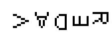
WC '98 P1 - Optical Recognition

Time limit: 1.0s **Memory limit:** 16M

Woburn Challenge 1998

Luke Skywalker wants to break into Darth Vader's quarters on an Imperial Cruiser to steal his supply of nasal spray. This nasal spray is rumoured to alleviate his incessant wheezing, resulting in improved Stormtrooper discipline (because they won't be laughing at him so often), resulting in a much deadlier attack force. But alas, Luke is a wimp and so he sends R2D2 to do the dirty work. Unfortunately, Luke was not really a bright lad and didn't find out which room belonged to Vader; therefore R2D2 has to find the correct room based on the name that is printed on the door. However, the Empire is smarter than Luke (heck, Chewy is smarter than Luke) and has made life a little tough for the droid by not orienting all the letters correctly. For example, if the room belonged to VADER, any of the 5 letters could be rotated by some multiple of 90° and so, VADER could actually be printed on the door as  .

Your job is to equip R2D2 with the algorithm to recognize which door he should break into.

Input Specification

You will be given the "shape" of the 5 letters that make up the name that R2D2 is going to look for as follows (each letter will be represented by a 7×7 grid):

Line 1: 1st letter
Line 2-8: The shape of the 1st letter
Line 9: 2nd letter
Line 10-16: The shape of the 2nd letter
Line 17: 3rd letter
Line 18-24: The shape of the 3rd letter
Line 25: 4th letter
Line 26-32: The shape of the 4th letter
Line 33: 5th letter
Line 34-40: The shape of the 5th letter

The next 21 lines will represent a 21-line grid ($1 \leq$ width ≤ 150) which is the grid you have to scan to determine which letters are represented in the grid. Note that, in the grid, a letter can be the same size, twice as big or 3 times as big as the letter you are given. In addition, a letter can be rotated by some multiple of 90°. Letters will be separated by at least one empty column, i.e. one column of nothing but the letter `o`. Letters need not be the same size or rotation, and they don't need to occupy the same lines in the grid.

You will then be given more 21-line grids (same letters) that you have to decipher. The number of grids you get will be variable and you must keep expecting grids until you see the end of the file. Note that it is not necessary for the grid to contain any letters.

Output Specification

You will output the word that is shown in each grid (a separate line for each grid). If any of the letters in the grid matches NONE of the 5 letters whose shape you have been given, then output only `NONE` (all capitals) for that grid. If there are no letters in the grid at all, then output `EMPTY`.

Sample Input

```
V
*ooooo*
*ooooo*
o*ooo*o
o*ooo*o
oo*o*oo
oo*o*oo
ooo*ooo
A
ooo*ooo
oo*o*oo
oo*o*oo
o*****o
o*ooo*o
*ooooo*
*ooooo*
```

... and so on (3 more letters will be given)

```
**ooooooooooo**ooooooooooooooooooooooooooo
**ooooooooooo**ooooooooooooooooooooooooooo
**ooooooooooo**ooooooooooooooooooooooooooo
**ooooooooooo**ooooooooooooooooooooooooooo
oo**oooooo**ooooooooooooooo*ooooooooooooo
oo**oooooo**ooooooooooooooo*o*ooooooooooo
oo**********ooooooooooooooo*o*ooooooooooo
oo**********oooooooooooooo*ooo*ooooooooooo
oooo**oo**ooooooooooooooo*ooo*ooooooooooo
oooo**oo**ooooooooooooooo*ooooo*ooooooooooo
oooo**oo**ooooooooooooooo*ooooo*ooooooooooo
oooo**oo**oooooooooooooooooooooooooooooooo
oooooo**ooooooooooooooooooooooooooooooooo
oooooo**ooooooooooooooooooooooooooooooooo
ooooooooooooooooooooooooooooooooooooooooo
ooooooooooooooooooooooooooooooooooooooooo
ooooooooooooooooooooooooooooooooooooooooo
ooooooooooooooooooooooooooooooooooooooooo
ooooooooooooooooooooooooooooooooooooooooo
ooooooooooooooooooooooooooooooooooooooooo
```

Sample Output

```
AV
```