

Word Searching

Filename: word

The Problem

Arup's step-daughter Simi really likes word searches. Whenever he goes out to dinner, Simi insists that Arup help her with finding all of the words on the kids "activity" sheet. At first this activity was quite engaging for Arup. He loved the satisfaction that he'd get when uncovering the next word. However, as time went on, Simi didn't move on to other games, and searching for words became less rewarding, as hard as that is to believe. Arup has decided that he wants his BHCSIers to write a program for him that will automatically find all the words in a word search.

However, he realizes that games change, and he anticipates a day when there will be word searches that allow mismatching words! For example, the word "BROCCOLI" might not appear exactly in the grid, but maybe one diagonal might store "BRACCOLY". In this case, we see that when we line up the word being searched with this portion of the grid, all but 2 letters match. Arup wants your program to find the maximum "score" for each word to be searched for. For all words that are in the search grid exactly, a score of 100 is awarded. For all words that do not appear in the search grid exactly, the score awarded will equal the maximum number of letters that match in valid placement on the grid of the exact same number of letters. Thus, in the case described above, the score awarded would be six, since six letters match between the word "BROCCOLI" and the area of the search grid storing "BRACCOLY." (This assumes that no other valid placement does better.)

It must be noted that a partial match that extends off the board scores no points. Thus, given the following grid:

O	M	P	U	T	E	R	X	R	C
A	B	C	D	E	F	G	H	I	J
X	Q	D	C	O	M	P	T	E	R

The score awarded when searching for "COMPUTER" is 3. In this case, note that if we line up the word "COMPUTER" so that the "C" is to the left of the top left corner of the grid, 7 letters match. But, since the "C" is off the grid, this match is invalid. Similarly, we are not allowed to match "COMPUTER" with the string starting on the third row and fourth column, because in this match (where "COMP" would match), the R in "COMPUTER" would go off the board.

The best match is lining up "COMPUTER" with the third row and third column going in the forward horizontal direction. In this match, the last three letters "TER" line up to give this word a score of three.

Finally, any letter in the grid can be matched with as many words as possible, so there's no rule against having two different matches that completely overlap.

The Input

The first line of the input file will consist of a single integer n representing how many word searches you will solve. For each set of data, the first line of input will contain three integers r ($0 < r < 80$), c ($0 < c < 80$), representing the number of rows and columns, respectively, for this word search puzzle, and k ($0 < k < 80$), the number of words for which to search. The following r lines will contain exactly c uppercase letters representing the corresponding row of the word search. After these r lines, the following k lines will each contain one word for which to search. Each of these words will be unique (within a single word search) and comprised of in between 1 and 79 uppercase letters.

The Output

For each word search, output a single line with the following format:

Maximum Score for Word Search #m: X.

where m is the number of the word search starting with 1 and X is the maximum number of points that can be achieved for the given word search in question.

Sample Input

```
2
3 10 3
OMPUTERXRC
ABCDEFGHIJ
XQDCOMPTER
COMPUTER
FED
RATE
3 3 4
CAT
HAT
DOG
CAT
HAT
DOG
TAD
```

Sample Output

```
Maximum Score for Word Search #1: 105.
Maximum Score for Word Search #2: 400.
```