

Problem

You are given a binary string S of length N . You can perform the following operation on it:

- Pick any non-empty even-length [subsequence](#) of the string.
- Suppose the subsequence has length $2k$. Then, move the first k characters to the beginning of S and the other k to the end of S (without changing their order).

For example, suppose $S = 01100101110$. Here are some moves you can make (the chosen subsequence is marked in red):

- $01100101110 \rightarrow 10101011100$
- $01100101110 \rightarrow 10010111001$
- $01100101110 \rightarrow 00100111110$

What is the lexicographically smallest string that can be obtained after performing this operation several (possibly, zero) times?

Note: A binary string A is said to be lexicographically smaller than another binary string B of the same length if there exists an index i such that:

- $A_1 = B_1, A_2 = B_2, \dots, A_{i-1} = B_{i-1}$
- $A_i = 0$ and $B_i = 1$.

Input Format

- The first line of input will contain a single integer T , denoting the number of test cases. The description of the test cases follows.
- Each test case consists of two lines of input.
 - The first line of each test case contains a single integer N , the length of string S .
 - The second line contains a binary string of length N .

Output Format

For each testcase, output a single line containing the lexicographically shortest binary string that can be obtained by performing the given operation several times.

Constraints

Problem

You are given a binary string S of length N . You can perform the following operation on it:

- Pick any non-empty even-length [subsequence](#) of the string.
- Suppose the subsequence has length $2k$. Then, move the first k characters to the beginning of S and the other k to the end of S (without changing their order).

For example, suppose $S = 01100101110$. Here are some moves you can make (the chosen subsequence is marked in red):

- $01100101110 \rightarrow 10101011100$
- $01100101110 \rightarrow 10010111001$
- $01100101110 \rightarrow 00100111110$

What is the lexicographically smallest string that can be obtained after performing this operation several (possibly, zero) times?

Note: A binary string A is said to be lexicographically smaller than another binary string B of the same length if there exists an index i such that:

- $A_1 = B_1, A_2 = B_2, \dots, A_{i-1} = B_{i-1}$
- $A_i = 0$ and $B_i = 1$.

Input Format

- The first line of input will contain a single integer T , denoting the number of test cases. The description of the test cases follows.
- Each test case consists of two lines of input.
 - The first line of each test case contains a single integer N , the length of string S .
 - The second line contains a binary string of length N .

Output Format

For each testcase, output a single line containing the lexicographically shortest binary string that can be obtained by performing the given operation several times.

Constraints