

Содержание

1	Анализ задачи	3
	Список использованных источников	5

1 Анализ задачи

Постановка задачи: Алгоритм Ли для поиска пути (гексагональная сетка) на равностороннем четырехстороннем (в виде параллелограмма) дискретном рабочем поле. Длина сторон рабочего поля может варьироваться в пределах $[10;100]$. Требуется реализовать указанный в теме алгоритм. При этом вывод итоговых и промежуточных результатов должен быть реализованы в виде *svg* графики.

1.1 Гексагональная сетка

В основе гексагональной сетки (*англ.* hexagon grid) лежит правильный шестиугольник (*англ.* hexagon), внутренние углы которого равны 120° . На гексагональной сетке две параллельные стороны всех шестиугольников могут располагаться либо параллельно оси ординат, либо оси абсцисс. Возьмем первый вариант и выведем математические формулы, которые описывают гексагональную сетку и её элементы на равностороннем треугольном дискретном рабочем поле (*РТДРП*).

Прежде всего необходимо задать размерность шестиугольника *hexSize*. Она определяет расстояние углов от центра фигуры. Размерность параллелограмма обозначим через *dimension*.

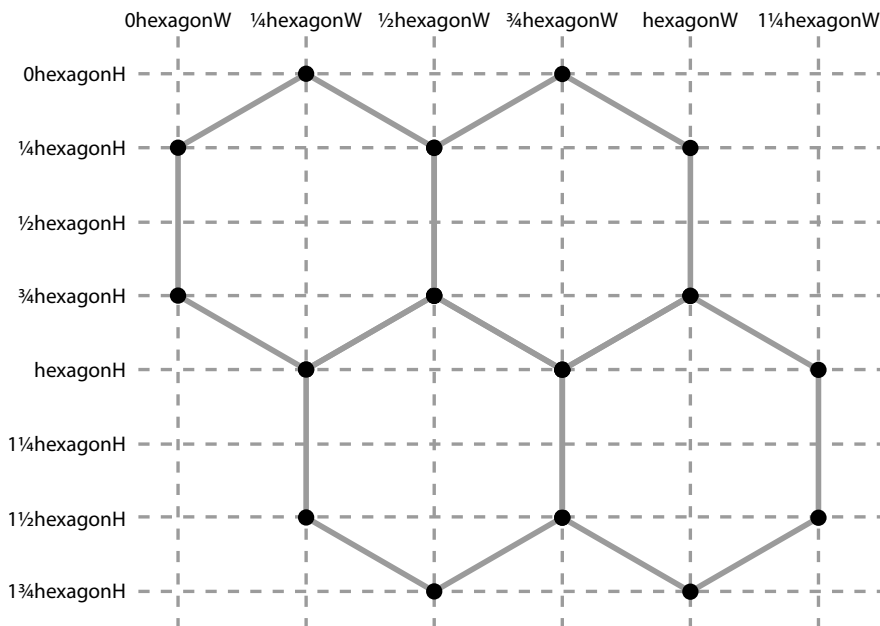


Рисунок 1.1 — Гексагональная сетка.

Высота данного поля $gridHeight$ вычисляется по формуле:

$$gridHeight = hexH(\frac{3}{4}dimension + \frac{1}{4}), \quad (1.1)$$

где $hexH$ — высота шестиугольника, которую можно найти по формуле $2hexSize$. Ширина данного поля $gridWidth$ вычисляется по формуле:

$$gridWidth = hexW \times dimension, \quad (1.2)$$

где $hexW$ — ширина шестиугольника, которая равна $hexSize\sqrt{3}$.

1.2 Алгоритм Ли

Алгоритм Ли — волновой алгоритм поиска пути на карте, алгоритм трассировки. С его помощью можно построить путь, или трассу, между двумя любыми элементами в лабиринте. Из начального элемента распространяется в четырёх направлениях волна. Тот элемент, в который она пришла, образует фронт волны.

Элементы первого фронта волны являются источниками вторичных волн.

Элементы второго фронта генерируют волну третьего фронта и так далее. Процесс заканчивается тогда, когда достигается конечный элемент. На втором этапе строится трасса. Построение производится в соответствии с некоторыми правилами:

а) при построении трассы движение проходит в зависимости от выбранных приоритетов,

б) путевые координаты уменьшаются при переходе от начального элемента к конечному.

Эти приоритеты выбираются в процессе разработки. В зависимости от выбора тех или иных приоритетов получаются различные трассы. Но в любом случае длина трассы остается одной и той же.

Используя волновой алгоритм, можно найти трассу в лабиринте с любым количеством стен. В этом и заключается **преимущество** его использования.

Недостаток алгоритма Ли заключается в том, что при построении трассы требуется большой объем памяти.

Список использованных источников

1. *Patel, Amit*. Hexagonal Grids [Электронный ресурс] (дата обращения: 01.05.2018). — 2015. <https://www.redblobgames.com/grids/hexagons/>.
2. *А., Новиков Ф.* Дискретная математика для программистов / Новиков Ф. А. — Питер, 2013.