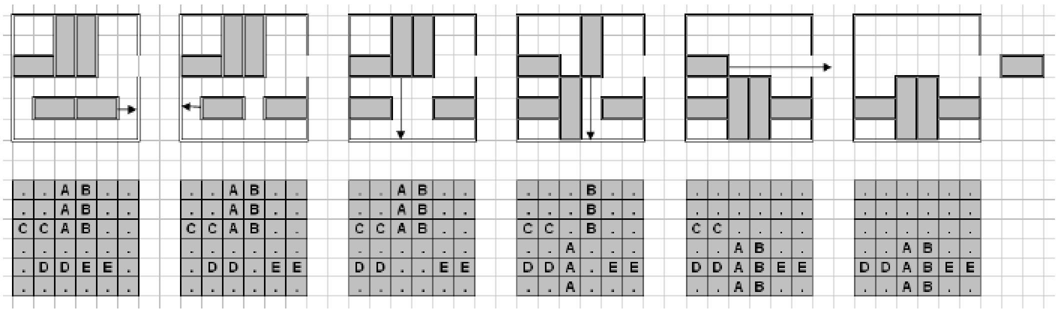
4600 Block Game

Bud bought this new board game. He is hooked. He has been playing it over and over again such that he thinks can finish the game with the minimum number of moves, but he is uncertain. He wants you to help him check whether the moves he has listed are indeed the minimum number of moves.



```
..AB..  ..AB..  ..AB..  ...B..  ......  ......
..AB..  ..AB..  ..AB..  ...B..  ......  ......
CCAB..  CCAB..  CCAB..  CC.B..  CC....  ......
......  ......  ......  ..A...  ..AB..  ..AB..
.DDEE.  .DD.EE  DD..EE  DDA.EE  DDABEE  DDABEE
......  ......  ......  ..A...  ..AB..  ..AB..
```

You are given a 6×6 board, and a set of 2×1 or 3×1 (vertical) or 1×2 or 1×3 (horizontal) pieces. You can slide the horizontal pieces horizontally only, and the vertical pieces vertically only. You are only allowed to slide a piece if there is no other piece, nor a wall, obstructing its path.

There will be one special 1×2 horizontal piece. There will also be a gap in the wall, on the right side, on the same row as the special piece, that only the special piece can fit through. The goal of the game is to get that one special horizontal piece out of the gap on the right side.

A "move" in this game is when you take a piece and slide it any number of squares (i.e. if you slide a piece horizontally one square, that is one move, and sliding it 2 squares at once is also considered one move).

Input

Input will consist of multiple datasets. Each data set will begin with a line with a single capital letter, indicating the special piece which must move off of the board.

The next 6 lines will consist of 6 characters each. These characters will either be a '.' (period), indicating an empty square, or a capital letter, indicating part of a piece.

The letters are guaranteed to form pieces that are 1×2, 1×3, 2×1 or 3×1, and no letter will be used to represent more than one piece on any given board. The letter indicating the special piece is guaranteed to correspond to a 1×2 piece somewhere on the board.

The end of data is indicated by a single '*' (asterisk) on its own line.

Output

For each data set, print a single integer, indicating the smallest number of moves necessary to remove the given piece, or '-1' if it isn't possible. Print each integer on its own line. There should be no blank lines between outputs.

Sample Input

```
C
..AB..
..AB..
```

```
CCAB..
......
.DDEE.
......
A
......
......
......
......
AA....
......
Z
.ZZ..X
.....X
.....X
.....Y
.....Y
.....Y
*
```

Sample Output

```
5
1
-1
```