LED Strip

Stofl bought himself a LED strip that can be managed via microcontroller. According to the official product specification he can toggle any lamp by sending its index over I^{2}C. As it turns out this almost works but the led strip has a very unfortunate bug: Not only the i-th lamp is being toggled but also the $i + 1$-th lamp. Some more research revealed that the developers of the LED strip could not agree on whether to use 0-based or 1-based indices so they decided to support *both*.

Trying to make the best out of the situation, Stofl decided that he doesn't care about the exact configuration of lights, but on the number of "groups"; a group is a non-empty sequence of 1s surrounded by two 0s (or the end of the strip).

Given a starting configuration of n lamps, print out some indices that can be toggled to have at least g groups visible using at most t toggle operations. It is guaranteed that this is possible for all inputs.

Input

The first line contains three integers n (number of LEDs), g (minimal number of groups) and t (maximal number of operations).

On the second line a bitstring (i.e. only consisting of 0 and 1) of length n follows. This represents the starting configuration of the lights. A 1 at position i indicates that the i-th LED is powered at the start and a 0 indicates no power.

Output

Print an integer m on the first line, the number of toggle operations. On the following m lines print a number i so that the i-th and $i + 1$-th light are switched. The actions will be performed in the order of the output.

Limits

There are 4 test groups, each of which is worth 25 points.

- In test group 1: $1 \leq n \leq 10^3, 0 \leq g \leq n/3, t = n$
- In test group 2: $1 \leq n \leq 10^3, 0 \leq g \leq n/3, t = n/3$
- In test group 3: $1 \leq n \leq 10^6, 0 \leq g \leq n/3, t = n$
- In test group 4: $1 \leq n \leq 10^6, 0 \leq g \leq n/3, t = n/3$

Examples

Input	Output

`3 1 1`
`000`
	1
	1

The strip looks afterwards as follows: `110`

Input	Output

`14 4 5`
`00011100011111`
	`2`
	`1`
	`11`

The strip looks afterwards as follows: `11011100010011`