Hello, **2024101067**.

Lowest Common Ancestor

Submit solution

My submissions
All submissions
Best submissions

Points: 100 (partial)
Time limit: 1.0s
Memory limit: 256M

Allowed languages
C

Problem Statement

Given a tree represented by an array of integers where each element represents the parent of a corresponding node, find the **Lowest Common Ancestor (LCA)** of two given nodes `a` and `b`.

The **Lowest Common Ancestor** of two nodes `a` and `b` in a tree is the deepest node that has both `a` and `b` as descendants. A node is considered a descendant of itself.

You are given an array `parent` of size `n`, where `parent[i]` represents the parent of node `i`. If `parent[i] = -1`, node `i` is the root of the tree. It is guaranteed that the tree is connected, meaning there is a path between any two nodes, and that there are no self-loops, i.e., for every node `i`, the parent of `i` is not `i`.

Input Format

- The first line contains an integer `n`, the number of nodes in the tree.
- The second line contains two integers `a` and `b`, the two nodes for which the LCA is to be found.
- The third line contains `n` space-separated integers, where the `i-th` integer represents the parent of the `i-th` node. If the integer is `-1`, the `i-th` node is the root node.

Output Format

- Print the LCA of nodes `a` and `b`. If no LCA is found (which should not happen in a valid tree), print "No LCA found (invalid tree or nodes)".

- $1 \le n \le 10^4$
- $1 \le a, b \le n$
- $-1 \le parent[i] \le n$ and $parent[i] \ne 0$

Example

Input

```
7
4 7
-1 1 1 2 2 3 3
```

Copy

Output

```
1
```

Copy

Clarifications

Request clarification

No clarifications have been made at this time.