Three Nodes

Time limit: 1.0s **Memory limit:** 512M

You are given a tree with N nodes, numbered from 1 to N, connected by $n-1$ edges. Each edge connects two nodes with the length of 1. You're also given Q queries. Each query contains three non-negative integers x, y, z.

For each query, you need to find three nodes in the tree, denoted as a, b, and c, such that $\text{dist}(a,b)=x$, $\text{dist}(a,c)=y$, $\text{dist}(b,c)=z$, where $\text{dist}(u,v)$ is the length of the simple path from node u to node v. Specially, $\text{dist}(u,u)=0$.

It is guaranteed that there exists at least one solution for each query. If there are multiple solutions, output any valid solution.

Input Specification

The first line contains one integer N ($1 \le N \le 2 \times 10^5$), representing the number of nodes in the tree.

Each of the following $N-1$ lines contain two integers u and v ($1 \le u, v \le N$), indicating an edge between nodes u and v in the tree.

The nexct line contains one integer Q ($1 \le Q \le 2 \times 10^5$), representing the number of queries.

Each of the next Q lines contains three integers x, y, and z ($0 \le x, y, z \le N-1$), representing a query.

Output Specification

For each query, output three integers separated by space: a, b, and c, representing the nodes a, b, and c that satisfy the given conditions. There may be multiple valid solutions for each query. You can output any valid solution.

Constraints

The test cases in this problem are not batched.

Points	Additional constraints
10%	$N \le 500$, $Q \le 500$
20%	$N \le 2\,000$, $Q \le 2\,000$
20%	$Q = 1$
20%	$Q \le 10$
10%	There exists an edge between node i and $i+1$.
10%	$x = 0$ for all queries.
10%	No additional constraints

Sample Input

```
10
7 10
2 8
10 2
8 1
9 7
4 5
1 6
9 4
4 3
10
3 2 1
5 4 1
6 6 0
3 0 3
1 5 4
2 5 7
6 5 1
2 1 3
2 0 2
2 2 0
```

Sample Output

```
2 6 1
7 6 1
9 6 6
6 2 6
6 1 7
8 6 4
9 6 1
1 2 6
6 8 6
8 6 6
```