Problem H. Shared Memory Switch

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

Your task is to build an optimal algorithm for the shared memory switch.

A shared memory switch is one of the simplest nontrivial buffering architectures considered in the field of algorithms for networking. In this problem we consider a shared memory switch with multiple output queues and uniform (identical) packets. Incoming packets in this model are destined to one of the several output queues, which share a common buffer of finite size B. At the end of each second all nonempty queues transmit one packet each. When the buffer overflows your algorithm must decide which packet to drop. Your goal is to design an algorithm achieving maximal throughput (equivalently, dropping as few packets as possible).

You will be given n queries in the following format:

- -1: a single second passes.
- k $(1 \le k \le n)$: a packet arrives to the k-th queue.

You can assume that all packets remaining in the buffer after all queries will be sent as well or, equivalently, that there is an infinite number of -1 queries that follow the queries in the input.

Input

The first line contains contains two non-negative integers n and B $(n, B \le 2 \cdot 10^5)$: the number of queries and the amount of memory available to the switch correspondingly.

The second line contains n queries in the format described above.

Output

On the first line, output the only integer: the number of transmitted packets.

On the second line, output the numbers of queries (1-based) with arrivals of the packets that were transmitted. You can output these numbers in any order.

Examples

standard input	*standard output*

`14 5`
`1 1 1 2 2 2 -1 1 -1 1 -1 1 -1 1`
	`9`
	`3 6 8 5 10 4 12 14 2`

`14 4`
`1 1 1 2 2 2 -1 1 -1 1 -1 1 -1 1`
	`8`
	`3 6 8 5 10 4 12 14`