

## Problem A. New Year Train

Input file:            a.in  
Output file:           a.out  
Time limit:            2 seconds  
Memory limit:         128 megabytes

On the New Year Eve a government of one country decided to send a train with gifts to each town of the country. For each of  $N$  towns exactly one wagon with gifts was sent. The route was organized in such way that at each place the last wagon would be detached, then in the next place the other last one and so on. Just before the departure it turned out that the loading workers did not pay attention to numeration of the wagons and loaded the gifts into the wagons in random order. It was impossible to detach a wagon from the middle of the train and there was no time to rearrange gifts.

Luckily, there was a depot with parallel tracks. At the entrance of the depot each wagon could be directed to any of the tracks, and wagons could leave the depot from the other side in the right sequences: 1, 2, 3, 4, and so on.

For example, if at the entrance of the depot with three parallel tracks there were six wagons standing in order: 2, 5, 1, 4, 6, 3, then wagons 2, 5, 6 could be directed to the first track; wagons 1, 4 to the second track, wagon 3 to the third track could be directed respectively. In this case wagons could leave the depot in the right order.

Fortunately, there were enough tracks in the depot to rearrange the train.

### Input

The first line of input file contains two integers  $N$  and  $M$  — the number of wagons in the train and the number of tracks in the depot respectively ( $1 \leq N \leq 800\,000$ ,  $1 \leq M \leq 100\,000$ ,  $M \leq N$ ). The second line contains  $N$  integers — sequence of wagons before the entrance to the depot.

It's guaranteed that solution always exists.

### Output

On the first line output file must contain  $N$  integers — which track should be chosen for each wagon from input data. On the second line of output file write the number of tracks in order the wagons should leave the depot in the sequence 1, 2, 3, and so on.

### Examples

a.in	a.out
6 3	1 1 2 2 1 3
2 5 1 4 6 3	2 1 3 2 1 1

### Note

In 30% of testcases  $N \leq 100$ .

In 60% of testcases  $N \leq 10\,000$ .