

C. Hidden Word

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Let's define a grid to be a set of tiles with 2 rows and 13 columns. Each tile has an English letter written in it. The letters don't have to be unique: there might be two or more tiles with the same letter written on them. Here is an example of a grid:

```
ABCDEFGHIJKLM
NOPQRSTUVWXYZ
```

We say that two tiles are adjacent if they share a side or a corner. In the example grid above, the tile with the letter 'A' is adjacent only to the tiles with letters 'B', 'N', and 'O'. A tile is not adjacent to itself.

A sequence of tiles is called a path if each tile in the sequence is adjacent to the tile which follows it (except for the last tile in the sequence, which of course has no successor). In this example, "ABC" is a path, and so is "KXWIIHIJK". "MAB" is not a path because 'M' is not adjacent to 'A'. A single tile can be used more than once by a path (though the tile cannot occupy two consecutive places in the path because no tile is adjacent to itself).

You're given a string s which consists of 27 upper-case English letters. Each English letter **occurs at least once** in s . Find a grid that contains a path whose tiles, viewed in the order that the path visits them, form the string s . If there's no solution, print "Impossible" (without the quotes).

Input

The only line of the input contains the string s , consisting of 27 upper-case English letters. Each English letter occurs at least once in s .

Output

Output two lines, each consisting of 13 upper-case English characters, representing the rows of the grid. If there are multiple solutions, print any of them. If there is no solution print "Impossible".

Examples

input
ABCDEFGHIJKLMNOPSQTUVWXYZ
output
YXWVUTGHIJKLM ZABCDEF SRQPON

input
BUVTYZFQSNRIWOXXGJLKACPEMDH
output
Impossible