

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Отчет по домашнему заданию
Алгоритмы и структуры данных

Вариант задания 21

Студент: Поспелов Илья Михайлович

Группа: ИУ8-53

Преподаватель: Чесноков В.О.

Москва, 2016

Задание:

Одна из распространенных тактик в онлайн-шутерах и не только – rush, или быстрое перемещение в сторону противника. Пусть у вас есть некоторая карта, и две стартовые точки для двух противоборствующих команд. Все игроки, кроме вас, перемещаются с обычной скоростью, осторожно. Определите маршруты движения, которые обеспечат вам внезапность.

Уточнения задания:

- Игроков в каждой команде по 5
 - Все передвигаются с обычной скоростью (шагом)
 - Игрок передвигается в два раза быстрее остальных (бежит)
 - Область видимости игроков не ограничена по дальности, только препятствиями
 - Так как не известно, куда смотрят враги, то считаем их поле обзора равным 360^0
 - Карта и стартовые точки задаются пользователем и подаются как входные данные
 - Требуется рассчитать “оптимальное положение” для игрока (точка, с которой можно будет сделать выстрел)
 - Требуется построить кратчайший путь до “оптимального положения”
-

Входные данные:

Текстовый файл (in.txt) в котором содержится информация о карте размером NxN

(Пример файла совпадает с картинкой, показанной в следующей части)
0-препятствие; 1-проход; 2-старт игрока; 3-старт врага;

```
1 2 1 1 1 1 0 0 0 0 0 0 0
1 1 1 1 1 1 0 1 1 1 1 1 1
0 0 0 0 0 1 0 1 1 1 1 1 1
1 1 1 1 1 1 0 0 0 0 0 1 1
1 1 1 1 1 1 0 1 1 1 1 1 1
1 1 1 1 1 1 0 1 1 1 1 1 1
1 1 1 1 1 1 0 1 1 1 0 1 1
1 1 0 0 0 0 0 1 1 1 0 1 1
1 1 1 1 1 1 0 1 1 0 0 0 1
1 1 1 1 1 1 0 1 1 0 1 1 1
1 1 1 1 1 1 1 1 1 0 1 1 1
0 0 0 0 0 0 0 0 1 0 1 1 1
1 1 1 1 1 1 1 1 1 0 1 3 1
```

Выходные данные:

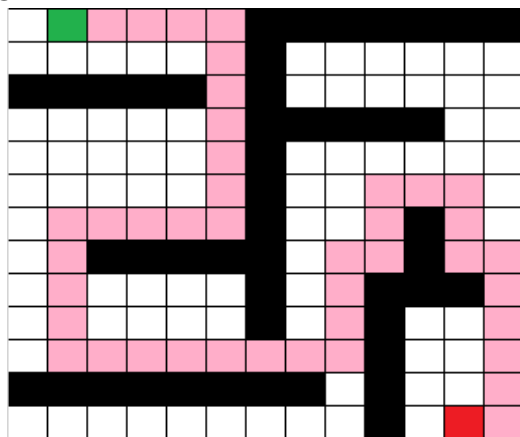
#-препятствие или недостижимые точки; 1-проход; S-старт игрока;

[illegible]

Так как пользователь может задать карту $N \times N$ с любыми допустимыми значениями N (см. входные условия), а расположение точек старта команд может быть максимально далеко и не расположено привычно по углам, то придется перебирать все возможные варианты путей, то задача принадлежит классу P . На некоторых этапах будут убираться лишние клетки\точки карты, которые можно не рассматривать, для ускорения поиска кратчайшего пути.

1. Рассчитываем расстояние в клетках (точках) между стартами команд

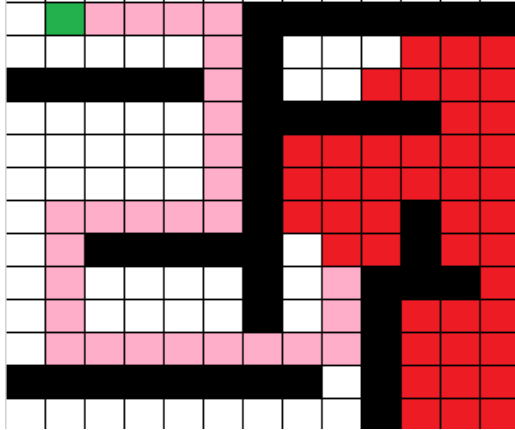
37



Волновой алгоритм поиска кратчайшего пути от зеленой точки до красной. (т.к. карта хранится в двумерном массиве)

2. Делим на 3 с округлением в большую сторону, т.к. за одно и то же время будет пройдено $2/3$ расстояния игроком и $1/3$ врагом
 $\lceil 37/3 \rceil = 13$

3. Находим все клетки, где может оказаться персонаж за это время



Волновой алгоритм из красной точки до зеленой точки, но количество шагов будет ограничено значением из пункта 2.

4. Теперь задачей является нахождение кратчайшего пути до ближайшего места “встречи взглядами”
5. Велика вероятность, что появятся такие “желательные” клетки, куда добраться не будет возможности + всего “желательных” точек может быть несколько, и не известно, какая из них удобней расположена с учетом полученных данных о поле видимости. В качестве алгоритма поиска будет волновой, так как он перебирает по одной клетке в каждую сторону за ход, а значит, вероятность пропустить “желательную” точку сводится к нулю.