

Burrows Wheeler Precompression

The **Burrows-Wheeler transform (BWT)**, also called **block-sorting compression**, is an algorithm used in data compression techniques such as bzip2. It was invented by Michael Burrows and David Wheeler.

When a character string is transformed by the BWT, none of its characters change value. The transformation permutes the order of the characters. If the original string had several substrings that occurred often, then the transformed string will have several places where a single character is repeated multiple times in a row. This is useful for compression, since it tends to be easy to compress a string that has runs of repeated characters by techniques such as move-to-front transform and run-length encoding.

For example, the string:

```
SIX.MIXED.PIXIES.SIFT.SIXTY.PIXIE.DUST.BOXES
```

could be transformed into this string, which is easier to compress because it has many repeated characters:

```
TEXYDST.E.IXIXIXSSMPPS.B..E.S.EUSFXDIIOIIIT
```

Now the Burrows-Wheeler algorithm works as follows:

- Given an input string **S**, eg: "abcba".
- Find all rotations of **S**.

eg: "abcba", "bcbaa", "cbaab", "baabc", "aabcb"

- Now sort the strings hence produced.

eg: "aabcb", "abcba", "baabc", "bcbaa", "cbaab"

- Arrange the strings in a $\text{len}(S) \times \text{len}(S)$ grid.

```
aabcb  
abcba  
baabc  
bcbaa  
cbaab
```

- Output the row number (1-based indexing) containing the original input string. Also output the strings formed by characters in the last column.

eg: 2 bacab

Now given the output of Burrows-Wheeler, can you recover the original string?

Input Format:

The input file consists of multiple testcases.

The first line of each testcase contains one integer, **R**, indicating the row number containing the original input string in the sorted matrix.

The second line of each testcase contains one string, **Col**, which is the last column of the grid. (1

$\leq \text{len}(\text{Col}) \leq 1000$)

Col contains only lowercase characters. $1 \leq R \leq \text{len}(\text{Col})$.

Input terminates with a line containing $R=0$ which must not be processed.

Output Format:

Print the original input string to the burrow wheeler's algorithm.

Testdata:

30 testcases

Sample Input:

```
2
bacab
3
rwlb
11
baaabaaaabbbaba
0
```

Sample Output:

```
abcba
rbwl
baaabbbbaaaaaab
```