

The citizens of Bytetown, AB, could not stand that the candidates in the mayoral election campaign have been placing their electoral posters at all places at their whim. The city council has finally decided to build an *electoral wall* for placing the posters and introduce the following rules:

- Every candidate can place exactly one poster on the wall.
- All posters are of the same height equal to the height of the wall; the width of a poster can be any integer number of bytes (*byte* is the unit of length in Bytetown).
- The wall is divided into segments and the width of each segment is one byte.
- Each poster must completely cover a contiguous number of wall segments.

They have built a wall 10000000 bytes long (such that there is enough place for all candidates). When the electoral campaign was restarted, the candidates were placing their posters on the wall and their posters differed widely in width. Moreover, the candidates started placing their posters on wall segments already occupied by other posters. Everyone in Bytetown was curious whose posters will be visible (entirely or in part) on the last day before elections.

Your task is to find the number of visible posters when all the posters are placed given the information about posters' size, their place and order of placement on the electoral wall.

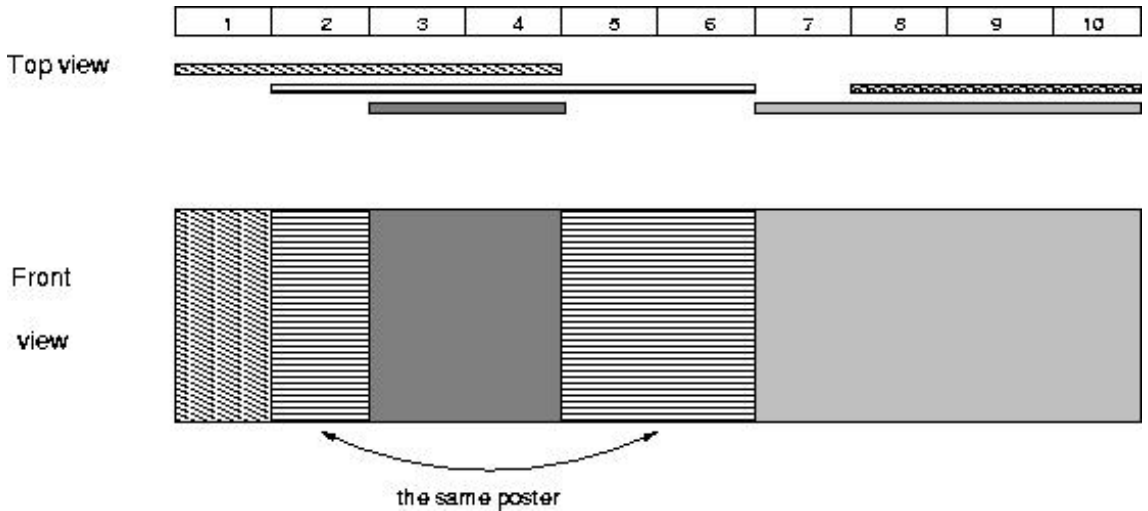
Input

The first line of input contains a number c giving the number of cases that follow. The first line of data for a single case contains number $1 \leq n \leq 10000$. The subsequent n lines describe the posters in the order in which they were placed. The i -th line among the n lines contains two integer numbers l_i and r_i which are the number of the wall segment occupied by the left end and the right end of the i -th poster, respectively. We know that for each $1 \leq i \leq n$, $1 \leq l_i \leq r_i \leq 10000000$. After the i -th poster is placed, it entirely covers all wall segments numbered l_i, l_{i+1}, \dots, r_i .

Output

For each input data set print the number of visible posters after all the posters are placed.

Note: The picture below illustrates the case of the sample input.



Sample Input

```
1
5
1 4
2 6
8 10
3 4
7 10
```

Sample Output

```
4
```