

Разработка клеточного автомата – Муравей Лэнгтона на C#

Ульянов Егор Андреевич

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье рассматривается и описывается разработка клеточного автомата – Муравей Лэнгтона. Автомат будет разрабатываться на языке программирования C# с помощью игрового движка Unity 3D. Практическим результатом является разработанный клеточный автомат.

Ключевые слова: Муравей Лэнгтона, C#, Unity 3D

Developing a Cellular Automaton - Langton's Ant on C

Ulianov Egor Andreevich

Sholom-Aleichem Priamursky State University

Student

Abstract

This article discusses and describes the development of a cellular automaton - Langton's Ant. The machine will be developed in the C # programming language using the Unity 3D game engine. The practical result is the developed cellular automaton.

Keywords: Langton's Ant, C#, Unity 3D

Клеточный автомат-это дискретная модель вычислений, изучаемая в теории автоматов. Клеточные автоматы также называются клеточными пространствами, автоматами тесселяции, однородными структурами, клеточными структурами, структурами тесселяции и итерационными массивами. Клеточные автоматы нашли применение в различных областях, включая физику, теоретическую биологию и моделирование микроструктуры.

Клеточный автомат состоит из регулярной сетки ячеек, каждая из которых находится в одном из конечного числа состояний, таких как ВКЛ и ВЫКЛ. Сетка может быть любой размерности. Для каждой ячейки определяется набор ячеек, называемый ее окрестностью, относительно указанной ячейки. Начальное состояние выбирается путем назначения состояния для каждой ячейки. Новое поколение создается согласно некоторому фиксированному правилу, определяющему новое состояние каждой клетки в терминах текущего состояния клетки и состояний клеток в ее окрестности. Обычно, правило обновления состояния ячеек одинаково для каждой ячейки и не изменяется во времени, а применяется ко всей сетке

одновременно, хотя известны исключения, такие как стохастический клеточный автомат и асинхронный клеточный автомат.

Эта концепция была первоначально открыта в 1940-х годах Станиславом Уламом и Джоном фон Нейманом. Хотя некоторые из них изучались в 1950-х и 1960-х годах, только в 1970-х годах и в "игре жизни" Конвея, двумерном клеточном автомате, интерес к этому предмету расширился за пределы академических кругов. В 1980-х годах Стивен Вольфрам занялся систематическим изучением одномерных клеточных автоматов. Вольфрам опубликовал новый вид науки в 2002 году, утверждая, что клеточные автоматы имеют применение во многих областях науки. К ним относятся компьютерные процессоры и криптография.

Основные классификации клеточных автоматов, описанные вольфрамом, пронумерованы от одного до четырех. Это, по порядку, автоматы, в которых паттерны обычно стабилизируются в однородность, автоматы, в которых паттерны развиваются в основном стабильные или колеблющиеся структуры, автоматы, в которых паттерны развиваются, казалось бы, хаотично, и автоматы, в которых паттерны становятся чрезвычайно сложными и могут длиться долгое время со стабильными локальными структурами. Этот последний класс считается вычислительно универсальным или способным моделировать машину Тьюринга. Особые типы клеточных автоматов являются обратимыми, когда только одна конфигурация приводит непосредственно к следующей, и тотальными, когда будущее значение отдельных клеток зависит только от общего значения группы соседних клеток. Клеточные автоматы могут моделировать различные реальные системы, в том числе биологические и химические.

Муравей Лэнгтона - это двумерная универсальная машина Тьюринга с очень простым набором правил, но сложным эмерджентным поведением. Он был изобретен Крисом Лэнгтоном в 1986 году и основан на квадратной решетке черных и белых клеток. Универсальность муравья Лэнгтона была доказана в 2000 году. Идея была обобщена несколькими различными способами, которые добавляют больше цветов и больше существ.

Правила клеточного автомата

Рассмотрим бесконечную плоскость, разбитую на клетки, покрашенные некоторым образом в чёрный и белый цвет. Пусть в одной из клеток находится «муравей», который на каждом шаге может двигаться в одном из четырёх направлений в клетку, соседнюю по стороне. Муравей движется согласно следующим правилам:

На чёрном квадрате — повернуть на 90° влево, изменить цвет квадрата на белый, сделать шаг вперед на следующую клетку.

На белом квадрате — повернуть на 90° вправо, изменить цвет квадрата на чёрный, сделать шаг вперед на следующую клетку.

Эти простые правила вызывают довольно сложное поведение: после некоторого периода довольно случайного движения муравей, видимо, начинает непременно строить дорогу из 104 шагов, повторяющуюся бесконечно, независимо от изначальной раскраски поля. Это наводит на

мысль, что «магистральное» поведение является стабильным аттрактором муравья Лэнгтона.

Муравей Лэнгтона также может быть описан как клеточный автомат, в котором почти всё поле покрашено в чёрно-белый цвет, а клетка с «муравьём» имеет один из восьми различных цветов, кодирующих соответственно все возможные комбинации чёрного/белого цвета клетки и направления движения муравья.

Цель данной статьи рассмотреть возможности игрового движка Unity 3D в разработке клеточных автоматов, а конкретнее муравья Лэнгтона.

И.А. Савин, О.В. Батенькина рассмотрели процесс написания скриптовых сценариев при разработке виртуального тренажера[1]. В своей работе Э.Р. Гараева, И.И. Бикмуллина, И.А. Барков описали возможности Unity 3D на предмет создания 3D-моделей[2]. С.А. Суродин в своей статье представил сценарий углубленного изучения одного из лучших движков, существующих на данный момент, для создания красивых 2D и 3D игр[3]. В своей работе Р.Ф. Гайнуллин, В.А. Захаров, Е.А. Аксенова изучили инструмент для разработки двух- и трёхмерных игр – Unity 3D[4]. А.Ю. Субботина, Н. И. Хохлов в результате их работы была предложена трехмерная версия игры Конвея «Жизнь»[5].

Создаем 3Dпроект и называем его. Далее создаем структуру папок, а именно папку моделей с дочерней папкой материалы, папку для скриптов см. рисунок 1.

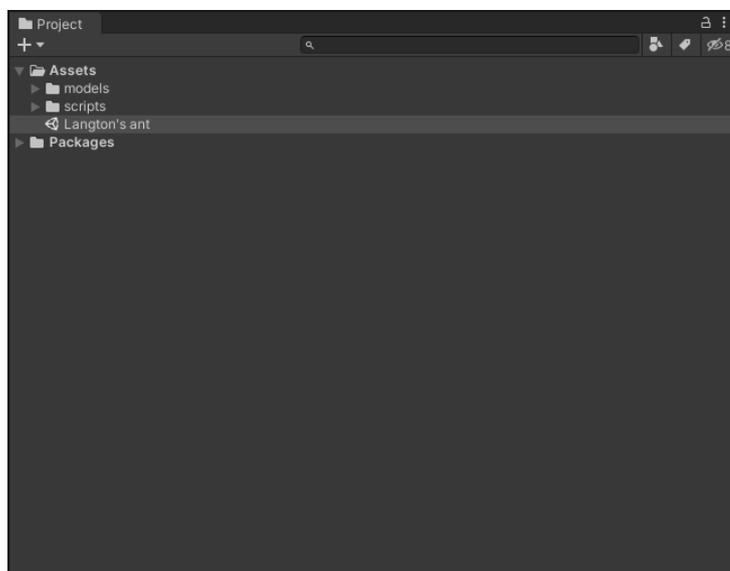


Рис. 1. Создание структуры папок

Далее понадобится три скрипта, называемы `xant_movement` (для движений муравья), `change_plane` (меняет цвет квадрата), `map_generator` (Генерирует карту). Также понадобится 3префаба. Создаем два пустых объекта, один называем `ant`, другой `map_generator`. Третьим префабом, будет 3Добъект «plane» с названием `white_plane` и размером 0.1, 01, 01. Переносим все в папку моделей. Там же создаем материал с шейдером `Unlit/Color`

белым цветом, материал с помощью draganddropпереносим на префаб, тем самым окрашивая его в белый. Копируем материал, меняем цвет на красный. Клонировуем «белый квадрат»окрашивая его см. рисунок 2.

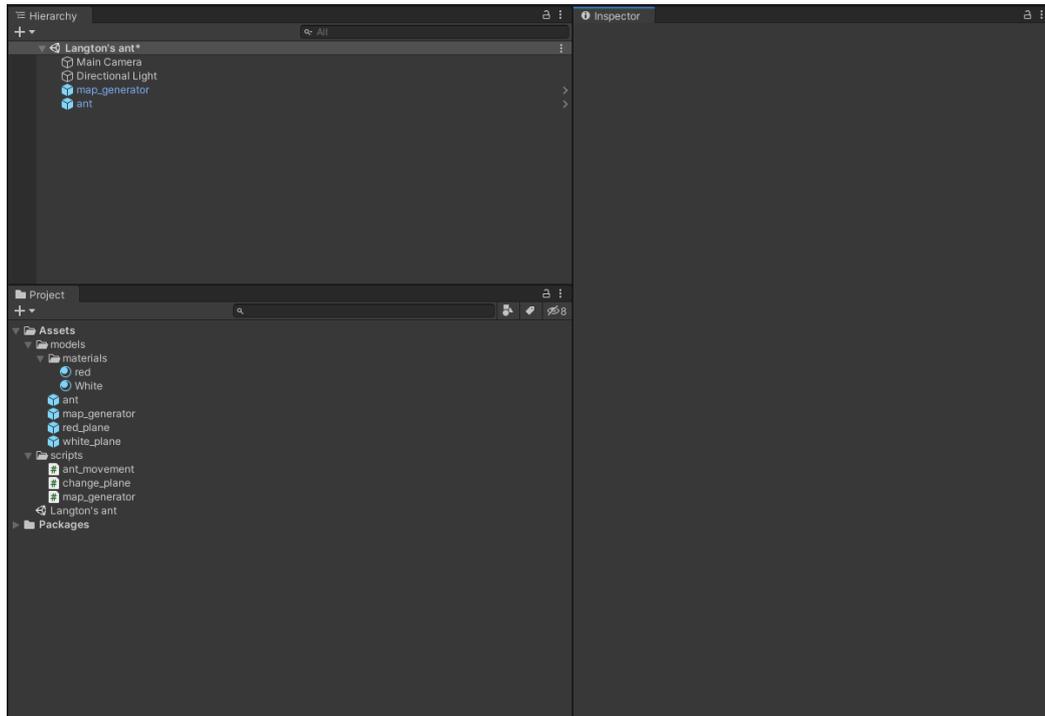


Рис. 2. Создание скриптов и префабов

Для того чтобы видеть генерируемую карту, настроим камеру см. рисунок 3.

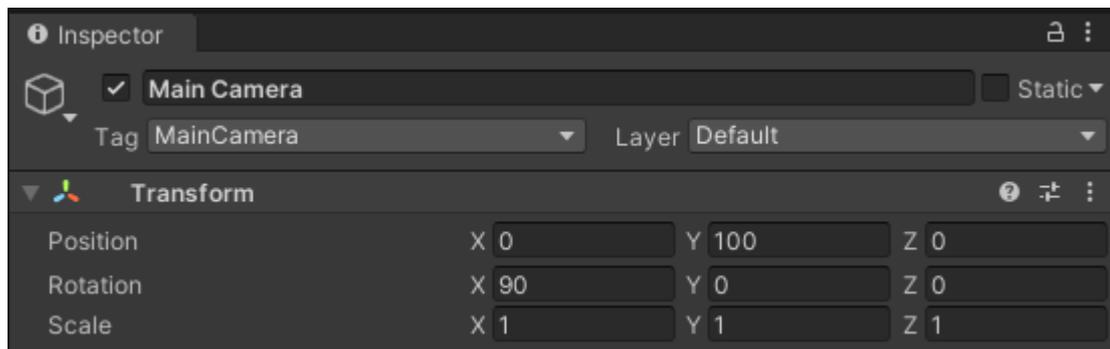


Рис. 3. Настройка камеры

Далее необходимо для каждого квадрата убрать компонент MeshCollider и добавить BoxCollider с размерами 1, 1, 1. Для префаба муравья надо добавить тот же BoxCollider, но с размерами 0.1, 0.1, 0.1, и с галочкой isTrigger. Также муравью необходим компонент Rigidbody см. рисунок 4 и 5 соответственно.

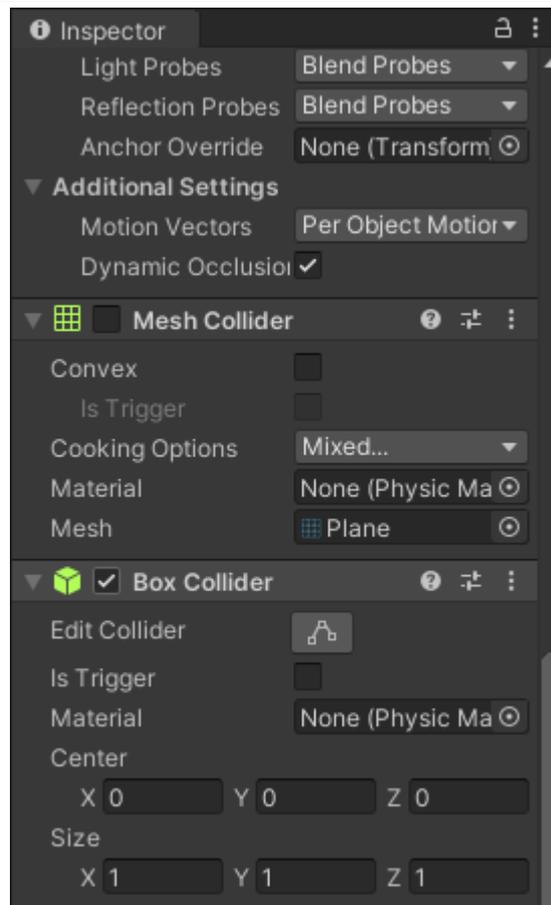


Рис. 4. Настройка квадратов

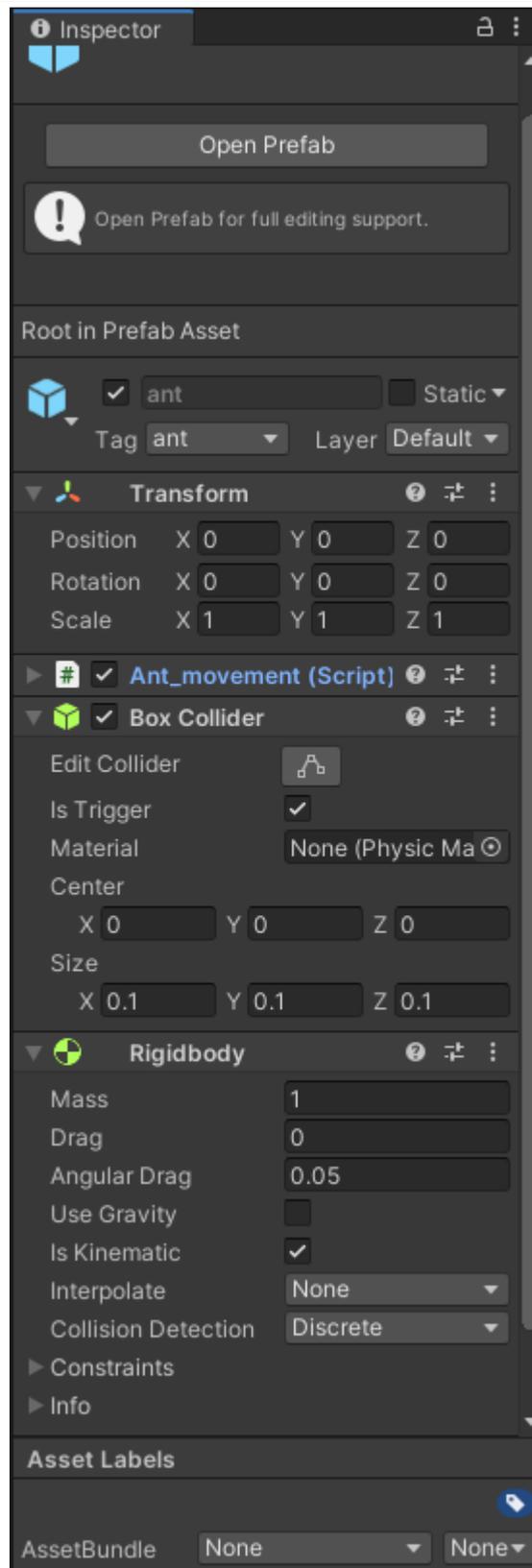


Рис.5. Настройка муравья

Переносим скрипты на соответствующие префабы; `change_plane` на оба квадрата, `ant_movement` на `ant`, а `map_generator` на одноименный. Далее необходимо создать три новых тега, и раздать в соответствии с правилами префабам (`white_plane-left`, `red_plane-right`) см. рисунок 6.

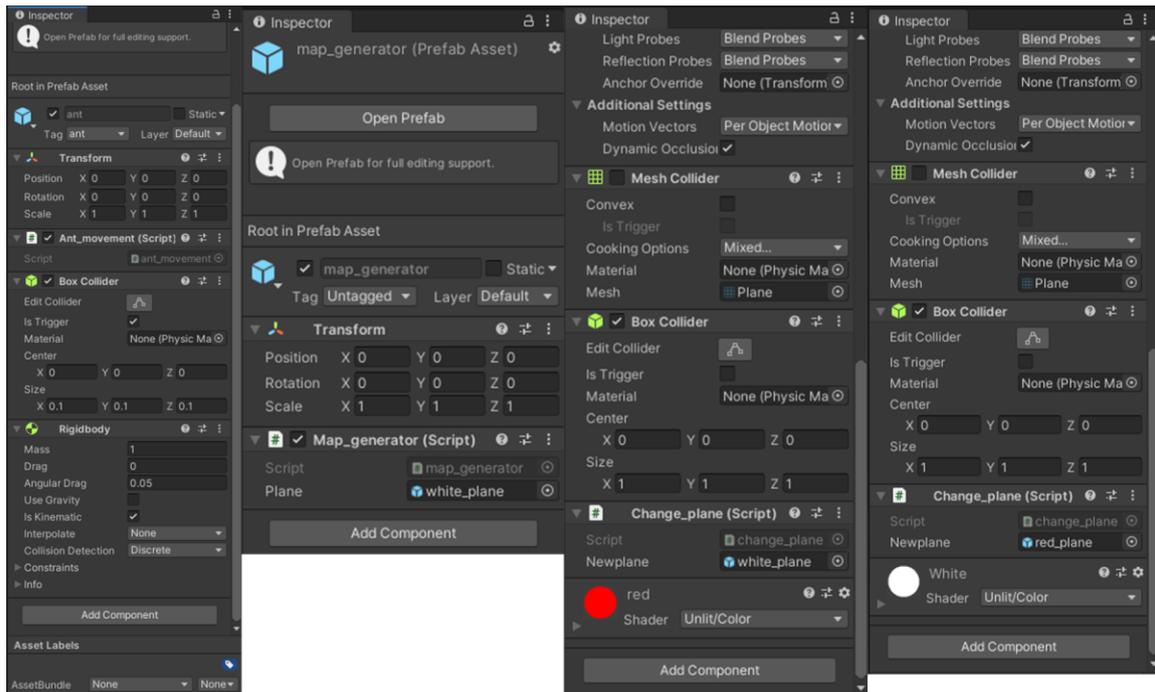


Рис. 6. Добавление скриптов и тегов

Далее приступаем к скриптам, и начнем с генератора мира. Для создания белого поля, в настройках скрипта в префабе ставим white_plane см. рисунок 7 - 9.

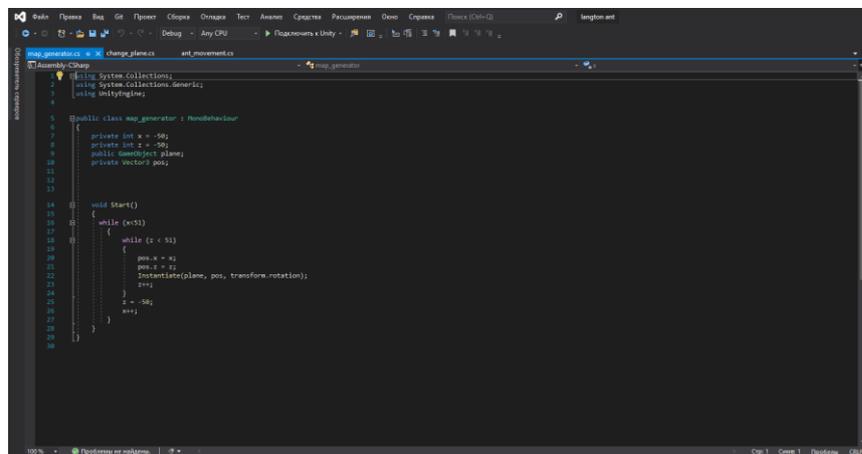


Рис. 7. Генератор мира



Рис. 8. Настройка скрипта



Рис. 9. Результат генератора

Далее напишем скрипт движения муравья см. рисунок 10.

```
Assembly-CSharp
ant_movement
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 [Script Unity | Ссылка: 0]
6 public class ant_movement : MonoBehaviour
7 {
8     private Vector3 rot;
9     [Сообщение Unity | Ссылка: 0]
10    void OnTriggerEnter(Collider plane)
11    {
12        rot = transform.eulerAngles;
13        if (plane.tag == "left")
14        {
15            rot.y -= 90;
16        }
17        if (plane.tag == "right")
18        {
19            rot.y += 90;
20        }
21    }
22
23    [Сообщение Unity | Ссылка: 0]
24    void FixedUpdate()
25    {
26        transform.eulerAngles = rot;
27        transform.Translate(Vector3.forward);
28    }
29 }
```

Рис. 10. Скрипт движения муравья

Дописываем код изменений цвета квадратов см. рисунок 11.

```
Assembly-CSharp change_plane
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class change_plane : MonoBehaviour
6  {
7
8      public GameObject newplane;
9
10     void OnTriggerEnter(Collider ant) {
11         if (ant.tag == "ant")
12         {
13             Destroy(gameObject);
14             Instantiate(newplane, transform.position, transform.rotation);
15         }
16     }
17
18 }
19
20 }
```

Рис. 10. Скрипт изменений цвета квадратов

Далее проверяем автомат см. рисунок 11.

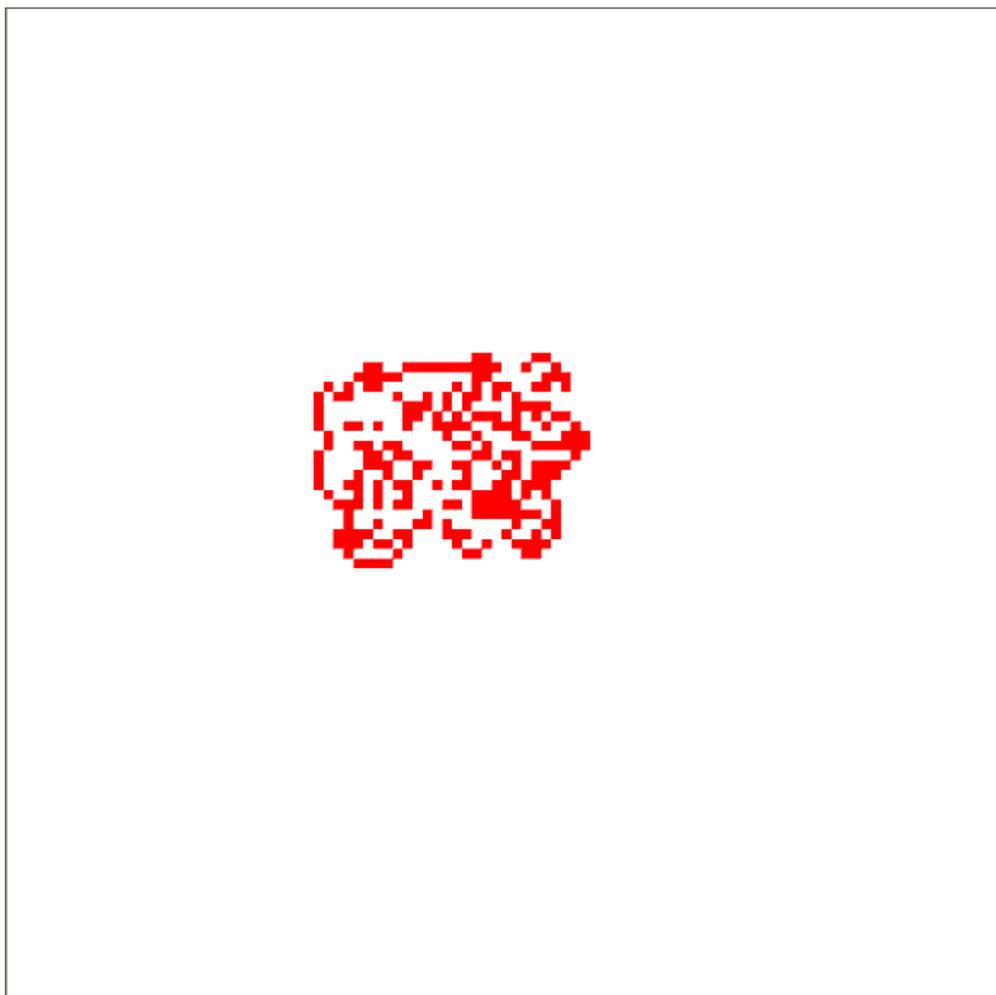


Рис. 11. Готовый клеточный автомат муравей Лэнгтона

Были проанализированы существующие аналоги и методы разработки, а также выбрана среда разработки. Для реализации поставленной задачи отлично подошла разработка с помощью Visual Studio и языка программирования C#. Такой выбор заметно упростил разработку проекта, так как в интернете имеется достаточное кол-во документации. Во время создания игры был полученный ценный опыт работы с этим средством разработки.

В итоге был разработан клеточный автомат муравей Лэнгтона. Созданный автомат имеет потенциал к развитию, а именно: добавление новых функций; увеличение карты.

Библиографический список

1. Савин И.А., Батенькина О.В. Написание скриптов для трехмерного графического движка // Визуальная культура: дизайн, реклама, информационные технологии. 2018. № 12-7 (28). С. 7-15.
2. Долженко А.И., Глушенко С.А. Особенности подготовки 3d-объектов, смоделированных в Blender, для импорта в Unity 3D // Прикаспийский журнал: управление и высокие технологии. 2014. №6. С. 92-96.
3. Суродин С.А. Unity 3D. разработка сценария проектирования в среде Unity 3D// Информатика и вычислительная техника. 2015. №3. С. 504-511.
4. Гайнуллин Р.Ф., Захаров В.А., Аксенова Е.А. Создание 2d игры на Unity 3D 5.4 // Вестник современных исследований. 2018. №4. С. 78-82.
5. Субботина А.Ю., Хохлов Н. И. Реализация клеточных автоматов "игра жизнь" и клеточного автомата Кохомото-ооно с применением технологии MPI // Компьютерные исследования и моделирование. 2010. №17. С. 319-322.