There are some permutation generation techniques in Knuth's book "The Art of Computer Programming - Volume 1". One of the processes is as follows:

For each permutation $A_1A_2 \ldots A_{n-1}$ form n others by inserting a character n in all possible places obtaining

$$nA_1A_2 \ldots A_{n-1}, A_1nA_2 \ldots A_{n-1}, \ldots, A_1A_2 \ldots nA_{n-1}, A_1A_2 \ldots A_{n-1}n$$

For example, from the permutation 231 inserting 4 in all possible places we get 4231 2431 2341 2314

Following this rule you have to generate all the permutation for a given set of characters. All the given characters will be different and there number will be less than 10 and they all will be alpha numerals. This process is recursive and you will have to start recursive call with the first character and keep inserting the other characters in order. The sample input and output will make this clear. Your output should exactly mach the sample output for the sample input.

Input

The input contains several lines of input. Each line will be a sequence of characters. There will be less than ten alphanumerals in each line. The input will be terminated by "End of File".

Output

For each line of input generate the permutation of those characters. The input ordering is very important for the output. That is the permutation sequence for '`abc`' and '`bca`' will not be the same.

Separate each set of permutation output with a blank line.

Sample Input

```
abc
bca
dcba
```

Sample Output

```
cba
bca
bac
cab
acb
abc

acb
cab
cba
abc
bac
bca

abcd
bacd
bcad
bcda
acbd
cabd
cbad
cbda
acdb
cadb
cdab
cdba
abdc
badc
bdac
bdca
adbc
dabc
dbac
dbca
adcb
dacb
dcab
dcba
```