Chip and Dale have devised an encryption method to hide their messages. They first agree secretly on a number that will be used as the number of columns in a matrix. The sender prepares an intermediate format by removing capitalizations and punctuations and spaces from the message. The sender then enters the letters of the intermediate format along the diagonals of the matrix and pads with extra random letters so as to make a rectangular array of letters. For example, if the message is "There's no place like my office on a muggy day" and there are five columns, Dale would write down

```
t h r n a
e e o c k
s p e e f
l l m f o
i y i n g
o c a g a
e m y y x
u d x x x
```

Note how Dale includes only the letters and writes them all in lower case. Dale then sends the message to Chip by writing the letters in each row. So, the message in its intermediate format would be encrypted as

```
thrnaeeockspeefllmfoiyingocagaemyyxudxxx
```

In this example, Dale used the character 'x' to pad the message out to make a rectangle, although he could have used any letter.

Your job is to recover for Chip the message in its intermediate format from the encrypted one.

Input

There will be multiple input sets. Input for each set will consist of two lines. The first line will contain an integer in the range 2 . . . 20 indicating the number of columns used. The next line is a string of up to 200 lower case letters. The last input set is followed by a line containing a single zero (0). This line should not be processed.

Output

Each input set should generate one line of output, giving the message in its intermediate format followed by the padding letters.

Sample Input

```
5
thrnaeeockspeefllmfoiyingocagaemyyxudxxx
3
thsiiesaysxx
0
```

Sample Output

```
theresnoplacelikemyofficeonamuggydayxxxx
thisiseasyxx
```