Поиск кратчайшего пути

Срочная новость! 18 февраля 2021 года марсоход "Персеверанс" прибыл на Марс для исследования кратера Езеро в рамках миссии НАСА «Марс-2020».

Через несколько дней исследований марсоход провалился сквозь скальную породу внутрь кратера. Судя по первым снимкам это руины, оставшиеся от древней цивилизации.

Во время падения повредился Радиоизотопный термоэлектрический генератор и единственным источником энергии остались литий-ионные батареи, которые заряжаются от солнечной энергии. Но под землёй нет солнечного света. Нужно как можно скорее выбраться на поверхность.

Марсоход укомплектован отдельным беспилотным летательным аппаратом, оснащённый камерами ночного видения, с помощью которого он смог получить карту местности и увидеть выход. Только оказалось что проектировщики марсохода не предусмотрели алгоритмов поиска пути. А электроэнергии на неоптимальный маршрут может не хватить. Марсоход использует радиационно устойчивый одноплатный компьютер на базе процессора RAD750 с частотой 133 МГц и 128 Мбайт динамической памяти и может запускать программы на языке Java. Инженеры НАСА договорились использовать интерфейс RouteFinder (описание ниже). Пока они спорят о том какой алгоритм лучше успей их опередить и написать оптимальный алгоритм и спасти научные исследования Марса. Вычислительной мощности компьютера марсохода и памяти очень мало, постарайся учесть это в своей программе.

Торопись, у него осталось совсем немного энергии чтобы выбраться на поверхность!

```
/**

* Интерфейс поиска маршрута

*/
public interface RouteFinder

{

    /**

    * Поиск кратчайшего маршрута между двумя точками
    * @param map карта
    * @return карта с построенным маршрутом
    */
    char[][] findRoute(char[][] map);
}
```

На вход функции findRoute передается двумерный массив размерностью KxL (1<=K,L<=10000). В качестве элементов данного массива допускаются следующие символы:

- # преграда
- . дорога
- @ начальная точка
- Х конечная точка

Допустимо перемещение только на соседние клетки по вертикали и по горизонтали (по диагонали перемещение недопустимо). В случае, если построение маршрута невозможно, функция findRoute должна вернуть null. В качестве результата необходимо получить массив символов с построенным маршрутом. Маршрут прокладывается символом '+'.

Пример 1

Ввод

```
...@.
.####
....
####.
.X...
```

Вывод

```
+++@.
+###
+++++
####+
.X+++
```

Пример 2

Ввод

```
..X..
#####
....
.@...
```

Вывод

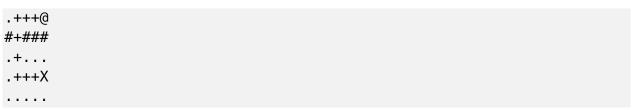
null

Пример 3

Ввод

```
....@
#.###
.....X
```

Вывод



Проверка задания

Проверка задания будет осуществляться автоматически с помощью набора тестов. Тесты буду проверять корректность ответа, а также время работы алгоритма. За каждый тест будут начисляться баллы. Помимо автотестов будет проверяться код человеком при условии, что программа работает.