

12807 GPS

You have to write a program that provides navigation instructions for going from one place to another based on a map.

The program will receive a map and the coordinates of the origin and destination points, and should provide turn by turn instructions for reaching from the origin to the destination following the shortest path.

Input

The input format is as follows:

- An integer in a single line which says the number of problems to solve. Then, for each problem:
 - Two integers in a line separated by a space indicating the *width* and *height*, respectively, of the map (1000 or less).
 - The map, which takes $height + 2$ lines of exactly $width + 2$ characters:
 - The first and last lines are decorative and should be ignored.
 - The first and last characters of each line are also decorative and should be ignored.
 - The rest of characters represent the map:
 - * Each character represents an area cell of $1\text{ km} \times 1\text{ km}$.
 - * Each one can be either '*' (asterisk, used to represent roads) or ' ' (space, used to represent everything else).
 - * The user can move from one cell with '*' to any adjacent cell with '*'. Each cell has at most 4 adjacent cells (north, south, east and west; i.e.: moving in diagonal is not allowed).
 - * The upper left corner has coordinates 0,0. Coordinates grow to the right and downwards.
 - * North is upwards.
 - Four integers in a line, separated by spaces, indicating the x and y coordinates of the origin point, and the x and y coordinates of the destination point, respectively.

Output

The output for each problem consists of a list of step by step instructions to go from the origin to the destination through a shortest path.

In case that there are more than one shortest path, the program should provide the one which, at each step, goes preferably east, then south, then west and finally north. That is, given two shortest routes, the most preferable one is that which, once it arrives to the first different step, goes east if possible, otherwise goes south if possible, otherwise goes west if possible and otherwise goes north.

If there is no way to go from the origin to the destination using the roads of the map, then the program should print 'No route found.'

To describe a route, the program should print one instruction per line:

- The first instruction should be one of 'Turn to the north.', 'Turn to the south.', 'Turn to the east.' or 'Turn to the west.'

- The following instructions can be ‘Turn left.’, ‘Turn right.’ or ‘Continue x km.’, where x is an integer number.
- The last line should be ‘You have reached your destination.’

The program must use as few instructions as possible to describe the route. Also, it should print a blank line after each solution (so, and the end of the output there are two new line characters).

Sample Input

```

2
10 8
+-----+
|           |
|   ***** |
|   *   *   |
|  ***   *   |
|           * |
|         *** |
|           * |
|           * |
+-----+
1 3 7 7
15 15
+-----+
|***       |
| *        |
| *        |
| *        |
| *****  |
|     **** |
|     *****|
|     **** |
|         ****|
|             *|
|             ***|
|             * *|
|             ***|
|             * |
|             *****|
+-----+
0 0 14 14
    
```

Sample Output

```

Turn to the east.
Continue 2 km.
Turn left.
Continue 2 km.
Turn right.
Continue 4 km.
    
```

Turn right.
Continue 6 km.
You have reached your destination.

Turn to the east.
Continue 2 km.
Turn right.
Continue 4 km.
Turn left.
Continue 7 km.
Turn right.
Continue 4 km.
Turn left.
Continue 2 km.
Turn right.
Continue 2 km.
Turn left.
Continue 1 km.
Turn right.
Continue 2 km.
Turn right.
Continue 1 km.
Turn left.
Continue 2 km.
Turn left.
Continue 3 km.
You have reached your destination.