Ants

At the beginning of every game, you are given a set of ants which all start in your own ant hill. From there, you can send them off to find food or fight your enemies.

Rules

The surroundings as seen by the ants will be described using the symbols shown in Fig 1. Note that these symbols are always of qualitative nature: a '$' can be any number from 1 to 100 resources of food. Similarly, one can place several ants of the same team on a single tile. If an ant is standing on top of a resource tile, the food is not seen. Also, ants are not seen inside their ant hill.

```
# - solid wall (impenetrable)
$ - food resources (0 < r <= 10)
O - your own ants (one or more)
X - enemy ants (one or more)
H - your own ant hill
E - an enemy ant hill
```
Fig 1: List of the map symbols

The ultimate goal of the game is for your ant population to remain standing as the last team. This can be done by collecting food resources (to get more ants) and by fighting your enemies.

- Food is collected from the food resource fields '$' by your ants, one by one.

- Once you gather 8 resources, an extra ant is born at your ant hill

- When fighting, two ants always kill each other leaving 10 resources back on that tile.

- When an ant hill is attacked, the attacked team loses 3 random ants, each leaving 10 food.

(Note however, that no more than 10 food resources can be on any tile and that no food will be left if an ant is killed while inside the ant hill.)

```
3
0 -1 0
.......
.......
...H...
...O...
.O.....
...O...
$$.....
-2 -2 1
#......
#....H.
#....O.
#..O...
#....O.
#$$$...
##$$$..
0 -3 0
...H...
...O...
.O.....
...O...
$$.....
$$$....
#$$$$..
```
Fig 2: Input

Interaction

Every turn, your program is given the current number N of ants, followed by each ant's properties:

- The first line contains 3 integer values depicting the ant's relative coordinates (X_i, Y_i) from your ant hill and whether the ant is currently carrying food or not.

- On the following 7 lines, the 7x7 surroundings of this particular ant are printed.

Your program then has to print a string of the length N, consisting of the letters 'N', 'W', 'S' and 'E'. These movement directions must be given in the same order the ants were listed in the input.

- Ants may not move onto a solid wall tile '#'.

- An ant not carrying food will automatically pick up a food when moved onto a food tile.

- An ant carrying food will automatically unload this resource upon return to the ant hill.

- If an ant is moved onto a field containing enemy ants, it kills one of them along with itself.

- If an ant is moved onto the enemy ant hill 'E', three ants of this team are picked at random and killed. *This occurs even if there are ants currently inside the enemy ant hill!*

Ants killed in combat will not be listed any more during the next turn. If all ants of a player are killed, his ant hill is removed from the map too.

A simple input example for a turn can be seen in the upper box on the right. A possible order string for this situation would be "SNW", moving the first ant down, the second up and the third to the left.

The complete game situation for the sample can be seen on the right. The dimensions of the map will not exceed 1000x1000 tiles.

```
################################
##........######........$$$$$#####
#.........##.............$$$$$###
#....H.....#.............$$$##
#....O.....#...$$$$X........$$$#
#..O.........$$$$$$.XX.........#
#....O.......$$$$$$....X.......#
#$$$..........$$$$..#...XX.....#
##$$$.................#....E....#
###$$$$$...........###.........#
#####$$$$........######.......##
################################
```
Fig 3: Complete situation for better understanding

Server

You can upload an try your programs against others on the ant server: http://172.7.10.7:

In case of questions, feel free to ask us directly at the translation table of Switzerland.