

# Fight against involution

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            1 second  
Memory limit:         256 megabytes

MianKing chose a course in this semester. There are  $n$  students in this course, and everyone needs to write a final paper. Let  $w_i$  denote the word count of the  $i$ -th student's final paper.

The  $i$ -th student has a lower bound  $L_i$  and an upper bound  $R_i$  on the number of words in his final paper so that  $L_i \leq w_i \leq R_i$

The grade rule of this course is very amazing. The grade of the  $i$ -th student  $g_i$  is  $n - K_i$ ,  $K_i$  is the number of  $j \in [1, n]$  satisfies that  $w_j > w_i$ .

Every student wants to achieve the highest possible grade, so under the optimal decision  $w_i$  will equal to  $R_i$  for the  $i$ -th student.

But MianKing found an interesting thing: let's assume that  $\forall i \in [1, n], L_i = 1000, R_i = 10000$ . Under the optimal decision  $w_i$  are all equal to 10000 and the grades of the students are all  $n$ . But if everyone only writes 1000 words in their final papers, their grades are still all  $n$  and they can use the time they save to play games.

Now to fight against involution, MianKing wants to decide  $w_i$  for each student, and his plan has to satisfy the following conditions:

1. For each student, his grade cannot be less than that in the original plan.
2. Minimize the sum of  $w_i$ .

You need help MianKing calculate the minimum value of  $\sum_{i=1}^n w_i$

## Input

The first line has one integer  $n$ .

Then there are  $n$  lines, the  $i$ -th line has two integers  $L_i, R_i$ .

$$1 \leq n \leq 10^5$$

$$1 \leq L_i \leq R_i \leq 10^9$$

## Output

Output the minimum value of  $\sum_{i=1}^n w_i$ .

## Examples

standard input	standard output
3 1 10000 1 10000 1 10000	3
4 1 2 2 2 2 4 3 4	10