

Restore Permutation

Input file: **standard input**
Output file: **standard output**
Time limit: 2 seconds
Memory limit: 256 megabytes

This is a run-twice problem. Your solution will be executed twice.

During the first run you are given p — a permutation of integers from 1 to n . Your program should output a binary string (a string of zeros and ones) of length not exceeding m . The value of m is unknown to your program and depends on the subtask, the corresponding value is indicated in the scoring system table. If your program outputs a string longer than m , it will receive a verdict of “Wrong Answer”.

Between the runs the jury program will swap two different elements of the original permutation, obtaining permutation q .

During the second run your program will be given permutation q and the binary string output by you in the first run as input. You are required to restore the original permutation.

Input

During the first run, the first input line contains the number 1 and an integer n — the length of the permutation ($2 \leq n \leq 10^6$). The second line contains n distinct integers p_i separated by a space — the elements of the permutation ($1 \leq p_i \leq n$).

During the second run, the first and second lines in the same format contain the number 2, the length of the permutation n , and the permutation q , and the third line contains a binary string of zeros (symbol ‘0’) and ones (symbol ‘1’) of length not exceeding m .

Output

During the first run, output a binary string of length not exceeding m , which will then be used to restore the permutation during the second run.

During the second run, output n distinct integers from 1 to n separated by a space — the elements of the original permutation p .

Interaction Protocol

To avoid receiving incorrect verdicts such as “Idleness Limit Exceeded” or “Security Violation”, terminate the output of each line with a newline character (‘\n’).

Scoring

Points for each subtask are awarded only if all tests of this subtask and the required subtasks are successfully passed.

Subtask	Points	Constraints	Required Subtasks
0	–	examples from the statement, $m = 2000$	
1	7	$n \leq 16, m = 2000$	0
2	15	$n \leq 10^5, m = 2 \cdot 10^6$	
3	13	$n \leq 2000, m = 200$	0, 1
4	11	$m = 32\,000$	2
5	17	$m = 7000$	2, 4
6	10	$m = 2000$	0 – 2, 4, 5
7	19	$m = 500$	0 – 2, 4 – 6
8	18	$m = 200$	0 – 7

Examples

standard input	standard output	Notes
1 4 4 1 2 3	100001010011	Первый запуск
2 4 4 3 2 1 100001010011	4 1 2 3	Второй запуск

standard input	standard output	Notes
1 5 2 3 4 5 1	111100100111010100011111100110 00100010011011110001111111010 011100100	Первый запуск
2 5 2 3 5 4 1 111100100111010100011111100110 00100010011011110001111111010 011100100	2 3 4 5 1	Второй запуск

Note

In the examples long binary strings are displayed with line breaks for correct display in the statement. In reality, there are no such line breaks. Also note that the input and output during the second launch depend on the output during the first launch and may not match the examples shown in the statement.