# 632   Compression (II)

The fast growth of program sizes and amount of stored data raises the need for efficient compression algorithms. They can significantly reduce the amount of storage space. Also the "Internet boom", still suppressed by slow links has big influence on new developments in this field.

One of the most modern methods is block sorting. Its first step performed on the data is Burrows-Wheeler transform (BWT). The idea, used by the authors, comes from well-established in image compression Fourier transform and it says: transform the data to the form which is easier to compress and then compress it using one of the well-known algorithms.

Let's assume that as input we have a character string $S$ of length $N$. The first step of the transform is to generate from string $S$, $N$ strings $S_0$, $S_1$, ..., $S_{N-1}$ in the following way:

- string $S_0$ is identical to $S$

- string $S_k$ is generated from string $S_{k-1}$ by moving the first character to the end.

The example that illustrates this algorithm is shown below:

$$S = \text{PASCAL}$$

| $S_0$ | P | A | S | C | A | L |
|---|---|---|---|---|---|---|
| $S_1$ | A | S | C | A | L | P |
| $S_2$ | S | C | A | L | P | A |
| $S_3$ | C | A | L | P | A | S |
| $S_4$ | A | L | P | A | S | C |
| $S_5$ | L | P | A | S | C | A |

In the second step all the strings are sorted to give:

| $S_4$ | A | L | P | A | S | C |
|---|---|---|---|---|---|---|
| $S_1$ | A | S | C | A | L | P |
| $S_3$ | C | A | L | P | A | S |
| $S_5$ | L | P | A | S | C | A |
| $S_0$ | P | A | S | C | A | L |
| $S_2$ | S | C | A | L | P | A |

The results of the transform, which are used in further work are:

- last column of just created array (CPSALA in this case)

- number of the row (starting from 0) in which $S_1$ is placed

It appears that such data are easier to perform an inverse transform and easier to compress by other methods than original input data.

Your task is to write a program, which performs above-described BWT transform on input string.

## Input

The first line of the input is an integer $M$, then a blank line followed by $M$ datasets. There is a blank line between datasets.

In the first line comes an integer $N < 1997$ which stands for string length. Subsequent lines contain a string to transform. Each line (besides the last one) contains exactly 50 characters.

## Output

For each dataset, there should be position of $S_1$ string in the first line. Subsequent lines should contain a string to transform. Each line (besides the last one) should contain exactly 50 characters. Print a blank line between datasets.

## Sample Input

```
1

6
pascal
```

## Sample Output

```
1
cpsala
```