A. String Similarity

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

A binary string is a string where each character is either `0` or `1`. Two binary strings a and b of equal length are *similar*, if they have the same character in some position (there exists an integer i such that $a_i = b_i$). For example:

- `10010` and `01111` are *similar* (they have the same character in position 4);
- `10010` and `11111` are *similar*;
- `111` and `111` are *similar*;
- `0110` and `1001` are not *similar*.

You are given an integer n and a binary string s consisting of $2n - 1$ characters. Let's denote $s[l..r]$ as the contiguous substring of s starting with l-th character and ending with r-th character (in other words, $s[l..r] = s_l s_{l+1} s_{l+2} \ldots s_r$).

You have to construct a binary string w of length n which is *similar* to **all of the following strings**: $s[1..n]$, $s[2..n+1]$, $s[3..n+2]$, ..., $s[n..2n-1]$.

Input

The first line contains a single integer t ($1 \le t \le 1000$) — the number of test cases.

The first line of each test case contains a single integer n ($1 \le n \le 50$).

The second line of each test case contains the binary string s of length $2n - 1$. Each character s_i is either `0` or `1`.

Output

For each test case, print the corresponding binary string w of length n. If there are multiple such strings — print any of them. It can be shown that at least one string w meeting the constraints always exists.

Example

input
4 1 1 3 00000 4 1110000 2 101
output
1 000 1010 00

Note

The explanation of the sample case (equal characters in equal positions are bold):

The first test case:

- **1** is similar to $s[1..1] =$ **1**.

The second test case:

- **000** is similar to $s[1..3] =$ **000**;
- **000** is similar to $s[2..4] =$ **000**;
- **000** is similar to $s[3..5] =$ **000**.

The third test case:

- **1**0**10** is similar to $s[1..4] =$ **1**1**10**;
- **1**0**1**0 is similar to $s[2..5] =$ **1**1**0**0;
- **10**1**0** is similar to $s[3..6] =$ **10**0**0**;
- 1**0**1**0** is similar to $s[4..7] =$ 0**0**0**0**.

The fourth test case:

- 0**0** is similar to $s[1..2] =$ 1**0**;

- $\mathbf{0}0$ is similar to $s[2..3] = \mathbf{0}1$.