

Developer Exam

Notes: Please complete the test in JavaScript. Using Node is preferable but not a must. You are not required to have a html page for this task. Outputs can be in the form of console logs.
You are creating an application where users can follow other users, for example like Twitter.
The product owner has requested a feature to show the closest link of followers between the users who is logged in and another user being searched for.

INPUT

Input Data must be read by the program from the file itself (not copy pasted into the Javascript code)

User datafile

Attached is a complete input user data: **data.json** that will be used as Input #1.

Brief explanation in Input 1:

```
[
  {
    "follows":
      [10, 1, 16, 11, 14, 9, 4, 5, 3, 17, 6, 15, 12, 7, 13]
  },
  {
    "follows":
      [ 19, 18 ]
  }
  . . .
]
```

In the excerpt above, a user is represented by an ID from 0 to n . The ID of the user is the index of the array. Hence, user ID 0 follows users of ID 10, 1, 16, and so on. User ID 1 is following users of ID 18, and 19.

Test file

Another input file, **input.json**, the content is in the list of scenarios below:

```
[
  { "from" : 1, "to" : 7 },
  { "from" : 2, "to" : 3 },
  . . .
]
```

where "from" is the user logged in, and "to" is the target user. input.json will be your Input #2.

In the excerpt above, at 1st scenario, we should be looking for the closest link of followers for user ID 1 to user ID 7, next scenario, the closest link from user ID 2 to user ID 3, and so on.
Complete list is in the Input #2 file.

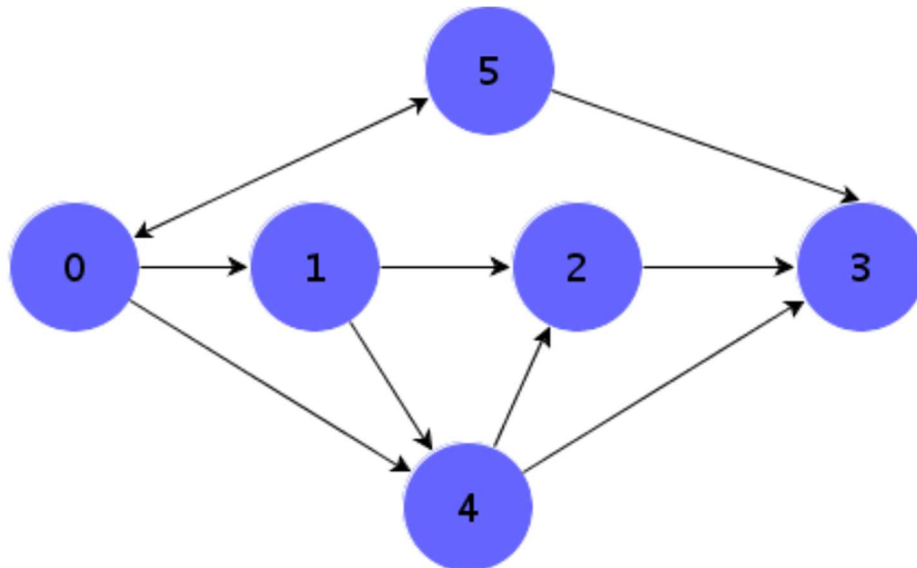
Expected Output

Given the two input files as above, the result should be like this:

Scenario	Shortest Link
Case 1 to 7:	1 -> 19 -> 7
Case 2 to 3:	2 -> 3
Case 3 to 9:	3 -> 2 -> 9
Case 4 to 0:	4 -> 0
Case 5 to 6:	5 -> 19 -> 6
...	

More explanation with example scenarios below:

The exam above can be further explained through a visual representation example below:



In this example, if a search scenario requested from '0' to '3', the program should output '0 4 3'. If there are more than one possibilities with the same degree, show only one possibility with the smallest ID number. In the example above, '0 5 3' has the same degree of link with '0 4 3', however the program should output '0 4 3' because '4 < 5'.

If there are no links found from Logged in user to Target User, display "Link not found"

Important note: There is no maximum degree from one user to another user, the closest link may be between any number of other users, e.g. 1 -> 2-> 3-> 4-> 5-> 6-> 7