DWITE '08 R2 #1 - My shiny digital key

Time limit: 2.0s **Memory limit:** 64M

DWITE Online Computer Programming Contest, November 2008, Problem 1

A house key is essentially a physical encoding of a digital key, where different heights of teeth represent different digits. If one could read this digital equivalent (let's say from a to-scale photograph), then the physical key itself is not needed to create a copy. Let's imagine that we are not lock-picking our way into someplace we shouldn't be at, but instead are running a legitimate locksmith business, maintaining a machine that cuts new copies of keys. We need to write some software that will scan the image of a key, and extract the heights of the teeth.

The input will contain 5 sets of input. Each line is 5 characters long, and there are 7 lines per set. There is an additional whitespace line separating each set. Dot `.` will be the empty space character in the "image", number sign `#` is a part of the key. Each set of `#`s is continuous, but there could be a line of input with no `#`s at all.

The output will contain 5 lines, each a 7 digit number, where each digit matches the height of the tooth on the corresponding line in the key. The top-most line of input is the left-most digit in the output. There are no spaces between the digits, and each digit is in the 0-5 range.

Note: Make sure to print any leading zeros, if such are present.

Sample Input

```
.....
#....
##...
###..
####.
#####
#....

#....
##...
###..
##...
###..
###..
##...
```

Sample Output

```
0123451
1232332
```

Problem Resource: DWITE