California, Rome, Dubai, Paris, Amsterdam, London, Rousse, Emiliyan is travelling again.

He is in cell *S* of a given map and wants to reach cell *F*. The remaining cells of the map are *.* and *#*. Cells *S*, *F* and *.* symbolize land and the *#* cells symbolize water. Unfortunately, he cannot swim and can only move on land. For a unit of time, he can move to cells that are directly up, down, left, or right from his current one.

There are also two-way airline routes. Each is characterized by 5 parameters , which means he can move from cell to cell or from cell to cell for units of time. Cells and are part of the land.

Find the minimum amount of time it takes to get from *S* to *F*.

**Input**

The first line of the file **tourism.in** contains the numbers and . Each of the next lines contains symbols that describe the map. It is guaranteed that there is exactly 1 *S* cell and exactly 1 *F* cell.

The next line contains the number , the number of airline lines, and the next lines contain 5 numbers each, , characterizing the corresponding airline route.

**Output**

On the only line of the file **tourism.out**, print the minimum time it takes to get from S to F. If there is no possible path, print .

**Constraints**

**Time limit: 0.4 sec.**

**Memory limit: 256 MB.**

**Sample test**

|  |  |
| --- | --- |
| **Input (tourism.in)** | **Output (tourism.out)** |
| 5 3  S#.  ...  ###  ...  ..F  1  5 3 1 3 100 | 104 |