

SAC '22 Code Challenge 3 P4 - Finding Friends

Time limit: 0.5s **Memory limit:** 256M
Java: 2.0s

SAC has N students with $N - 1$ friendships between two friends (note: a friendship is bidirectional).

Recently, Mr. DeMello was tasked with monitoring friends and the friend-distance between them. The friend-distance is the minimum amount of friendships required to travel between two friends.

However, Mr. DeMello discovered that friendships could be destroyed, and all the current friendships would be destroyed eventually.

The school administrators ask Mr. DeMello Q queries of 2 types:

1 u v : Output the friend-distance between friend u and friend v or **-1** if they are not connected by friends.

2 u v : Remove the friendship between friend u and friend v .

Since Mr. DeMello wants to get a raise, he has decided to ask you to kindly help him.

Can you do it?

Constraints

For all subtasks:

$$1 \leq u, v, s, t \leq N$$

$$u \neq v$$

$$s \neq t$$

Every friend can reach every other friend before any friendships are destroyed.

Every friendship will be deleted once in the queries.

Subtask 1 [30%]

$$2 \leq N \leq 500$$

$$N \leq Q \leq 1\,000$$

Subtask 2 [70%]

$$2 \leq N \leq 100\,000$$

$$N \leq Q \leq 200\,000$$

Input Specification

The first line will contain N and Q , the number of students and queries, respectively.

The next $N - 1$ lines will contain s and t , a friendship between student s and student t .

The next Q lines will contain one of the above 2 queries, a query for the friend-distance between student u and student v or a query to destroy a friendship between u and v .

Output Specification

For each type 1 query, output the friend-distance or `-1` if they cannot reach each other through friends.

Sample Input

```
5 8
1 2
3 2
4 5
4 1
1 1 3
2 2 3
1 1 3
2 4 1
1 5 3
1 1 2
2 1 2
2 4 5
```

Sample Output

```
2
-1
-1
1
```