Problem A
A-Skew-ed Reasoning
Time limit: 2 seconds

The following is based on a true story – the names have been changed because. . . well, because you always change names in stories like this one.

Professor Taylor Swift is grading a homework assignment on integer skew heaps. A skew heap is a binary tree with an integer stored in each node such that the value in any node is less than or equal to the values in any of its children. Note that the skew heap need not be a perfect binary tree; that is, the left and/or right subtree of any node may be empty.

Inserting a value x into a skew heap H is done using the following recursive procedure:

- If H is empty, make H a skew heap consisting of a single node containing x.
- Otherwise, let y be the value in the root of H.
  - If $y < x$, swap the two children of the root and recursively insert x into the new left subtree.
  - If $y \geq x$, create a new node with value x and make H the left subtree of this node.

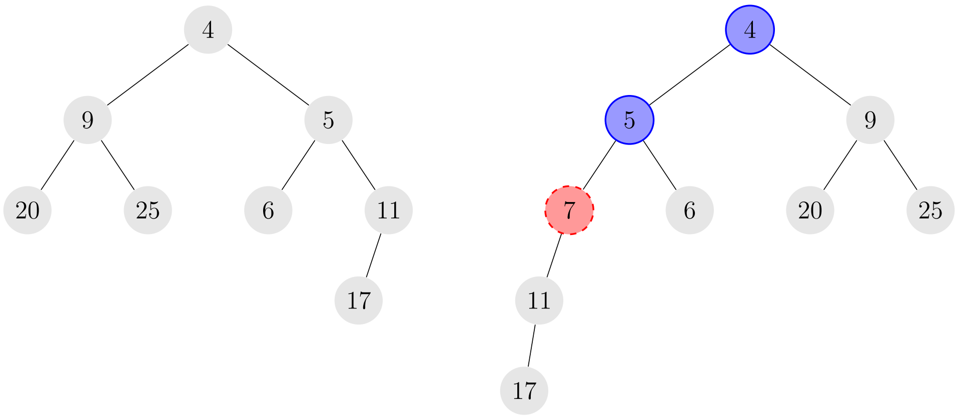

Figure A.1: Example of inserting the value 7 into a skew heap. The nodes storing 4 and 5 (marked in blue) have their children swapped, while the node storing 11 becomes the left child of the newly inserted node (marked in red).

Now, back to Professor Swift. The homework problem she has assigned asks the students to show the heap that results from inserting a given permutation of the numbers from 1 to n, in the given order, into an empty heap. Surprisingly, some of the students have wrong answers! That got Professor Swift wondering: For a given heap, is there an input permutation that would have produced this heap? And if so, what are the lexicographically minimal and maximal such input permutations?

Input

The first line of input contains an integer n ($1 \leq n \leq 2 \cdot 10^5$), the number of nodes in the tree. These nodes contain the numbers from 1 to n exactly. This is followed by n lines, the i^{th} of which contains two integers ℓ_i and r_i ($i < \ell_i \leq n$ or $\ell_i = 0$; $i < r_i \leq n$ or $r_i = 0$), describing the values of the left and right children of the node storing i, where a value of 0 is used to indicate that the corresponding child does not exist. It is guaranteed that this data describes a binary tree.

Output

Output the lexicographically minimal input permutation that produces the given tree under the insertion method for skew heaps, followed by the lexicographically maximal such input permutation. These permutations may coincide, in which case you still need to output both. If no input permutation producing the given tree exists, output `impossible`.

Sample Input 1

```
7
2 3
4 5
6 7
0 0
0 0
0 0
0 0
```

Sample Output 1

```
1 3 2 7 5 6 4
7 1 5 3 2 6 4
```

Sample Input 2

```
2
0 2
0 0
```

Sample Output 2

```
impossible
```

Sample Input 3

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

Sample Output 3

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