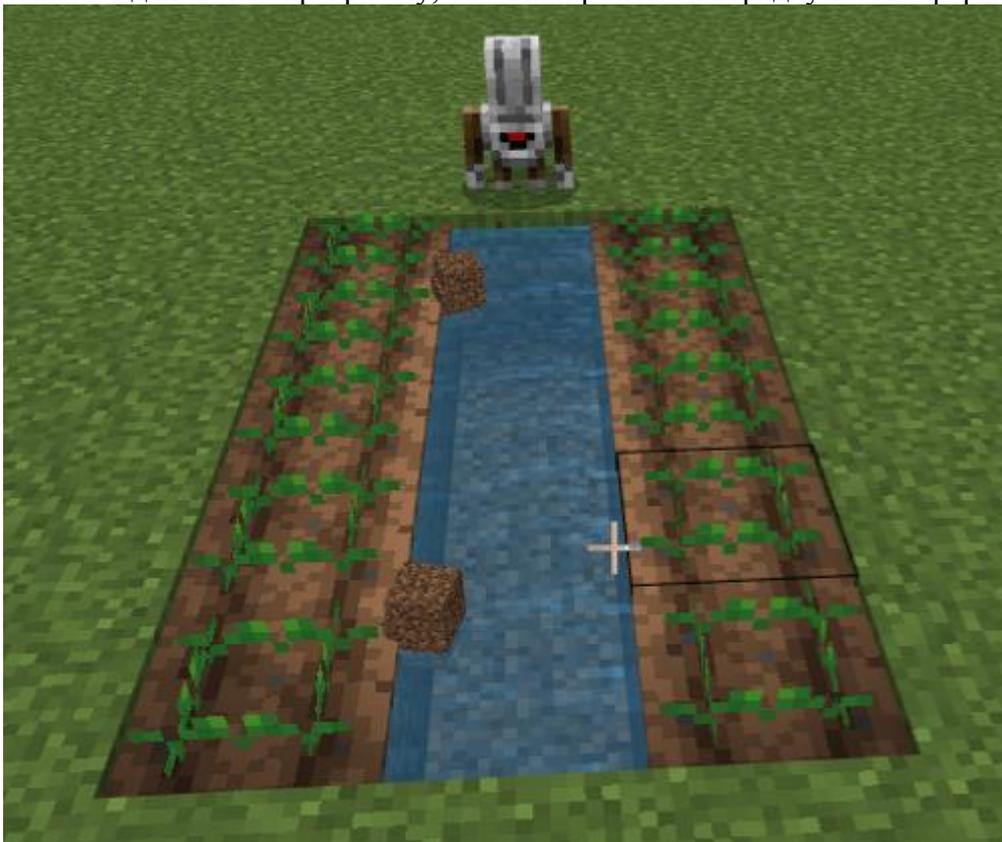
```
let year = 0

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 поздноато")
} else {
    player.say("Верно")
}
```

**7. Написать программу для агента с использованием циклов.**

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



**8. Создать и запрограммировать блоки для игры.**

Разместить 4 блока (как кнопки) разных цветов. Запрограммировать каждый блок на какое-либо событие. Например: один блок программируем 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.

**9. 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.

**10. 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.