Baltic OI '11 P4 - Treasures and Vikings

Time limit: 1.0s **Memory limit:** 64M

Baltic Olympiad in Informatics: 2011 Day 1, Problem 4

You have a treasure map that is arranged into an $N \times M$ grid. A grid square may be either sea or part of an island. In addition, the map shows the treasure and an enemy Viking ship that occupies one (sea) square. Finally, for convenience you have also drawn your own position.

Now you must set up a fixed route to get the treasure. The route must start at your position, end at the treasure, and consist of a sequence of moves. In each move, you can go only to an (horizontally or vertically) adjacent square that is not part of an island. But beware: The Viking ship might follow you, using the same kind of moves! After each of your moves according to your route, the Viking ship may move or not. Your move and the Vikings' reaction together is called a round.

After every round, the following checks are made:

- If you are in line with the Viking ship (you are in the same vertical or horizontal line as the Viking ship with only sea between the Viking ship and you), you are dead.
- If you aren't dead and at the treasure spot, you get the treasure.

Write a program that decides whether it is possible to set up a **fixed** route **in advance** such that you can get the treasure by following this route and will not get killed by the Vikings – no matter how the Viking ship moves.

Constraints

$1 \le N, M \le 700$

Subtask 1 [40%]

$1 \le N, M \le 200$

Subtask 2 [60%]

No additional constraints.

Input Specification

The first line of input contains two integers N and M, the dimensions of the map. Each of the following N lines contain M characters.

Each character describes a square in the map, and is either `.` (sea), `I` (part of an island), `V` (the Viking ship), `Y` (your position), or `T` (the treasure).

Each of `V`, `Y`, and `T` will occur exactly once.

Output Specification

The only line of the output must contain the string `YES`, if it is possible to set up a route to get the treasure, or `NO` otherwise.

Sample Input 1

```
5 7
Y.....V
..I....
..IIIII
.......
...T...
```

Sample Output 1

```
YES
```

Sample Input 2

```
5 7
Y....V.
..I....
..IIIII
.......
...T...
```

Sample Output 2

```
NO
```

Sample Input 3

```
2 3
.YT
VII
```

Sample Output 3

```
NO
```

Explanation for Sample 3

In the first example, the following route will let you get the treasure: Down, Down, Down, Right, Right, Right, Down.

In the second and third examples, there is no route to the treasure that you will survive.