Homework Assignment #10: Programming Exercise #2

Due Time/Date

1:20PM Tuesday, December 10, 2024. Late submission will be penalized by 20% for each working day overdue.

Problem Description

Solve Problem G "Kindergarten" of the 2024 World Finals of the International Collegiate Programming Contest (see `https://scoreboard.icpc.global/2024/problemset.pdf` or the attachment).

Important Notes

This assignment constitutes 4% of your grade. You may discuss the problem with others, but copying code is strictly forbidden. **Some of you may be requested to demonstrate your program.**

Submission Guidelines

- Pack everything, excluding compiler-generated files, in a .zip file, named with the pattern "`b127050xx-hw10.zip`".
- Upload the .zip file to the NTU COOL site for Algorithms 2024.
- If you use a Makefile, make sure that it outputs "`hw10`". Otherwise, make sure that the whole application can be compiled by a single command like "`gcc hw10.c`", "`g++ hw10.cpp`", "`javac hw10.java`", etc.

Grading

Your work will be graded according to its correctness, efficiency, and presentation. Before submission, you should have tested your program on several input cases. You should organize and document your program in such a way that other programmers, for example your classmates, can understand it. In the documentation of your program (preferably in the code as comments), you are encouraged to describe how you have applied the algorithmic techniques, in particular design by induction, learned in class.

Below is a more specific grading policy:

Criteria	Score
incomplete or doesn't compile	≤ 20
complete, compiles, but with major errors	≤ 40
complete, compiles, but with minor errors	≤ 70
correct (passing all test cases)	≤ 90
correct and reasonably efficient (at least around the class-average)	≤ 100
well-organized and with helpful code comments	$+ \leq 10$

Problem G
Kindergarten
Time limit: 2 seconds

Generated by ChatGPT 4o

Taking a group of kindergarten kids to the planetarium isn't easy. You really wanted to do this, to allow every kid a chance to get into the room with the giant telescope and take a look at Jupiter. And now that you're going, you remember the stories from other caretakers that kids can misbehave and leave some nasty surprises in the telescope room for the kids after them. You really want to avoid that.

You know the kids in your group very well. Each kid is jealous of one other kid, who is cooler than them, and they might misbehave in the telescope room if they know the kid they're jealous of will be there at some point after them—not necessarily immediately after them, just at some later point. You thought this would be easy—the coolest kid in the class doesn't have anyone to be jealous of, so she can go first, and then all the other kids in order of coolness. However, you just learned that the coolest kid is an exception—instead of being jealous of someone cooler than herself, she's jealous of some other random kid in the group. This sounds like a disaster!

Fortunately, you also know that each kid has some other kid they really, really like. So whenever a kid in the telescope room is thinking about setting up a surprise, if they know the kid they like is going to be in the room after them and before the kid they're jealous of, they will refrain from misbehaving. To make this formal, if a kid A is jealous of kid B, and really likes kid C, then there's a risk A will misbehave and set up a surprise in the telescope room if B will be in the telescope room after A, and C will be there either before A or after B.

Can you figure out an order in which the kids can go to the telescope room so no surprises occur?

Input

The first line contains an integer n ($3 \leq n \leq 200\,000$), the number of kids in your group. The kids are indexed from 1 to n in decreasing order of coolness. Each of the next n lines describes one of the kids. The i^{th} of these lines contains two integers: j_i, the index of the kid that the i^{th} kid is jealous of, and l_i, the index of the kid that the i^{th} kid really, really likes ($1 \leq j_i, l_i \leq n$, $j_i \neq l_i$, $j_i \neq i$, $l_i \neq i$, and $j_i < i$ for all i except 1).

Output

Output a line containing n integers, the order in which the kids should enter the telescope room. If there are multiple ways to order the children, output any of them. If no order exists output `impossible`.

Sample Input 1	Sample Output 1
4 4 2 1 4 2 4 2 1	1 2 3 4

Sample Input 2	Sample Output 2
4 2 3 1 4 2 1 1 2	impossible