

Problem J. Binary String To Subsequences

Time limit 2000 ms

Mem limit 262144 kB

You are given a binary string s consisting of n zeros and ones.

Your task is to divide the given string into the **minimum** number of **subsequences** in such a way that *each character* of the string belongs to exactly *one subsequence* and each subsequence looks like "010101 ..." or "101010 ..." (i.e. the subsequence should not contain two adjacent zeros or ones).

Recall that a subsequence is a sequence that can be derived from the given sequence by deleting zero or more elements without changing the order of the remaining elements. For example, subsequences of "1011101" are "0", "1", "11111", "0111", "101", "1001", but not "000", "101010" and "11100".

You have to answer t independent test cases.

Input

The first line of the input contains one integer t ($1 \leq t \leq 2 \cdot 10^4$) — the number of test cases. Then t test cases follow.

The first line of the test case contains one integer n ($1 \leq n \leq 2 \cdot 10^5$) — the length of s . The second line of the test case contains n characters '0' and '1' — the string s .

It is guaranteed that the sum of n does not exceed $2 \cdot 10^5$ ($\sum n \leq 2 \cdot 10^5$).

Output

For each test case, print the answer: in the first line print one integer k ($1 \leq k \leq n$) — the minimum number of subsequences you can divide the string s to. In the second line print n integers a_1, a_2, \dots, a_n ($1 \leq a_i \leq k$), where a_i is the number of subsequence the i -th character of s belongs to.

If there are several answers, you can print any.

Sample 1

Input	Output
-------	--------

Input	Output
4	2
4	1 2 2 1
0011	6
6	1 2 3 4 5 6
111111	1
5	1 1 1 1 1
10101	4
8	1 1 1 1 1 2 3 4
01010000	