A simple scheme for creating a compressed version of a text file can be used for files which contain no digit characters. The compression scheme requires making a list of the words in the uncompressed file. When a non-alphabetic character is encountered in the uncompressed file, it is copied directly into the compressed file. When a word is encountered in the uncompressed file, it is copied directly into the compressed file only if this is the first occurrence of the word. In that case, the word is put at the front of the list. If it is not the first occurrence, the word is not copied to the compressed file. Instead, its position in the list is copied into the compressed file and the word is moved to the front of the list. The numbering of list positions begins at 1.

Write a program that takes a compressed file as input and generates a reproduction of the original uncompressed file as output.

Input

You can assume that no word contains more than 50 characters and that the original uncompressed file contains no digit characters.

For the purposes of this problem, a word is defined to be a maximal sequence of upper- and lower-case letters. Words are case-sensitive — the word 'abc' is not the same as the word 'Abc'. For example,

x-ray	contains 2 words:	x and ray
Mary's	contains 2 words:	Mary and s
It's a winner	contains 4 words:	It and s and a and winner

There is no upper limit on the number of different words in the input file. The end of the input file is signified by the number '0' on a line by itself. The terminating '0' merely indicates the end of the input and should not be part of the output produced by your program.

Output

See the sample output below for details of the output format.

Sample Input

```
Dear Sally,

    Please, please do it--1 would 4
Mary very, 1 much.  And 4 6
8 everything in 5's power to make
14 pay off for you.

    -- Thank 2 18 18--
0
```

Sample Output

```
Dear Sally,

    Please, please do it--it would please
Mary very, very much.  And Mary would
do everything in Mary's power to make
it pay off for you.

    -- Thank you very much--
```