CCC '09 S3 - Degrees Of Separation

Time Limit: 1.0s   **Memory Limit:** 64M

Canadian Computing Competition: 2009 Stage 1, Junior #5, Senior #3

The main socializing tool for students today is Facebook. There are many interesting computational questions connected to Facebook, such as the "degree of separation" between two people.

For example, in the diagram below, there are many different paths between Abby and Alberto. Some of these paths are:

- Abby → Zoey → Alberto
- Abby → Natalie → Zoey → Alberto
- Abby → George → Ali → Kara → Richardo → Jeff → Alberto

The shortest path between Abby and Alberto has two steps (Abby → Zoey, and Zoey → Alberto), so we say the degree of separation is 2. Additionally, Alberto would be a friend of a friend of Abby.
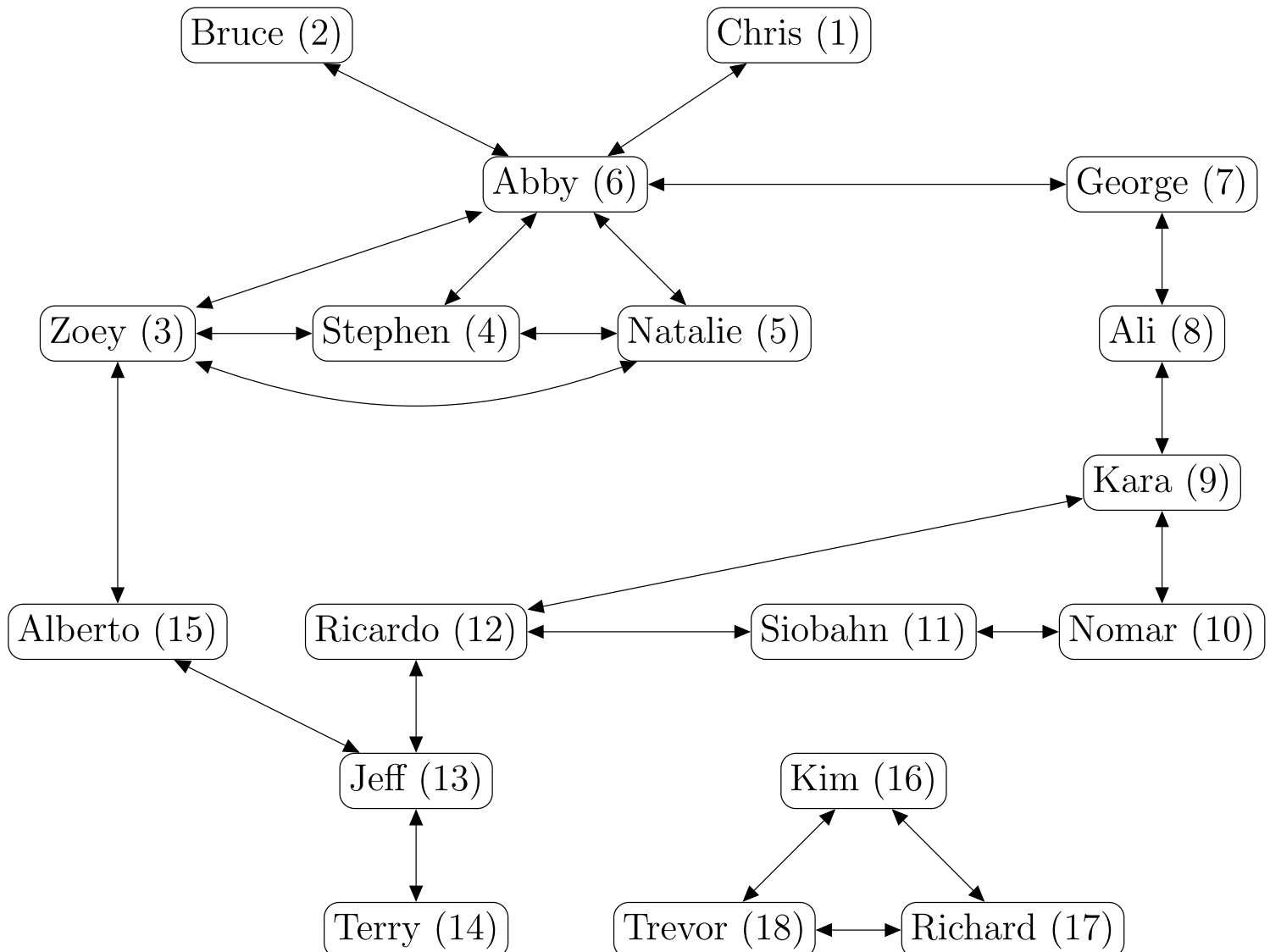
You can assume an initial configuration of who is friends with who as outlined in the diagram above. You will need to store these relationships in your program. These relationships can change though, and your program needs to handle

these changes. In particular, friendships can begin, possibly with new people. Friendships can end. You should be able to find friends of friends and determine the degree of separation between two people.

Input/Output Description

Your program will read in six possible commands, with the action to be performed by your program outlined below. You may assume that x and y are integers, with $x \neq y$, $x \geq 1$, $y \geq 1$, $x < 50$ and $y < 50$. You may also assume that instructions (i, d, n, f, s, q) occur one per line and parameters (zero, one or two integers) occur one per line.

- `i` x y - make person x and person y friends. If they are already friends, no change needs to be made. If either x or y is a new person, add them.
- `d` x y - delete the friendship between person x and person y.
- `n` x - output the number of friends that person x has.
- `f` x - output the number of "friends of friends" that person x has. Notice that x and direct friends of x are not counted as "friends of friends."
- `s` x y - output the degree of separation between x and y. If there is no path from x to y, output "Not connected".
- `q` - quit the program.

Sample Input

```
i
20
10
i
20
9
n
20
f
20
s
20
6
q
```

Sample Output

```
2
3
4
```

Explanation

- `n 20` : Person 20 has two friends (10 and 9)
- `f 20` : The friends of friends of 20 are 8, 11, 12.
- `s 20 6` : The shortest path is 20 → 9 → 8 → 7 → 6.