Codeforces Round #446 (Div. 2)

A. Greed

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Jafar has n cans of cola. Each can is described by two integers: remaining volume of cola a_i and can's capacity b_i ($a_i \le b_i$).

Jafar has decided to pour all remaining cola into just 2 cans, determine if he can do this or not!

Input

The first line of the input contains one integer n ($2 \le n \le 100\,000$) — number of cola cans.

The second line contains n space-separated integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 10^9$) — volume of remaining cola in cans.

The third line contains n space-separated integers that $b_1, b_2, \ldots, b_n$ ($a_i \le b_i \le 10^9$) — capacities of the cans.

Output

Print "YES" (without quotes) if it is possible to pour all remaining cola in 2 cans. Otherwise print "NO" (without quotes).

You can print each letter in any case (upper or lower).

Examples

input
2 3 5 3 6
output
YES

input
3 6 8 9 6 10 12
output
NO

input
5 0 0 5 0 0 1 1 8 10 5
output
YES

input
4 4 1 0 3 5 2 2 3
output
YES

Note

In the first sample, there are already 2 cans, so the answer is "YES".

B. Wrath

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Hands that shed innocent blood!

There are n guilty people in a line, the i-th of them holds a claw with length L_i. The bell rings and every person kills some of people in front of him. All people kill others at the same time. Namely, the i-th person kills the j-th person if and only if $j < i$ and $j \geq i - L_i$.

You are given lengths of the claws. You need to find the total number of alive people after the bell rings.

Input

The first line contains one integer n ($1 \leq n \leq 10^6$) — the number of guilty people.

Second line contains n space-separated integers $L_1, L_2, \ldots, L_n$ ($0 \leq L_i \leq 10^9$), where L_i is the length of the i-th person's claw.

Output

Print one integer — the total number of alive people after the bell rings.

Examples

input
4 0 1 0 10
output
1

input
2 0 0
output
2

input
10 1 1 3 0 0 0 2 1 0 3
output
3

Note

In first sample the last person kills everyone in front of him.

C. Pride

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You have an array a with length n, you can perform operations. Each operation is like this: choose two **adjacent** elements from a, say x and y, and replace one of them with $gcd(x, y)$, where gcd denotes the greatest common divisor.

What is the minimum number of operations you need to make all of the elements equal to 1?

Input

The first line of the input contains one integer n ($1 \le n \le 2000$) — the number of elements in the array.

The second line contains n space separated integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$) — the elements of the array.

Output

Print -1, if it is impossible to turn all numbers to 1. Otherwise, print the minimum number of operations needed to make all numbers equal to 1.

Examples

input
5 2 2 3 4 6
output
5

input
4 2 4 6 8
output
-1

input
3 2 6 9
output
4

Note

In the first sample you can turn all numbers to 1 using the following 5 moves:

- [2, 2, 3, 4, 6].
- [2, 1, 3, 4, 6]
- [2, 1, 3, 1, 6]
- [2, 1, 1, 1, 6]
- [1, 1, 1, 1, 6]
- [1, 1, 1, 1, 1]

We can prove that in this case it is not possible to make all numbers one using less than 5 moves.

D. Gluttony

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given an array a with n distinct integers. Construct an array b by permuting a such that for every non-empty subset of indices $S = \{x_1, x_2, \ldots, x_k\}$ $(1 \le x_i \le n, 0 < k < n)$ the sums of elements on that positions in a and b are different, i. e.

$$\sum_{i=1}^{k} a_{x_i} \neq \sum_{i=1}^{k} b_{x_i}.$$

Input

The first line contains one integer n $(1 \le n \le 22)$ — the size of the array.

The second line contains n space-separated distinct integers $a_1, a_2, \ldots, a_n$ $(0 \le a_i \le 10^9)$ — the elements of the array.

Output

If there is no such array b, print `-1`.

Otherwise in the only line print n space-separated integers $b_1, b_2, \ldots, b_n$. Note that b must be a permutation of a.

If there are multiple answers, print any of them.

Examples

input
2 1 2
output
2 1

input
4 1000 100 10 1
output
100 1 1000 10

Note

An array x is a permutation of y, if we can shuffle elements of y such that it will coincide with x.

Note that the empty subset and the subset containing all indices are not counted.

E. Envy

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

For a connected undirected weighted graph G, MST (minimum spanning tree) is a subgraph of G that contains all of G's vertices, is a tree, and sum of its edges is minimum possible.

You are given a graph G. If you run a MST algorithm on graph it would give you only one MST and it causes other edges to become jealous. You are given some queries, each query contains a set of edges of graph G, and you should determine whether there is a MST containing all these edges or not.

Input

The first line contains two integers n, m ($2 \le n$, $m \le 5 \cdot 10^5$, $n - 1 \le m$) — the number of vertices and edges in the graph and the number of queries.

The i-th of the next m lines contains three integers u_i, v_i, w_i ($u_i \neq v_i$, $1 \le w_i \le 5 \cdot 10^5$) — the endpoints and weight of the i-th edge. There can be more than one edges between two vertices. It's guaranteed that the given graph is connected.

The next line contains a single integer q ($1 \le q \le 5 \cdot 10^5$) — the number of queries.

q lines follow, the i-th of them contains the i-th query. It starts with an integer k_i ($1 \le k_i \le n - 1$) — the size of edges subset and continues with k_i distinct space-separated integers from 1 to m — the indices of the edges. It is guaranteed that the sum of k_i for $1 \le i \le q$ does not exceed $5 \cdot 10^5$.

Output

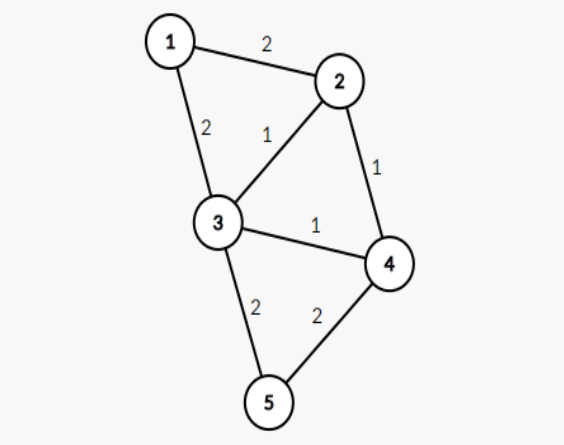
For each query you should print "YES" (without quotes) if there's a MST containing these edges and "NO" (of course without quotes again) otherwise.

Example

input
5 7 1 2 2 1 3 2 2 3 1 2 4 1 3 4 1 3 5 2 4 5 2 4 2 3 4 3 3 4 5 2 1 7 2 1 2
output
YES NO YES NO

Note

This is the graph of sample:



Weight of minimum spanning tree on this graph is 6.

MST with edges $(1, 3, 4, 6)$, contains all of edges from the first query, so answer on the first query is "YES".

Edges from the second query form a cycle of length 3, so there is no spanning tree including these three edges. Thus, answer is "NO".