You are participating in an adventure game called **An Amazing Race**. You are now at the coast of a desert island, and you have to visit a number of checkpoints, then go to the target at the highest point of the island. Of course, the first person who arrives at the target wins the race.

The temperature is now 45°C. It is a must to drink lots of water in such hot environment. However, you are not allowed to take any drinking water with you in your journey (!!). Water is only supplied at checkpoints as well as the starting and target locations, so you would like to choose your route such that the maximum distance between two consecutively visited water-supplying spots is minimized. Observing this, you would then like to find a shortest route.

You may write a program to help you.



~~###~~~~~#~~

## Input

The input contains several test cases. The first line of each case gives three integers r, c, c, c (3  $\leq$  r, c  $\leq$  20; 1  $\leq$  c  $\leq$  10). Then there are r lines, each with c characters, which is the map of the island. A tilde sign 'in the map denotes sea area, which cannot be travelled past. A hash sign '#' denotes normal land area. The island is of course surrounded by sea. 'S' marks the starting location, while 'T' is the target. As mentioned above, the starting location is always on the coast. Each checkpoint is denoted by a unique letter from the set  $\{A, \ldots, K\}$ . You must visit at least c different checkpoints, and then finish your journey at the target location 'T'. You may visit a checkpoint more than once, and the visiting order is also unimportant.

In each step, you may travel from a land area to another only if they share a common edge, i.e. you may only go north, south, east or west (by 'north' we mean facing upwards). The island is rather flat, hence the time required to travel to an adjacent square is the same everywhere on the island (no need to consider the gradient). You can further assume that all checkpoints are safely reachable, so it is indeed possible for you to win the race, provided that your route is planned optimally observing the above-mentioned requirements. The sample input corresponds to the figure.

## **Output**

For each case, your program should output a single line with the description of an optimal path. Use the letters 'N', 'S', 'E' and 'W' to denote north, south, east and west respectively. If there are many solutions, output the lexicographically smallest one.

**Explanation:** The optimal path is shown in red on the right:

## Sample Input

15 15 2
~~~~~~~~~~~~
~~###~~~~#~~~
~##T###~#####~~
~~##########
~~#########
~~##########
~##B#######*
~#########
~~##########
~~#########
~#########
~~###~S#####~~~
~~~##~~####~~~~
##

## Sample Output

EEEENNNNNWWWWWWWNNNN