Competitive Programming SS24

Submit until end of contest

Problem: order (1.5 second timelimit)

You just started a contest containing n problems. The problem setters intended there to be no two problems of the same difficulty, so ideally, you would like to solve them from the easiest to the hardest. After reading the problem statements, you made a guess $1 \leq l \leq r \leq n$ for each problem, meaning that you think this problem is at least the l-th easiest and at most the r-th easiest. Due to your plethora of experience you can guarantee that the real difficulty is always in the range you guessed.

You now want to derive an order to solve the problems in from these guesses. You consider an order valid if from your guesses it is possible that the difficulties are increasing. You are satisfied with any of them.

Input The input begins with n ($2 \leq n \leq 3 \cdot 10^5$), the number of problems. The i-th of the next n lines contains l_i, r_i ($1 \leq l_i \leq r_i \leq n$), denoting that you guess the i-th problem to be between the l_i-th and r_i-th easiest problem.

Output Output a single line, containing the problem ids (numbers 1 to n) in any order. The number i must appear between the l_i-th and r_i-th position in the list.

It is guaranteed that there is at least one valid solution. If there are multiple, you can print any of them.

Sample input

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

Sample output

```
1 3 2
```

Sample input

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

Sample output

```
3 1 2
```

Sample notes In the first sample, your guesses only allow for one valid solution. In the second, $[1, 3, 2]$ would have also been a valid order.