# F. Космические Рейнджеры (60 баллов)

В параллельной вселенной человечество научилось путешествовать к соседним звёздам.

Для этого построили «Звёздные ускорители» (ЗУ), которые могут перебрасывать корабль на несколько световых лет в одну из четырёх сторон (вперед/назад/вправо/влево). Да, параллельная вселенная двухмерная, не спрашивайте.

ЗУ отличаются мощностью от 1 до 9 — и могут использовать как всю, так и только часть своей мощности для перемещения корабля.

Услуга эта дорогая, но, что удивительно, цена фиксирована и не зависит от затраченной мощности.

Таким образом, выгодно строить маршрут так, чтобы количество межпространственных прыжков было минимальным.

Вы попали в эту вселенную и только что купили себе новенький космический корабль с последней версией местных звёздных карт.

Интерфейс заботливо показывает вам фрагмент галактики с проложенным по нему оптимальным маршрутом.

From: 0:2

To: 8:7

0:2 -> 0:3 -> 4:3 -> 4:7 -> 8:7

===============================================================================

1 1 4 2 5 5 4 2 2

1 5 5 2 1 4

[1] 3 1 5 5 2 5 3

|

[5]--3---5---2--[5] 1 2 2 5

|

1 3 5 | 3 1 4 5

|

1 5 5 4 1 5 5 5 2

|

2 2 2 1 1 4 1 3

|

1 1 3 3 [5]------3---5--[1] 5

1 5 4 2 5 1 3 1 5

1 1 4 5 2 4 3

К сожалению, в очередном обновлении разработчики допустили ошибку, из-за чего модуль расчёта пути сломался.

Вам предстоит починить его.

В качестве решения этого задания отправьте файл .js, в котором объявлена функция pathFinder:

function pathFinder(input) {

// ...

}

## Формат ввода

В параметре input в функцию pathFinder приходит строка следующего формата:

<x1>:<y1>

<x2>:<y2>

<map line>

...

<map line>

* <x1>:<y1> — начальная координата пути, например, 0:2
* <x2>:<y2> — конечная координата пути, например, 8:7
* <map line> — строка карты выражена последовательностью цифр от 0 до 9

Каждая цифра на карте - это то, на сколько шагов можно переместиться из этой точки. Например, из точки с значением 1 можно перейти только на соседние 4 точки.

Количество цифр в каждой строке карты одинаковое.

Количество строк на карте совпадает с количеством цифр в строках.

Максимальный размер карты: 40 x 40.

## Формат вывода

Функция должна вернуть массив со списком оптимальных путей.

Порядок оптимальных путей в массиве не важен.

Пустой массив, если невозможно добраться до конечной точки.

### Пример 1:

Ввод:

2:3

4:4

00014

30020

00000

70100

11100

Вывод:

[["2:3","2:4","1:4","0:4","0:3","0:1","3:1","3:0","4:0","4:4"]]

### Пример 2:

Ввод:

0:2

8:7

0515320501

3150514510

0102010523

5510001000

1402152200

0310530201

0551451213

4101452055

0252411510

4110045253

Вывод:

[]

### Пример 3:

Ввод:

2:3

3:0

2012

3001

7000

1920

Вывод:

[["2:3","0:3","0:2","0:0","2:0","3:0"],["2:3","0:3","0:2","0:1","3:1","3:0"]]

## Примечания

* Решение будет проверяться в браузере (**Chrome 78**).
* Можно использовать синтаксис до **es2018** включительно.