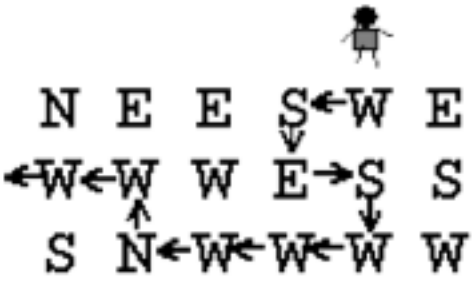
Problem 4: Robot Motion

Input file: Problem4.txt



Grid 1

Grid 2

A robot has been programmed to follow the instructions in its path. Instructions for the next direction the robot is to move are laid down in a grid. The possible instructions are

N north (up the page)
S south (down the page)
E east (to the right on the page)
W west (to the left on the page)

For example, suppose the robot starts on the north (top) side of Grid 1 and starts south (down). The path the robot follows is shown. The robot goes through 10 instructions in the grid before leaving the grid.

Compare what happens in Grid 2: the robot goes through 3 instructions only once, and then starts a loop through 8 instructions, and never exits.

You are to write a program that determines how long it takes a robot to get out of the grid or how the robot loops around.

Input:

There will be one or more grids for robots to navigate. The data for each is in the following form. On the first line are three integers separated by blanks: the number of rows in the grid, the number of columns in the grid, and the number of the column in which the robot enters from the north. The possible entry columns are numbered starting with one at the left. Then come the rows of the direction instructions. Each grid will have at least one and at most 10 rows and columns of instructions. The lines of instructions contain only the characters N, S, E, or W with no blanks. The end of input is indicated by a row containing 0 0 0.

Output

For each grid in the input there is one line of output. Either the robot follows a certain number of instructions and exits the grid on any one the four sides or else the robot follows the instructions on a certain number of locations once, and then the instructions on some number of locations repeatedly. The sample input below corresponds to the two grids above and illustrates the two forms of output. The word "step" is always immediately followed by "(s)" whether or not the number before it is 1.

Sample input:

```
3 6 5
NEESWE
WWWESS
SNWWWW
4 5 1
SESWE
EESNW
```

NWEEN
EWSEN
0 0 0

Sample output:
10 step(s) to exit
3 step(s) before a loop of 8 step(s)

Problem 5: Scrabble

Input file: Problem5.txt

The board game Scrabble (copyright Hasbro, Inc.) requires players to draw seven letter tiles from a pool, and form words on a game board, arranged horizontally or vertically, with their letters and the letters already played; just what constitutes a ``word'' is often subject to dispute, so it's handy to have a dictionary nearby when you play. Scoring is based on the point values of the letters used to form the words, with the letters associated with point values as follows:

A-1 B-3 C-3 D-2 E-1 F-4 G-2 H-4 I-1 J-8 K-5 L-1 M-3
N-1 O-1 P-3 Q-10 R-1 S-1 T-1 U-1 V-4 W-4 X-8 Y-4 Z-10

Scoring is also affected by the positions on the board where the letters are played. Some positions have the feature that they double or triple the score of the letter played there. This bonus goes only to the first player to play a tile on that position, and not to another player who later uses that letter in a word. The score for a word is obtained by summing the letter scores (including any doubling and tripling of letter values). For example, the word ``MOUSE'', which would normally score 7 points, would score 10 if played so that the ``M'' falls on one of the double letter positions. There are also positions that double or triple the score of the entire word. These bonuses can accumulate; thus, if our word ``MOUSE'' managed to land with the ``M'' on a double letter position, and the ``E'' on a double word position, then the resulting score for the word is 20. We will denote these kinds of positions as follows:

- Ordinary position
- 2 Double letter score
- 3 Triple letter score
- 4 Double word score

- 5 Triple word score

The map below illustrates the scrabble board in terms of these symbols; note the symmetries.

```
5--2---5---2--5
-4---3---3---4-
--4---2-2---4--
2--4---2---4--2
----4-----4----
-3---3---3---3-
--2---2-2---2--
5--2---4---2--5
--2---2-2---2--
-3---3---3---3-
----4-----4----
2--4---2---4--2
--4---2-2---4--
-4---3---3---4-
5--2---5---2--5
```

Finally, there is a 50 point bonus if all seven letters get played in a single turn.

The game begins with one player placing a word of up to seven letters on the board so that one letter covers that '4' (double word score) marker in the center of the board.

Now, it's your move.

Input

You will receive, in the file', a series of Scrabble puzzles, and in the file ``words'' a list of legal words. Each puzzle poses the problem of playing second, after your opponent has made the first move of the game; thus, there is just one word you will have to intersect with when you choose your word. Your opponent's first word is also in the list of legal words. In our nonstandard version of the game, we require that the word you play must use one of the letters of your opponent's word to form your own. The

objective in a puzzle is to find a word that uses some of your letters and one of the letters already played to form the highest scoring word, including bonuses for double-letter, triple-letter, double-word and triple-word positions, and the 50 point bonus for using all seven letters.

Each puzzle is represented on one line of the input file. A puzzle consists of the following information:

- The seven letters you have drawn; these are the first seven characters of the line.
- A digit from 1 to 7, indicating which letter of your opponents opening word landed on the center square.
- The word your opponent played, of up to seven letters.

You may assume that the first word is played horizontally, and thus that your word will be played vertically. The symmetry of the board actually makes it irrelevant whether the word was played horizontally or vertically.

In the file ``words'', the list of words that can be used in the game, each word is on a line by itself, at the beginning of the line, with no trailing spaces. The words in the file are contain only uppercase letters, with no digits, punctuation, or lowercase letters. You may assume that there are no more than 25000 words in the list, and that no word is more than eight letters long.

Output

You will print one line for each puzzle. In that line you will show your word, in a field 8 characters wide, left-justified, padded by spaces on the right as necessary; a digit in a 2 character field, right-justified, that shows which letter of the original word you used, a digit in a 2 character field, right-justified, that shows which letter of your word is part of your opponent's word, and finally a number in a 4 character field, right justified. that is your score for your word.

Sample Input

RDAWEER1ABDOMEN
OADPZBI5OPIATE
RTWXAIB3SMELL
AEFGOOS1BILLION
CYOKDAR7DOLTISH
CYOKDAR1DOLTISH

Sample Output

The line of digits is intended to guide you in proper output alignment, and is *not* part of the output that your solution should produce.

```
12345678901234567
WANDER    7 3  20
ZAP       4 2  27
AXIS      1 4  27
FALSE     3 3  18
DOCKYARD  1 1 100
DOCKYARD  1 8 122
```