Hypertheseus

`theseus.c`, `theseus.C`, `theseus.p`, `theseus.java`

For one of their clients, ACM prepares a TV advertisement that features ancient heroes, for example Prometheus, Achilles, Odysseus and many others. To demonstrate how difficult was life for these heroes, ACM runs computer simulations of their most famous tasks. In this problem, we will try to solve the problem of Theseus.

Theseus was an Athenian hero who killed Minotaur, the half-man, half-bull monster residing in an inescapable Labyrinth constructed by Daedalus. The hardest part of Theseus' task was not killing the monster — the legend says Minotaur was just sleeping when Theseus came to him. Thus, the hero was able to batter the monster with his bare hands. But then the *real* challenge came: to find the way out. As you may know, Ariadne, beautiful daughter of king Minos, played an important role in this part. But that's a different story.

As the technology advanced a lot since the ancient times, there are several important differences in today's simulations:

1. Labyrinth-making skills advanced as well. Two-dimensional labyrinths are not a big challenge anymore. Thus, our labyrinth is d-dimensional. It looks like a regular "grid" of $n_1 \times n_2 \times \ldots \times n_d$ (hyper)cubes, each of them being either empty (corridor) or filled with a rock (wall). Theseus moves in steps, each step means travelling between two neighboring empty hypercubes in any dimension. Two hypercubes are considered neighboring, if and only if the difference in their space coordinates is exactly one in one dimension, and zero in all other dimensions.

2. Our Minotaur is stronger, thus, it is no more possible to batter him with bare hands. Theseus must use a sword which lies somewhere inside the labyrinth. Note that before he takes the sword, he cannot go through the hypercube where Minotaur resides!

Input Specification

The input consists of several labyrinth descriptions. Each description begins with a line containing a single integer number d ($2 \leq d \leq 20$) — the number of dimensions. On the second line of each description, there are d integer numbers $n_1, n_2, \ldots, n_d$ separated by spaces. These numbers give the size of the labyrinth in every dimension, measured in unit hypercubes ($\forall i : 2 \leq n_i$). You may assume that the total number of hypercubes in any labyrinth will not exceed $2^{20} = 1048576$.

Then a labyrinth map follows, the description is defined recursively: Two-dimensional labyrinth (of the size $n_1 \times n_2$) is described by n_2 lines containing n_1 characters each. Every character describes a single hypercube ("square" in the case of 2 dimensions) and is either "`#`" (rock), "`.`" (free space), or one of uppercase letters "`T`" (Theseus' position), "`M`" (Minotaur), and 'S' (sword). Each of these three objects will appear exactly once in the whole labyrinth. The hypercubes containing the letters are considered "empty" otherwise, i.e., Theseus can walk (and must walk) through them.

For better clarity of the input, each two-dimensional map is followed by one empty line.

For $d > 2$, any d-dimensional labyrinth is a sequence of n_d "layers", each layer being a description of a $(d-1)$-dimensional labyrinth.

The last labyrinth description is followed by a line containing zero. This line terminates the input and no output should be produced for it.

Output Specification

For each labyrinth, output the sentence "`Theseus needs` S `steps.`", where S is the smallest possible amount of steps needed to reach the hypercube with the sword, the hypercube with Minotaur, and then back to the original Theseus' position (where the exit is supposed to be). It is not possible to leave the labyrinth area, i.e., all hypercubes outside the labyrinth are considered rocks. Exit may appear in an inner hypercube, it may not reside on the "border".

If it is not possible to solve the situation at all, output sentences "`No solution. Poor Theseus!`"

Sample Input

```
2
3 3
T..
M#.
S..

2
3 2
T#S
..M

3
4 4 3
#S..
###.
#M..
###.

.###
.###
.###
....

....
###.
###T
####

0
```

Output for Sample Input

```
Theseus needs 8 steps.
No solution. Poor Theseus!
Theseus needs 40 steps.
```