



Problem D. Withdraw

Problem Description

You signed up for the Introduction to Algorithms course this semester, and found yourself stuck on Lab 0 (yep, the pre-course basics). Discouraged, you made up your mind on Week 3 of the semester that this is the course to withdraw. With the forms filled, you came the professor's door for the last signature. Unsurprisingly, many of your classmates are there too, queuing up to quit the course.

Let's switch to the perspective of the frustrated professor. He knew the TAs "accidentally" made Lab 0 too hard, but sees the potential in these students and tried to encourage them to stay on. For each student, the professor has a potential value p_i . As long as the potential value is positive, the professor will ask the student to rethink about it. But once the student is out of the room, the nightmare of Lab 0 shadows his mind once again, thus he rejoins the queue immediately. Thus in the end, every student that came to queue never left it with the form unsigned.

Of course the professor has other researches to work on, thus he can't do this all day. He still rejects students' withdrawal if he thinks they're still capable, but the potential value decreases by one on each time of ask. On the occasion the potential value turns zero, he finally gets fed up and signs the form for that student.

To remind himself of how many occasions each student came asking, the professor asked the TAs to keep a long record of the order each student came to withdraw from the course. The TAs are indeed annoying, but this time it is them annoyed. Given the initial order, IDs and potential values of each student, can you help the TAs get the overall sequence from the first ask to the last sign?

Input Format

On the first line there is an integer n indicating the number of students that chose to drop out of the course. n lines follow, each with two numbers, the student id id_i and their initial potential p_i . The n lines are listed in the initial order of the queue.

Output Format

Output one number on each line, each with an ID of the student that the professor met. Note that the numbers must be printed in time order, thus the first ID indicates the first student the professor met, and the last student the professor met is printed on the last line.

Constraints

- $1 \leq n \leq 100\,000$.
- $1 \leq id_i \leq 10^9$ for $i = 1, 2, \dots, n$.
- $id_i \neq id_j$ for $i \neq j$.
- $1 \leq p_i \leq 100\,000$ for $i = 1, 2, \dots, n$.
- $\sum_{i=1}^n p_i \leq 10\,000\,000$.
- All the inputs are integers.

Subtasks

1. (100 points) No additional constraints.

No.	Testdata Range	Time Limit (ms)	Memory Limit (KiB)
Samples	1	2000	262144
1	1-15	2000	262144

Samples

Sample Input 1

```
5
3 2
2 2
5 1
1 3
4 2
```

Sample Output 1

```
3
2
5
1
4
3
2
1
4
1
```

Everyone visited the professor in their first round, but student 5 left the queue with his form signed. The remaining 4 students queued for a second time, and all except student 1 left the queue afterwards. Student 1 signed his withdrawal form after the third round, thus the queue is emptied.

Hint

On output, avoid using `cout << endl;` as it may exceed time limits. `cout << '\n'` or `printf('\n')` are execution-wise faster options.