A. 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 ($1 \leq n \leq 2000$) — the number of elements in the array.

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

B. 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.

C. 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 ($u_i \ne v_i$, $1 \le w_i \le 5 \cdot 10^5$) — 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

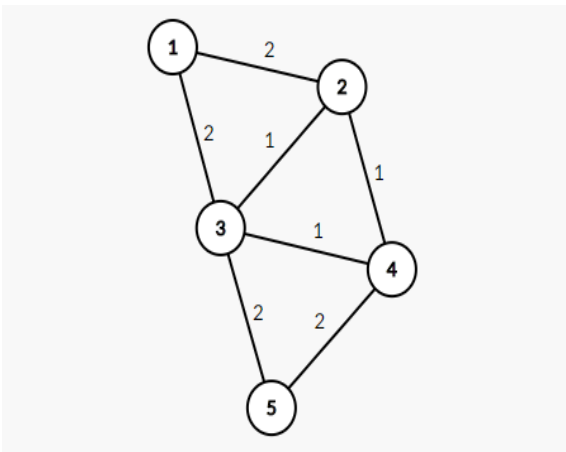
```
5 7
1 2 2
1 3 2
2 3 1
2 4 1
3 4 1
3 5 2
4 5 2
4
2 3 4
3 3 4 5
2 1 7
2 1 2
```

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`".

D. Sloth

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

Sloth is bad, mkay? So we decided to prepare a problem to punish lazy guys.

You are given a tree, you should count the number of ways to remove an edge from it and then add an edge to it such that the final graph is a tree and has a perfect matching. Two ways of this operation are considered different if their removed edges or their added edges aren't the same. The removed edge and the added edge can be equal.

A perfect matching is a subset of edges such that each vertex is an endpoint of exactly one of these edges.

Input

The first line contains n $(2 \le n \le 5 \cdot 10^5)$ — the number of vertices.

Each of the next $n - 1$ lines contains two integers a and b $(1 \le a, b \le n)$ — the endpoints of one edge. It's guaranteed that the graph is a tree.

Output

Output a single integer — the answer to the problem.

Examples

input

```
4
1 2
2 3
3 4
```

output

```
8
```

input

```
5
1 2
2 3
3 4
3 5
```

output

```
0
```

input

```
8
1 2
2 3
3 4
1 5
5 6
6 7
1 8
```

output

```
22
```

Note

In first sample, there are 8 ways:

- edge between 2 and 3 turns to edge between 1 and 3,
- edge between 2 and 3 turns to edge between 1 and 4,
- edge between 2 and 3 turns to edge between 2 and 3,
- edge between 2 and 3 turns to edge between 2 and 4,
- edge between 1 and 2 turns to edge between 1 and 2,
- edge between 1 and 2 turns to edge between 1 and 4,
- edge between 3 and 4 turns to edge between 1 and 4,
- edge between 3 and 4 turns to edge between 3 and 4.

E. Lust

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

A false witness that speaketh lies!

You are given a sequence containing n integers. There is a variable res that is equal to 0 initially. The following process repeats k times.

Choose an index from 1 to n uniformly at random. Name it x. Add to res the multiply of all a_i's such that $1 \le i \le n$, but $i \ne x$. Then, subtract a_x by 1.

You have to find expected value of res at the end of the process. It can be proved that the expected value of res can be represented as an irreducible fraction $\frac{P}{Q}$. You have to find $P \cdot Q^{-1} \bmod 1000000007$.

Input

The first line contains two integers n and k ($1 \le n \le 5000$, $1 \le k \le 10^9$) — the number of elements and parameter k that is specified in the statement.

The second line contains n space separated integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 10^9$).

Output

Output a single integer — the value $P \cdot Q^{-1} \bmod 1000000007$.

Examples

input
```
2 1
5 5
```
output
```
5
```

input
```
1 10
80
```
output
```
10
```

input
```
2 2
0 0
```
output
```
500000003
```

input
```
9 4
0 11 12 9 20 7 8 18 2
```
output
```
169316356
```