

Almost everybody has tried to solve a word search puzzle, where one is given a square grid of letters and attempts to find “hidden” words embedded horizontally, vertically or diagonally in the grid. In this problem, we generalize the traditional word search puzzle by relaxing the rule that words must be found in horizontally, vertically or diagonally contiguous positions. In our puzzle, words may be stretched (and/or bent) so that successive letters in a word may be found in any grid position. The goal is to find a least-stretched occurrence of the word in the letter grid.

### *Input Format*

The first line of input contains a positive integer  $n$ , which is the number of rows and columns in the letter grid. The next  $n$  lines of input contain  $n$  lowercase letters, separated by single blanks. Every letter of the alphabet appears in at least one position of the letter grid, so at least one stretched occurrence of every word can be found. The remaining lines of input each contain a word—a nonempty string of lowercase letters.

### *Output Format*

For each word in the input, find and print a least-stretched occurrence of the word, as shown in the output sample. An occurrence of a word in the letter grid is a sequence of grid positions (lower left corner is  $(0, 0)$  and upper right corner is  $(n-1, n-1)$ ) that contain the corresponding sequence of letters in the word. Note that if the same letter appears more than once in a word, an occurrence of that word in the letter grid may use the same grid position more than once (including twice in succession). The amount of stretch in an occurrence is the sum of the distances between the grid positions (viewed as coordinates in 2-dimensional Euclidean space) of consecutive word letters. For example, the stretch of a word of length  $d$  occurring in horizontally or vertically contiguous positions is  $d-1$ , and the stretch of a word of length  $d$  occurring in diagonally contiguous positions is  $(d-1)\sqrt{2}$ . Print the amount of stretch accurate to two decimal places.

### *Input and Output Sample*

*see reverse side of this page*

### *Input Sample*

```
12
c a s d a s d f o i u b
w a d r n s a o i e r q
n z r a s d f l k j w e
a s o a u w e r n v a s
p o w b r u t b n z m s
a l e g r u t y q p a a
a s d f l k w e r c z a
s f l k w j e r a p s o
x s d f m e w r n a z z
w e r r e t u p m o c m
u l a j g h a u a u u a
a j a s f d h h w e r a
craig
flood
computer
```

### *Output Sample*

least-stretched occurrence of "craig": 13.84

```
9 5
8 5
10 8
8 10
3 6
```

least-stretched occurrence of "flood": 3.41

```
6 9
7 9
7 10
7 10
6 11
```

least-stretched occurrence of "computer": 7.00

```
10 2
9 2
8 2
7 2
6 2
5 2
4 2
3 2
```