Problem A. 进行位翻转

Time limit 1000 ms
Mem limit 262144 kB

You are given a binary string of length n. You have **exactly** k moves. In one move, you must select a single bit. The state of all bits **except** that bit will get flipped (0 becomes 1, 1 becomes 0). You need to output the lexicographically largest string that you can get after using **all** k moves. Also, output the number of times you will select each bit. If there are multiple ways to do this, you may output any of them.

A binary string a is lexicographically larger than a binary string b of the same length, if and only if the following holds:

- in the first position where a and b differ, the string a contains a 1, and the string b contains a 0.

Input

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

Each test case has two lines. The first line has two integers n and k ($1 \leq n \leq 2 \cdot 10^5$; $0 \leq k \leq 10^9$).

The second line has a binary string of length n, each character is either 0 or 1.

The sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, output two lines.

The first line should contain the lexicographically largest string you can obtain.

The second line should contain n integers $f_1, f_2, \ldots, f_n$, where f_i is the number of times the i-th bit is selected. The sum of all the integers **must be equal to** k.

Sample 1

Input	Output
6 6 3 100001 6 4 100011 6 0 000000 6 1 111001 6 11 101100 6 12 001110	111110 1 0 0 2 0 0 111110 0 1 1 1 0 1 000000 0 0 0 0 0 0 100110 1 0 0 0 0 0 111111 1 2 1 3 0 4 111110 1 1 4 2 0 4

Note

Here is the explanation for the first testcase. Each step shows how the binary string changes in a move.

- Choose bit 1: 100001 → 111110.
- Choose bit 4: 111110 → 000101.
- Choose bit 4: 000101 → 111110.

The final string is 111110 and this is the lexicographically largest string we can get.