

## CMPUT 275 - Tangible Computing

### Interview Problem: Rivers

---

#### Description

Rivers are very interesting! Small rivers flow and join into larger rivers which then join into even larger rivers. This continues until eventually all rivers flow into the ocean. You are studying such a system of rivers and want to start by answering one basic type of query: given two rivers, at what point do they eventually connect?

We say two rivers  $u$  and  $v$  **connect** at river  $k$  if  $k$  is the first river that lies on both sequences of rivers connecting  $u$  and  $v$  to the ocean. In particular, if  $u = v$  then we simply say they connect at  $u$ .

Given a list of  $n$  rivers and a series of queries of this form, can you tell us when the two queried rivers connect?

#### Input

The input consists of multiple lines, the first contains two integers,  $1 \leq n \leq 100,000$ , the number of rivers, and  $1 \leq q \leq 1,000$ , the number of queries.

The next line contains  $n$  space separated integers indexed  $1, \dots, n$  with each  $0 \leq a_j \leq j-1$ , telling us that river  $j$  connects to river  $a_j$  (with river 0 actually being the ocean).

Then follow  $q$  lines, each with two space separated integers  $u$  and  $v$  with  $0 \leq u, v \leq n$ , describing two rivers.

#### Output

Output will contain  $q$  lines, one for each query. Each line will have a single integer  $k$  where rivers  $u$  and  $v$  connect.

#### Target Running Time

Each query must be solved in  $O(n)$  time for full credit. There is a very natural algorithm that can solve a query with this running time that does not rely on the standard library. Though, you are free to use the standard library if you wish.

#### Additional Requirements

You must solve a query within a function with the following signature:

```
int query(int* a, int n, int u, int v);
```

Here,  $a$  points to the start of the array of values  $a[]$  from the description,  $n$  is the number of rivers, and  $u, v$  are the two query values from a single query. The function should **return** the river where  $u, v$  connect. You should change the parameter names to something more descriptive, but their types must be exactly as given.

All `cin` and `cout` statements must be made within the `main` function. You may rely on additional helper functions if you want, but keep all input/output processing in `main()`.

#### Sample Input 1

```
3 2
0 0 1
1 3
1 2
```

### Sample Output 1

```
1
0
```

### Explanation

Given the input we have  $a_1 = 0$ ,  $a_2 = 0$ ,  $a_3 = 1$ .

To answer the query 1 3 we see that river 3 connects directly to river 1 so the river that  $u$  and  $v$  connect at is river 1.

To answer the query 1 2 we see that river 2 connects directly to the ocean and also river 1 connects directly to the ocean, so the river that  $u$  and  $v$  connect at is river 0, the ocean.

### Sample Input 2

```
7 1
0 1 2 0 1 5 6
3 7
```

### Sample Output 2

```
1
```

**Explanation:** Given the input we have  $a_1 = 0$ ,  $a_2 = 1$ ,  $a_3 = 2$ ,  $a_4 = 0$ ,  $a_5 = 1$ ,  $a_6 = 5$ ,  $a_7 = 6$ .

To answer the query 3 7 we see that river 3 connects to the ocean in the order 3 -> 2 -> 1 -> 0 and that river 7 connects to the ocean in the order 7 -> 6 -> 5 -> 1 -> 0. Both 3 and 7 flow to rivers 1 and 0, the first river they both flow to is 1, therefore they connect at 1.

### Grading Comments

Despite the fact this appears similar to a morning problem, it will be graded like a weekly exercise. In particular:

- Style matters. Use appropriate comments, proper indentation, etc. Include a file header. Consult the style guide on eClass.
- You must adhere exactly to the output specification: for example, if you output in the wrong order or print extra whitespaces then you will receive a deduction. The test centre must accept the output without any presentation error.

- You were only give a few test cases in the test centre files on eClass. We will test your solution on additional test cases that adhere to the input specification.
- Partial credit may be obtained if your solution works on some inputs but not all inputs in the described range.
- Adhere closely to the submission instructions for the weekly exercise. See the eClass code submission link for details.