

Many gas stations use plastic digits on an illuminated sign to indicate prices. When there is an insufficient quantity of a particular digit, the attendant may substitute another one upside down.

The digit “6” looks much like “9” upside down. The digits “0”, “1” and “8” look like themselves. The digit “2” looks a bit like a “5” upside down (well, at least enough so that gas stations use it).

Due to rapidly increasing prices, a certain gas station has used all of its available digits to display the current price. Fortunately, this shortage of digits need not prevent the attendant from raising prices. She can simply rearrange the digits, possibly reversing some of them as described above.

Input and Output

Your job is to compute, given the current price of gas, the next highest price that can be displayed using exactly the same digits. The input consists of several lines, each containing between 2 and 30 digits (to account for future prices) and a decimal point immediately before the last digit. There are no useless leading zeroes; that is, there is a leading zero only if the price is less than 1. You are to compute the next highest price that can be displayed using the same digits and the same format rules. An input line containing a decimal point alone terminates the input. If the price cannot be raised, print ‘The price cannot be raised.’

Sample Input

```
65.2
76.7
77.7
.
```

Sample Output

```
65.5
77.6
The price cannot be raised.
```