acm International Collegiate Programming Contest 2001

South Central USA Regional Programming Contest

Problem #8: Asteroids!

Introduction

You're in space.
You want to get home.
There are asteroids.
You don't want to hit them.

Input

Input to this problem will consist of a (non-empty) series of up to 100 data sets. Each data set will be formatted according to the following description, and there will be no blank lines separating data sets.

A single data set has 5 components:

1. *Start line* - A single line, "`START N`", where 1 <= *N* <= 10.
2. *Slice list* - A series of *N* slices. Each slice is an *N* x *N* matrix representing a horizontal slice through the asteroid field. Each position in the matrix will be one of two values:
   'O' - (the letter "oh") Empty space
   'X' - (upper-case) Asteroid present
3. *Starting Position* - A single line, "*A B C*", denoting the <A,B,C> coordinates of your craft's starting position. The coordinate values will be integers separated by individual spaces.
4. *Target Position* - A single line, "*D E F*", denoting the <D,E,F> coordinates of your target's position. The coordinate values will be integers separated by individual spaces.
5. *End line* - A single line, "`END`"

The origin of the coordinate system is <0,0,0>. Therefore, each component of each coordinate vector will be an integer between 0 and *N*-1, inclusive.

The first coordinate in a set indicates the column. Left column = 0.
The second coordinate in a set indicates the row. Top row = 0.
The third coordinate in a set indicates the slice. First slice = 0.

Both the Starting Position and the Target Position will be in empty space.

Output

For each data set, there will be exactly one output set, and there will be **no blank lines** separating output sets.

A single output set consists of a single line. If a route exists, the line will be in the format "*X Y*", where *X* is the same as *N* from the corresponding input data set and *Y* is the least number of moves necessary to get

your ship from the starting position to the target position. If there is no route from the starting position to the target position, the line will be "NO ROUTE" instead.

A move can only be in one of the six basic directions: up, down, left, right, forward, back. Phrased more precisely, a move will either increment or decrement a single component of your current position vector by 1.

Sample Input

```
START 1
O
0 0 0
0 0 0
END
START 3
XXX
XXX
XXX
OOO
OOO
OOO
XXX
XXX
XXX
0 0 1
2 2 1
END
START 5
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
XXXXX
XXXXX
XXXXX
XXXXX
XXXXX
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
OOOOO
0 0 0
4 4 4
```

Sample Output

```
1 0
```

```
3 4
NO ROUTE
```
