A. Am I Your Ancestor?

time limit per test: 1 second

memory limit per test: 256 megabytes

You are given a rooted tree $T = (V, E)$ with $V = \{1, 2, \dots, n\}$. The vertex 1 is the root of T. You are then given Q queries. Each query consists of two vertices $u, v \in V$ and the goal is to judge whether u is an ancestor of v. Note that a vertex is considered to be an ancestor of itself.

Input

The first line contains two positive integers n $(1 \leq n \leq 10^5)$ and Q $(1 \leq Q \leq 10^5)$. Each of the following $n - 1$ lines contains two integers u and v $(1 \leq u, v \leq n)$, meaning that there is an edge between u and v in T. Each of the following Q lines contains two integers u and v $(1 \leq u, v \leq n)$ which corresponds to a query.

Output

For each query, display YES if u is ancestor of v, otherwise display NO.

Example

input

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

output

```
YES
YES
NO
YES
NO
```

B. Nearest Node

time limit per test: 2 seconds

memory limit per test: 256 megabytes

You are given an undirected and unweighted tree $T = (V, E)$ with $n = |V|$ vertices and Q queries. For each query, you are given three vertices $a, b, c \in V$, and the goal is to find a vertex $v \in V$ to minimize $\mathrm{dist}(a, v) + \mathrm{dist}(b, v) + \mathrm{dist}(c, v)$. Here for two vertices u and v, $\mathrm{dist}(u, v)$ is the distance between u and v on T.

Input

The first line contains two positive integer n $(1 \le n \le 10^5)$ and Q $(1 \le Q \le 10^5)$. Each of the following $n - 1$ lines contains two integers u and v $(1 \le u, v \le n)$, meaning that there is an edge between u and v in T. Each of the following Q lines contains three integers a, b and c $(1 \le a, b, c \le n)$, which corresponds to a query.

Output

For each query, display a single integer which is the minimum value of $\mathrm{dist}(a, v) + \mathrm{dist}(b, v) + \mathrm{dist}(c, v)$.

Example

input

```
6 4
1 2
2 3
2 4
4 5
5 6
4 5 6
6 3 1
2 4 4
6 6 6
```

output

```
2
5
1
0
```

C. Strip

time limit per test: 1 second
memory limit per test: 256 megabytes

Alexandra has a paper strip with n numbers on it. Let's call them a_i from left to right.

Now Alexandra wants to split it into some pieces (possibly 1). For each piece of strip, it must satisfy:

- Each piece should contain at least l numbers.
- The difference between the maximal and the minimal number on the piece should be at most s.

Please help Alexandra to find the minimal number of pieces meeting the condition above.

Input

The first line contains three space-separated integers n, s, l ($1 \le n \le 10^5$, $0 \le s \le 10^9$, $1 \le l \le 10^5$).

The second line contains n integers a_i separated by spaces ($-10^9 \le a_i \le 10^9$).

Output

Output the minimal number of strip pieces.

If there are no ways to split the strip, output -1.

Examples

input
```
7 2 2
1 3 1 2 4 1 2
```
output
```
3
```

input
```
7 2 2
1 100 1 100 1 100 1
```
output
```
-1
```

Note

For the first sample, we can split the strip into 3 pieces: $[1, 3, 1], [2, 4], [1, 2]$.

For the second sample, we can't let 1 and 100 be on the same piece, so no solution exists.

D. Serious Subsequences

time limit per test: 2 seconds

memory limit per test: 256 megabytes

You are given a sequence of integers $a = (a_1, a_2, \ldots, a_n)$. We say a subsequence is serious if its length is at least L and at most R. I.e., a subsequence $(a_l, a_{l+1}, \ldots, a_r)$ is serious if and only if $r - l + 1 \in [L, R]$. For a serious subsequence $(a_l, a_{l+1}, \ldots, a_r)$, its weight is defined to be $\sum_{i=l}^{r} a_i$.

Choose k different serious subsequences and maximize their total weight.

Input

The first line contains four integers n ($1 \le n \le 500000$), k ($1 \le k \le 500000$), L and R ($1 \le L \le R \le n$). Each of the following n lines contains a single integer. The integer in the i-th line is a_i ($-1000 \le a_i \le 1000$).

Output

Display a single integer which is the maximum total weight k different serious subsequences.

Examples

input

```
4 3 2 3
3
2
-6
8
```

output

```
11
```

input

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

output

```
-6
```

E. Bottleneck Path

time limit per test: 0.5 seconds
memory limit per test: 256 megabytes

You are given a weighted undirected graph $G = (V, E)$ with $n = |V|$ vertices and $m = |E|$ edges, together with Q queries. For each query, you are given two vertices u and v, and the goal is to find the maximum edge weight W such that there exists a path from u to v that does not use any edges of weight less than W.

You might recall that such a path is called a bottleneck path from u to v, and W is the weight of the minimum weight edge on a bottleneck path.

Input

The first line contains two integers n $(1 \leq n \leq 10000)$ and m $(1 \leq m \leq 50000)$. Each of the following m lines contains three integers u $(1 \leq u \leq n)$, v $(1 \leq v \leq n)$ and w $(1 \leq w \leq 100000)$, meaning that there is an edge between u and v with weight w. The next line contains a single integer Q $(1 \leq Q \leq 30000)$. Each of the following Q lines contains two integers u $(1 \leq u \leq n)$ and v $(1 \leq v \leq n)$, which are the queries.

Output

For each query, display the minimum edge weight on a bottleneck path from u to v. If there does not exist any path from u to v, output -1.

Example

input

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

output

```
3
-1
3
```

F. Duff in the Army

time limit per test: 4 seconds

memory limit per test: 512 megabytes

Recently Duff has been a soldier in the army. Malek is her commander.

Their country, Andarz Gu has n cities (numbered from 1 to n) and $n - 1$ bidirectional roads. Each road connects two different cities. There exist a unique path between any two cities.

There are also m people living in Andarz Gu (numbered from 1 to m). Each person has and ID number. ID number of $i - th$ person is i and he/she lives in city number c_i. Note that there may be more than one person in a city, also there may be no people living in the city.

Malek loves to order. That's why he asks Duff to answer to q queries. In each query, he gives her numbers v, u and a.

To answer a query:

Assume there are x people living in the cities lying on the path from city v to city u. Assume these people's IDs are $p_1, p_2, ..., p_x$ in increasing order.

If $k = min(x, a)$, then Duff should tell Malek numbers $k, p_1, p_2, ..., p_k$ in this order. In the other words, Malek wants to know a minimums on that path (or less, if there are less than a people).

Duff is very busy at the moment, so she asked you to help her and answer the queries.

Input

The first line of input contains three integers, n, m and q $(1 \le n, m, q \le 10^5)$.

The next $n - 1$ lines contain the roads. Each line contains two integers v and u, endpoints of a road $(1 \le v, u \le n, v \ne u)$.

Next line contains m integers $c_1, c_2, ..., c_m$ separated by spaces $(1 \le c_i \le n$ for each $1 \le i \le m)$.

Next q lines contain the queries. Each of them contains three integers, v, u and a $(1 \le v, u \le n$ and $1 \le a \le 10)$.

Output

For each query, print numbers $k, p_1, p_2, ..., p_k$ separated by spaces in one line.

Examples

input

```
5 4 5
1 3
1 2
1 4
4 5
2 1 4 3
4 5 6
1 5 2
5 5 10
2 3 3
5 3 1
```

output

```
1 3
2 2 3
0
3 1 2 4
1 2
```

Note

Graph of Andarz Gu in the sample case is as follows (ID of people in each city are written next to them):