



PROBLEM 7 - COLOUR GRIDS

You are given a number of plastic tiles, all of the same size, and a target pattern. Your task is to assemble the tiles into a pile so that, looking from the top, it matches the target pattern. Each tile is composed of 16 squares of transparent or coloured plastic (opaque and identically coloured on both faces), arranged in a 4x4 grid. The target pattern is also a grid of colours, similar to a tile, but without transparencies. The tiles can be rotated clockwise through multiples of 90° and/or flipped about a horizontal axis. This leads to 8 possible transformations – rotated 0, 1, 2 or 3 times, or a flip followed by the same sequence of rotations. These transformations, in the above order, are numbered 0 to 7. The following diagrams provide an example, with colours denoted by upper case letters, and dots (‘.’) representing transparency:

Transform #0 (original)	Transform #1	Transform #2	Transform #3																																																																
<table border="1"> <tr><td>R</td><td>R</td><td>G</td><td>G</td></tr> <tr><td>R</td><td>R</td><td>.</td><td>G</td></tr> <tr><td>Y</td><td>Y</td><td>B</td><td>B</td></tr> <tr><td>Y</td><td>Y</td><td>B</td><td>B</td></tr> </table>	R	R	G	G	R	R	.	G	Y	Y	B	B	Y	Y	B	B	<table border="1"> <tr><td>Y</td><td>Y</td><td>R</td><td>R</td></tr> <tr><td>Y</td><td>Y</td><td>R</td><td>R</td></tr> <tr><td>B</td><td>B</td><td>.</td><td>G</td></tr> <tr><td>B</td><td>B</td><td>G</td><td>G</td></tr> </table>	Y	Y	R	R	Y	Y	R	R	B	B	.	G	B	B	G	G	<table border="1"> <tr><td>B</td><td>B</td><td>Y</td><td>Y</td></tr> <tr><td>B</td><td>B</td><td>Y</td><td>Y</td></tr> <tr><td>G</td><td>.</td><td>R</td><td>R</td></tr> <tr><td>G</td><td>G</td><td>R</td><td>R</td></tr> </table>	B	B	Y	Y	B	B	Y	Y	G	.	R	R	G	G	R	R	<table border="1"> <tr><td>G</td><td>G</td><td>B</td><td>B</td></tr> <tr><td>G</td><td>.</td><td>B</td><td>B</td></tr> <tr><td>R</td><td>R</td><td>Y</td><td>Y</td></tr> <tr><td>R</td><td>R</td><td>Y</td><td>Y</td></tr> </table>	G	G	B	B	G	.	B	B	R	R	Y	Y	R	R	Y	Y
R	R	G	G																																																																
R	R	.	G																																																																
Y	Y	B	B																																																																
Y	Y	B	B																																																																
Y	Y	R	R																																																																
Y	Y	R	R																																																																
B	B	.	G																																																																
B	B	G	G																																																																
B	B	Y	Y																																																																
B	B	Y	Y																																																																
G	.	R	R																																																																
G	G	R	R																																																																
G	G	B	B																																																																
G	.	B	B																																																																
R	R	Y	Y																																																																
R	R	Y	Y																																																																
Transform #4	Transform #5	Transform #6	Transform #7																																																																
<table border="1"> <tr><td>Y</td><td>Y</td><td>B</td><td>B</td></tr> <tr><td>Y</td><td>Y</td><td>B</td><td>B</td></tr> <tr><td>R</td><td>R</td><td>.</td><td>G</td></tr> <tr><td>R</td><td>R</td><td>G</td><td>G</td></tr> </table>	Y	Y	B	B	Y	Y	B	B	R	R	.	G	R	R	G	G	<table border="1"> <tr><td>R</td><td>R</td><td>Y</td><td>Y</td></tr> <tr><td>R</td><td>R</td><td>Y</td><td>Y</td></tr> <tr><td>G</td><td>.</td><td>B</td><td>B</td></tr> <tr><td>G</td><td>G</td><td>B</td><td>B</td></tr> </table>	R	R	Y	Y	R	R	Y	Y	G	.	B	B	G	G	B	B	<table border="1"> <tr><td>G</td><td>G</td><td>R</td><td>R</td></tr> <tr><td>G</td><td>.</td><td>R</td><td>R</td></tr> <tr><td>B</td><td>B</td><td>Y</td><td>Y</td></tr> <tr><td>B</td><td>B</td><td>Y</td><td>Y</td></tr> </table>	G	G	R	R	G	.	R	R	B	B	Y	Y	B	B	Y	Y	<table border="1"> <tr><td>B</td><td>B</td><td>G</td><td>G</td></tr> <tr><td>B</td><td>B</td><td>.</td><td>G</td></tr> <tr><td>Y</td><td>Y</td><td>R</td><td>R</td></tr> <tr><td>Y</td><td>Y</td><td>R</td><td>R</td></tr> </table>	B	B	G	G	B	B	.	G	Y	Y	R	R	Y	Y	R	R
Y	Y	B	B																																																																
Y	Y	B	B																																																																
R	R	.	G																																																																
R	R	G	G																																																																
R	R	Y	Y																																																																
R	R	Y	Y																																																																
G	.	B	B																																																																
G	G	B	B																																																																
G	G	R	R																																																																
G	.	R	R																																																																
B	B	Y	Y																																																																
B	B	Y	Y																																																																
B	B	G	G																																																																
B	B	.	G																																																																
Y	Y	R	R																																																																
Y	Y	R	R																																																																

Write a program that will read in a sequence of tiles and a target pattern and determine whether the target pattern can be made by some arrangement of all or some of the input tiles.

- If there are several ways of achieving the target, then choose the pile with fewest tiles.
- If there are still several ways of achieving the target, then at every point in the pile, in a top-down order, choose first the lowest numbered appropriate tile and, if still needed, the lowest numbered appropriate transformation.



INPUT FORMAT

The input will consist of one or more of problems. The first line of each problem contains the title of the problem as a string of 1 to 30 characters other than space, followed by a space, and the number of tiles n ($1 \leq n \leq 10$). This is followed in turn by $n+1$ lines, the first n lines specifying the tiles (implicitly numbered 0 to $n-1$); and the last line specifying the target pattern. Each of these $n+1$ lines contains 4 blocks of 4 characters, either an upper case letter representing a colour, or a full stop ('.') representing transparency. Blocks are separated by single spaces and represent grid rows, in successive row order. Characters composing blocks represent grid cells, in successive column order. The end of input is signified by a line consisting of a single '#'.

SAMPLE INPUT:

```
OBVIOUS 1
RRGG RRGG YYBB YYBB
BBGG BBGG YYRR YYRR
EASY 2
BBBB BBBB ..BB ..BB
RRRR RRRR .... ....
RRRR RRRR BBBB BBBB
Is-this-possible? 2
BBBB BBBB ..BB ..BB
RRRR RRRR .... ....
RRBB RRBB BBRR BBRR
#
```

OUTPUT FORMAT

Output consists of one line for each problem, consisting of the title of the problem, followed by a space, and either the word 'noway' (all lowercase) if the target cannot be achieved, or a description of the pile if the target can be achieved. The tiles in the pile should be listed in order from the top of the pile downwards, in the form: tile number (0 through $n-1$), slash ('/'), transformation number (0 through 7). Separate descriptions of successive tiles by single spaces.

SAMPLE OUTPUT:

```
OBVIOUS 0/7
EASY 1/0 0/1
Is-this-possible? noway
```