Last summer was so hot that it was impossible to do anything but stay in the bathroom and solve puzzles. These games were so boring that you decided to write programs to solve them. This gave you a good reason to go back to the lab and benefit from its air conditioner.

Here is the game you decided to solve automatically. Take a grid with n rows and m columns, with a letter in each cell. From the grid's cells, one can read words horizontally, vertically or diagonally, oriented forwards or backwards. You also have a list of words, and the goal is to highlight all occurrences of these words on the grid. The letters in non-highlighted cells of the grid form a word (called the *code word*) that can be read from top to bottom and left to right. The goal of the program is to find the code word automatically.

Input

The input file contains several test cases.

Each case begins with the two integers n and m on a line ($0 < n, m \leq 40$), separated by a blank. The second line contains the number k ($0 < k \leq 200$) of words in the list. Then follow the k words, one per line. Words do not have more than 22 letters. After these, follows the grid, that is to say n rows filled with m uppercase letters.

Output

For each case, write the code word on a line.

Note: The sample input data corresponds to the array:

B	B	A	R	E	D
A	C	A	L	H	F
R	W	O	E	I	I
E	R	O	S	N	N
D	A	R	N	D	E
A	L	L	Y	G	S

Looking for the word `ALLY`, we find it at the bottom:

B	B	A	R	E	D
A	C	A	L	H	F
R	W	O	E	I	I
E	R	O	S	N	N
D	A	R	N	D	E
A	**L**	**L**	**Y**	G	S

After highlighting it, we look now for `BARE` that appears twice:

B	**B**	**A**	**R**	**E**	D
A	C	A	L	H	F
R	W	O	E	I	I
E	R	O	S	N	N
D	A	R	N	D	E
A	**L**	**L**	**Y**	G	S

We then treat `BARED` that also appears twice:

B	**B**	**A**	**R**	**E**	**D**
A	C	A	L	H	F
R	W	O	E	I	I
E	R	O	S	N	N
D	A	R	N	D	E
A	**L**	**L**	**Y**	G	S

Proceeding until the end of the list, we are left with:

	C	A			
			E		
			S		
	A	R			

Thus, it results as sample output.

Sample Input

```
6 6
7
ALLY
BARE
BARED
FINES
HIND
LORD
WONG
BBARED
ACALHF
RWOEII
EROSNN
DARNDE
ALLYGS
```

Sample Output

```
CAESAR
```