CENTRAL-EUROPEAN OLYMPIAD IN INFORMATICS

Sárospatak, Hungary
28 July - 4 August 2005

Multi-key Sorting

Input file: `keys.in`	**100 points**
Output file: `keys.out`	**Time limit: 6 sec**
Source Code: `keys.pas/.c/.cpp`	**Memory limit:10 MB**

Consider a table with rows and columns. The columns are numbered from ***1*** to ***C***. For simplicity's sake, the items in the table are strings consisting of lower case letters.

Col. 1	Col. 2	Col. 3
apple	red	sweet
apple	green	sour
pear	green	sweet
banana	yellow	sweet
banana	brown	rotten

Table 1

Col. 1	Col. 2	Col. 3
banana	brown	rotten
apple	green	sour
pear	green	sweet
apple	red	sweet
banana	yellow	sweet

Table 2

Col. 1	Col. 2	Col. 3
apple	green	sour
apple	red	sweet
banana	brown	rotten
banana	yellow	sweet
pear	green	sweet

Table 3

You are given the operation Sort(k) on such tables: Sort(k) sorts the rows of a table in the order of the values in column ***k*** (while the order of the columns does not change). The sort is stable, that is, rows that have equal values in column ***k***, remain in their original order. For example, applying Sort(2) to Table 1 yields Table 2.

We are interested in sequences of such sort operations. These operations are successively applied to the same table. For example, applying the sequence Sort(2); Sort(1) to Table 1 yields Table 3.

Two sequences of sort operations are called equivalent if, for any table, they have the same effect. For example, Sort(2); Sort(2); Sort(1) is equivalent to Sort(2); Sort(1). However, it is not equivalent to Sort(1); Sort(2), because the effect on Table 1 is different.

Task

Given a sequence of sort operations, determine a shortest equivalent sequence.

Input

The first line of the text file `keys.in` contains two integers, ***C*** and ***N***. ***C*** ($1 \le C \le 1\ 000\ 000$) is the number of columns and ***N*** ($1 \le N \le 3\ 000\ 000$) is the number of sort operations. The second line contains ***N*** integers, k_i ($1 \le k_i \le C$). It defines the sequence of sort operations Sort(k_1); ...; Sort(k_N).

Output

The first line of the text file `keys.out` contains one integer, ***M***, the length of the shortest sequence of sort operations equivalent to the input sequence (Subtask A). The second line contains exactly ***M*** integers, representing a shortest sequence (Subtask B). You can omit the second line if you solve only Subtask A.

Example

keys.in	keys.out
4 6 1 2 1 2 3 3	3 1 2 3