# Birjik and Nicole's Tree Game

Nicole and Birjik are seniors taking a break from job applications by creating the following game:

- Birjik draws a *rooted tree* with n vertices numbered from 1 to n, rooted at vertex 1.

- Nicole creates q queries. For each query, she takes a copy of the tree and colors k of its vertices *black*, leaving the remaining $n - k$ vertices *white*.

- The goal of the game is to answer each query by finding the respective values of $c_0, c_1, c_2, \ldots, c_k$, where each c_i is the number of subtrees containing exactly i *black* vertices (including the subtree's root vertex).

For example, the diagram below depicts a query on a tree where $k = 3$ and the *black* vertices are 2, 5, and 4:
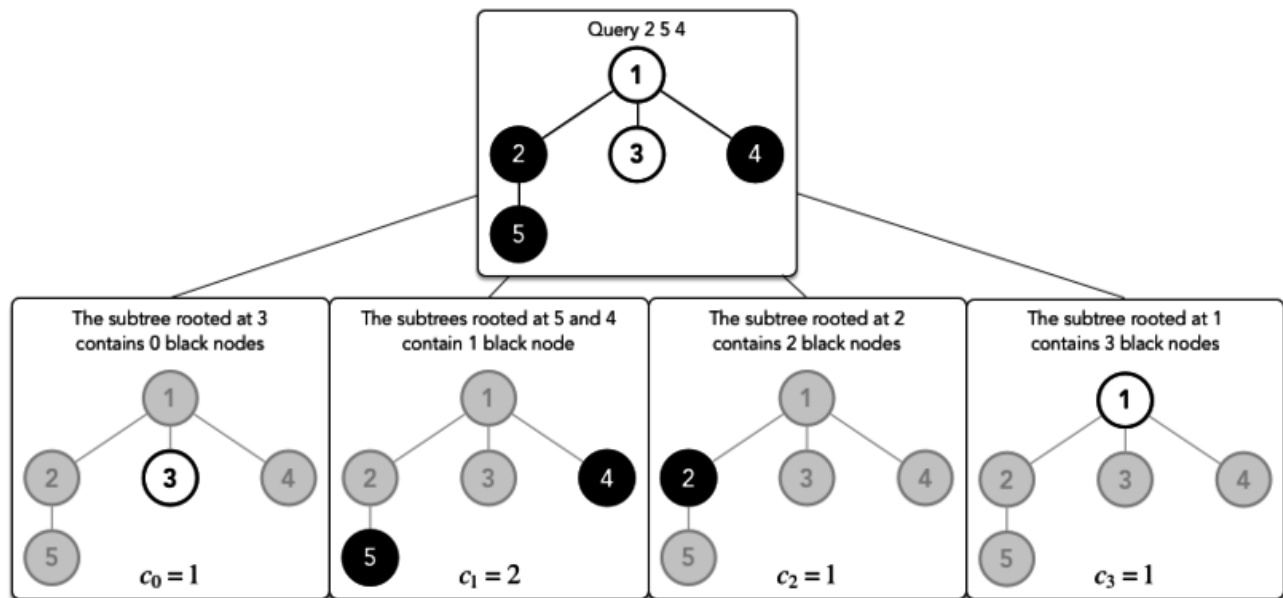
Given the tree and q queries, solve each query by printing the values of $c_0, c_1, c_2, \ldots, c_k$ on a new line.

Input Format

The first line contains an integer, n, denoting the number of vertices of the tree.
Each of the $n - 1$ subsequent lines contains two space-separated integers, u and v, describing an edge connecting vertices u and v.
The next line contains an integer, q, denoting the number of queries. The $2 \cdot q$ subsequent lines describe each query over two lines:

1. The first line contains an integer denoting k.

2. The second line contains k space-separated integers describing the respective IDs of the vertices to color *black*.

Constraints

- $1 \le n, q, k \le 3 \times 10^5$

- $1 \le u, v, k \le n$

- It is guaranteed that the given graph is a tree.

- It is guaranteed that the k vertex IDs given in each query are distinct IDs that exist in the tree.

- The sum of k over all queries in a test case is $\le 3 \times 10^5$

Output Format

For each query, print a single line containing $k + 1$ integers describing the respective values of $c_0, c_1, \ldots, c_k$. Recall that each c_i is the total number of subtrees containing exactly i *black* vertices.
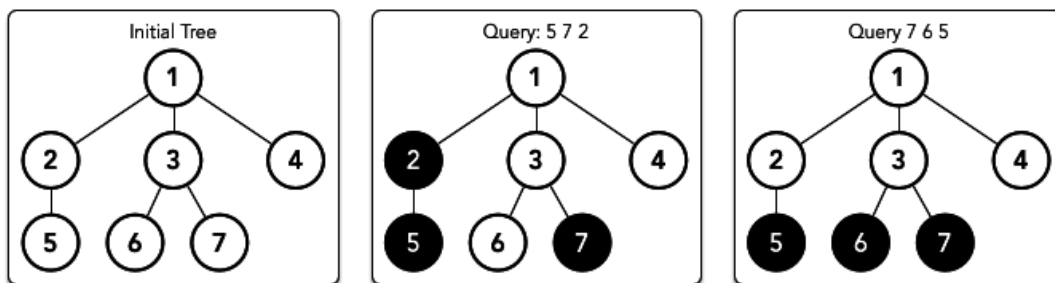
Sample Input 0

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

Sample Output 0

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

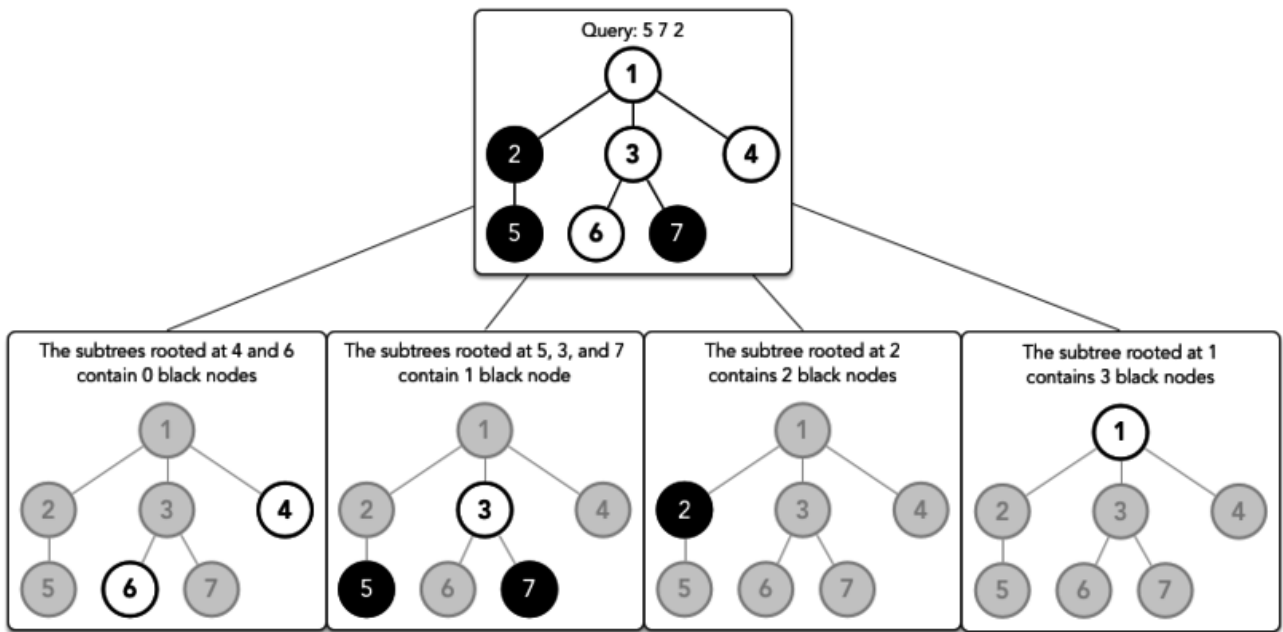**Explanation 0**

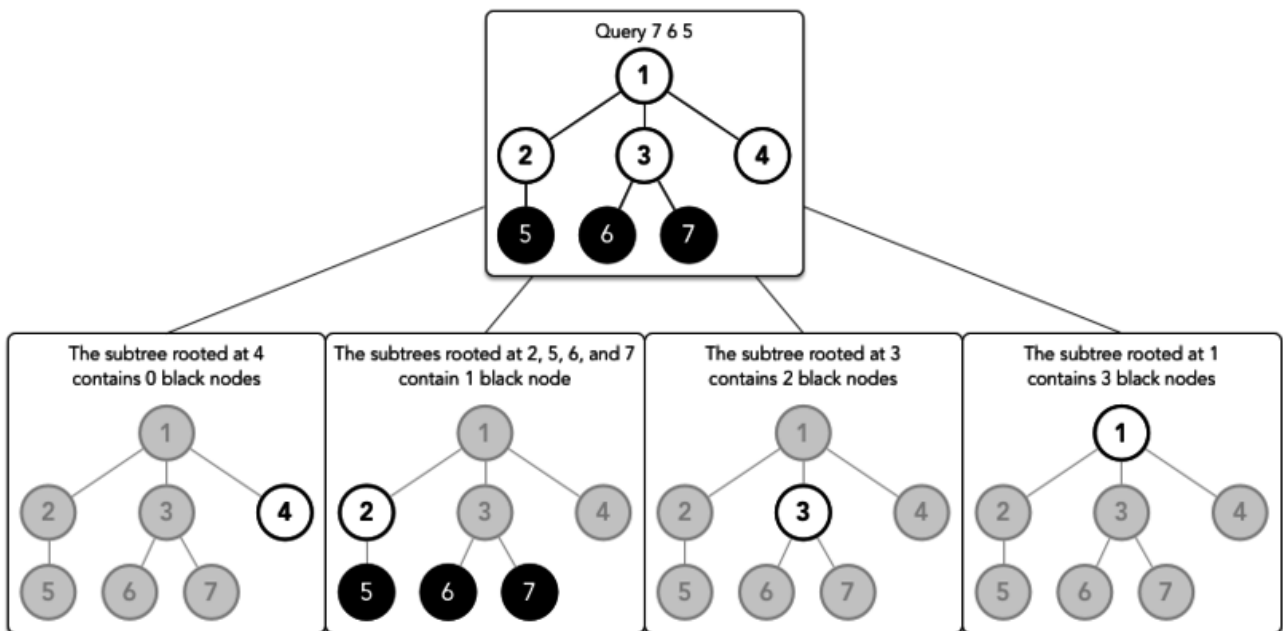In this example, the graph and queries look like this:

We perform the following $q = 2$ queries:

1. Color vertices 5, 7, and 2 black:

Query: 5 7 2

| The subtrees rooted at 4 and 6 contain 0 black nodes | The subtrees rooted at 5, 3, and 7 contain 1 black node | The subtree rooted at 2 contains 2 black nodes | The subtree rooted at 1 contains 3 black nodes |

We then print the respective values of $c_0, \ldots, c_{k=3}$ as 2 3 1 1 on a new line.

2. Color vertices 7, 6, and 5 black:

Query 7 6 5

| The subtree rooted at 4 contains 0 black nodes | The subtrees rooted at 2, 5, 6, and 7 contain 1 black node | The subtree rooted at 3 contains 2 black nodes | The subtree rooted at 1 contains 3 black nodes |

We then print the respective values of $c_0, \ldots, c_{k=3}$ as 1 4 1 1 on a new line.

Sample Input 1
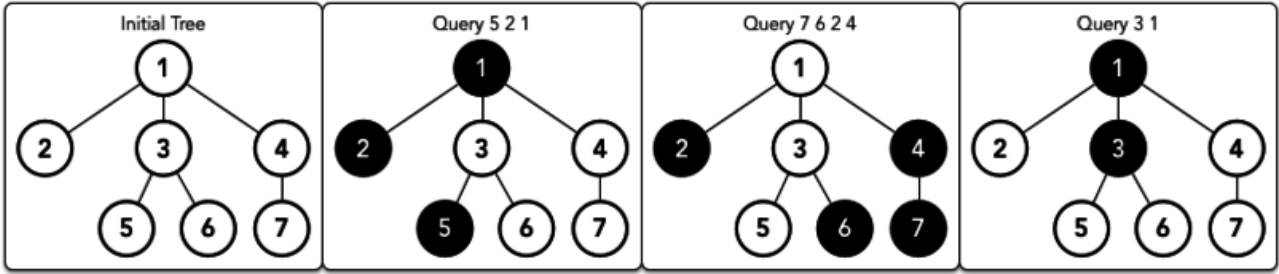
```
7
2 1
1 3
4 1
7 4
3 5
3 6
3
3
5 2 1
4
7 6 2 4
```

```
2
3 1
```

Sample Output 1

```
3 3 0 1
1 4 1 0 1
5 1 1
```

Explanation 1

In this example, the graph and queries look like this:

Follow the same process as *Sample Case 0* to verify the *Expected Output* values.