

## C. 狐狸家族 (foxfamilies)

Time limit: 2 seconds

Memory limit: 1024 MiB

阿爾卑斯山的一大片區域最近被劃定為自然保護區。一開始，保護區裏都沒有狐狸。不過多虧了持續的保育措施，保護區內的狐狸數量正一天天恢復。每天都會有一隻新狐狸來到這裡。生物學家 Simona 正在觀察這個恢復過程，她對於任何時刻狐狸們組成的不同家族數量很感興趣。Simona 知道，每隻狐狸  $i$  都有一塊獲取獵物的領地，可以用一個線段  $[L_i, R_i]$  ( $L_i < R_i$ ) 來表示。這些領地可能會互相重疊，甚至完全被包含在另一塊領地裡面。根據她的研究，Simona 知道：如果兩隻狐狸  $i$  及  $j$  的狩獵領地有其中一塊被包含在另一塊裡面（也就是  $L_i \leq L_j < R_j \leq R_i$  或是  $L_j \leq L_i < R_i \leq R_j$ ），那這兩隻狐狸就是直系親屬。兩隻狐狸屬於同一個家族，是指牠們是直系親屬，或是透過一連串的直系親屬關係而有關連。<sup>1</sup>

狐狸  $i$  ( $0 \leq i \leq N-1$ ) 會在第  $i$  天抵達，然後就一直待在保護區裡，永遠維持著同樣的狩獵領地  $[L_i, R_i]$ 。每隻狐狸的到來可能改變、也可能不會改變家族間的關係。在每一天結束時，Simona 都想知道在狐狸  $i$  抵達之後，總共有幾個狐狸家族。

### Input

輸入的第一行包含一個整數  $N$ ，表示總天數。接下來的  $N$  行，每行包含兩個整數  $L_i$  及  $R_i$ ，用來描述狐狸  $i$  的狩獵領地範圍。

### Output

輸出  $N$  行。第  $i$  行 ( $0 \leq i \leq N-1$ ) 輸出一個整數，表示狐狸  $i$  抵達後當時的狐狸家族數量。

### Constraints

- $1 \leq N \leq 100\,000$ .
- $0 \leq L_i < R_i \leq 200\,000$ .
- 沒有任何一組  $(L_i, R_i)$  會重複出現。

### Scoring

你的程式將會在分組成子任務 (subtasks) 的一些測資上進行測試。要獲得一個子任務的分數，你必須正確解出該子任務中包含的所有測資。

- Subtask 0 [0 points]: 範例測資。
- Subtask 1 [10 points]:  $N \leq 100$ .
- Subtask 2 [15 points]:  $N \leq 2000$ .
- Subtask 3 [16 points]:  $R_i - L_i \leq 2$ .
- Subtask 4 [23 points]:  $L_i < L_{i+1}$ .
- Subtask 5 [36 points]: 沒有額外限制。

<sup>1</sup>正式定義的說法是：兩隻狐狸  $a$  及  $b$  屬於同一個家族，若且唯若存在一個狐狸序列  $c_0, c_1, \dots, c_{m-1}$ ，使得  $a = c_0$  且  $b = c_{m-1}$ ，並且對於所有的  $0 \leq i < m-1$ ， $c_i$  是  $c_{i+1}$  的直系親屬。

## Examples

stdin	stdout
4 1 4 3 6 3 4 6 7	1 2 1 2
6 0 1 1 2 2 3 3 4 4 5 2 4	1 2 3 4 5 4
5 0 5 1 4 2 7 3 6 4 5	1 1 2 2 1

## Explanation

第一個範例符合子任務 1、2 和 5 的範圍限制。第二個範例符合子任務 1、2、3 和 5 的範圍限制。第三個範例符合子任務 1、2、4 和 5 的範圍限制。

**第一個範例。** 第一隻狐狸抵達後，只有 1 個家族。第二隻狐狸抵達後，變成有 2 個家族，因為  $[1, 4]$  和  $[3, 6]$  有重疊，但兩個範圍彼此間沒有一個包含另一個的關係。接著，具有領地範圍  $[3, 4]$  的狐狸來了：這塊領地同時被  $[1, 4]$  及  $[3, 6]$  包含，所以這兩個家族就被合併，現在的家族數量變成 1 個。最後，領地範圍  $[6, 7]$  的狐狸抵達，這範圍沒有包含之前的任一塊領地範圍，也沒被任何領地包含，所以牠自己成立一個新家族，家族數量變成 2 個。

