# Реализация приложения

## Основные сведения о программе

Для исследования алгоритма волновой трассировки была разработана программа с использованием следующих фреймворков: XNA Framework и MonoGame, в которой можно реализовать произвольный планарный граф с помощью 2D Текстуры.

Результатом работы программы является визуальное представление кратчайшего пути.

На рисунке 8 показана начальное окно программы. Генерируется сетка для последующей генерации лабиринта.

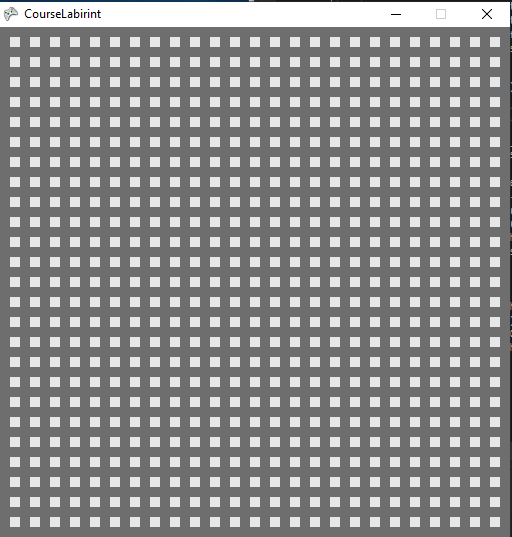


Рисунок 24 - Начальное состояние программы

Далее с помощью алгоритма поиска в глубину начинается генерация лабиринта. В конечном результате получается идеальный лабиринт без петель и изолированных областей. Для большего количества маршрутов выполняется удаление которых непроходимых клеток. В результате работы получается конечная матрица узлов, изображенная на рисунке 25. Данный метод генерации позволяет избежать наличия недостижимых областей в графе.

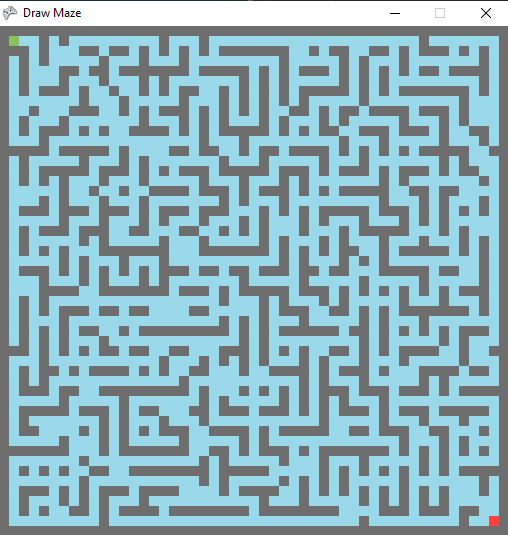


Рисунок 25 - Конечная матрица узлов

После нажатия клавиши “Пробел” происходит обход всей матрицы, с пометкой посещенных узлов и расстояния до стартового узла. Это необходимо для последующего построения феромонных троп. Один из этапов прохода матрицы показан на рисунке 26.

Далее запускается алгоритм муравьиной колонии, задаются начальные параметры метода, запускается движение муравьев. После некоторого количества итераций находится кратчайший найденный путь от старта до финиша и алгоритм завершает свою работу. Если необходимо оптимизировать маршрут или начать поиск другого пути, нажимается кнопка “F2”.

Изображение конечного результата показано на рисунке 27.

При реализации данного приложения был использован язык программирования C# и среда разработки Visual Studio. Этапы выполнения программы показаны на рисунке 28.

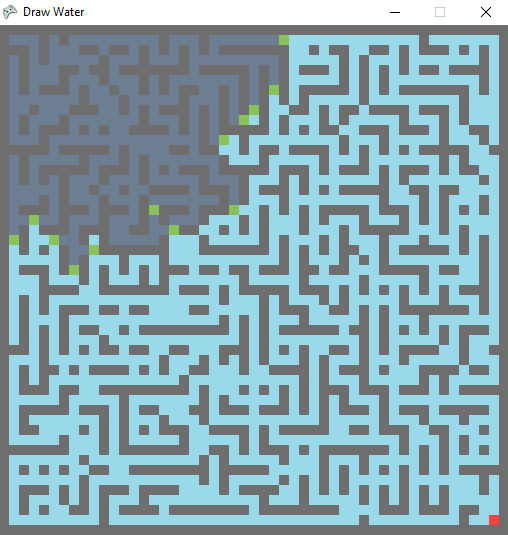


Рисунок 26 - Распространение волны

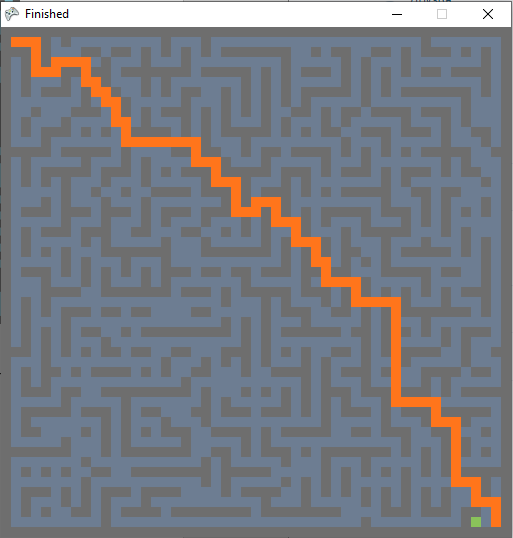


Рисунок 27 - Конечный результат

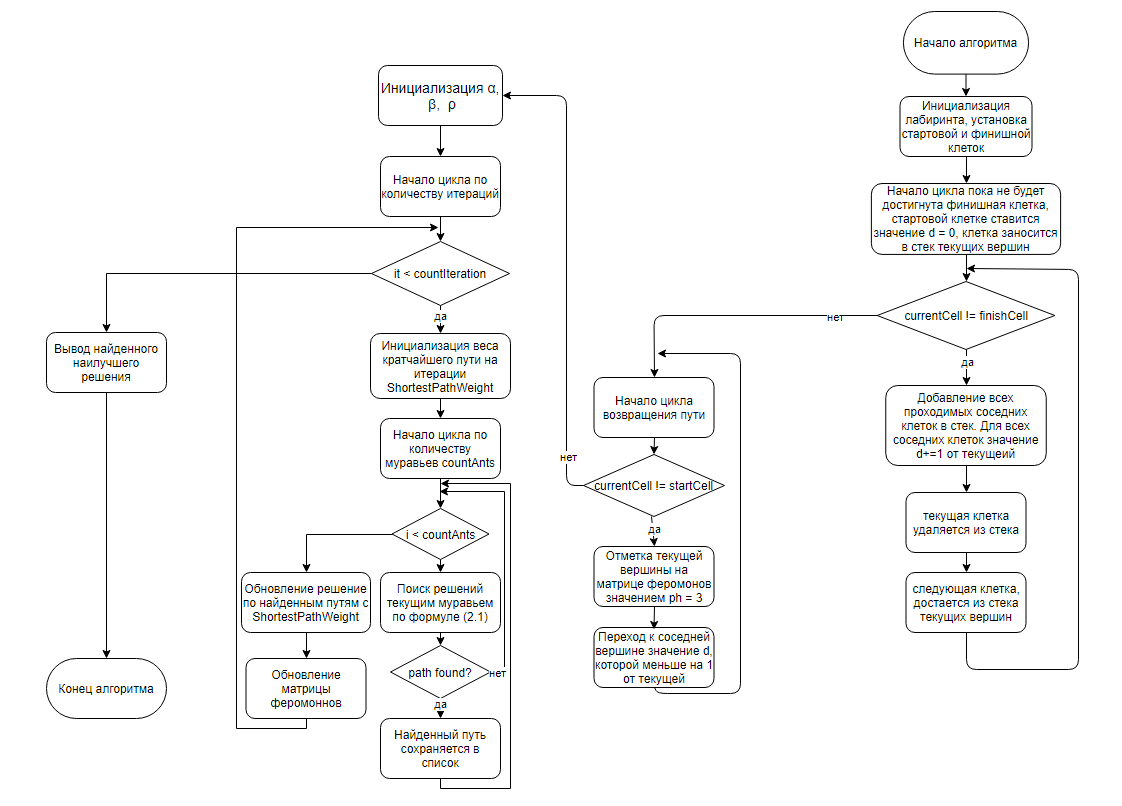


Рисунок 28 - Блок схема комбинированного алгоритма

## Особенности реализации

Программа включает следующие основные модули(файлы):

а) Program.cs – основной файл, отвечающий за инициализацию игрового окна;

б) Game1.cs – файл игрового окна, основной файл программы, с предустановленными методами XNA Framework содержит в себе:

1) Game1 – конструктор класса, задаёт первоначальные настройки окна, задает координаты для стартового и финишного узлов;

2) Initialize – предустановленный метод фреймворка, отвечающий за инициализацию;

3) MainPoint – метод, визуализирующий стартовый и финишный узлы;

4) DrawMaze – метод, рисующий идеальный лабиринт;

5) BreakWalls – метод, удаляющий непроходимые узлы;

6) GetNeighbours – метод, проверяющий не посещённых соседей для генерации лабиринта;

7) RemoteWalls – метод, соединяющий узлы, путем удаления между ними стен;

8) Path – метод, реализующий этап алгоритма Ли – распространение волны;

9) GetNeighboursPath – метод, возвращающий всех достижимых соседей текущего узла;

10) PathBack – метод, который формирует феромонные тропы;

11) GetNeighboursPathBack – функция, возвращающая соседний узел, помеченный на 1 меньше числа текущего узла;

13) AntsColony – метод, реализующий муравьиный алгоритм;

14) GetAntbours – метод, возвращающий всех достижимых соседей для текущего муравья;

12) LoadContent – предустановленный метод фреймворка, отвечает за первоначальную загрузку текстуры, в данном случае сетки;

13) Update – предустановленный метод фреймворка, основной в программе, отвечает за обновление окна программы в зависимости от времени, также в нем реализованы кнопки перехода между этапами на “Пробел” и запуск муравьиного алгоритма снова на “F2”;

14) Draw – предустановленный метод фреймворка, отвечает за отрисовку текстур во время работы;

в) Matrix.cs –файл для отрисовки сетки текстуры;

г) Ants.cs – файл, хранящий феромонные тропы на матрице узлов.