

C. Fox Families (foxfamilies)

Una gran área en los Alpes ha sido declarada recientemente reserva natural. Al principio, no había zorros en la reserva. Sin embargo, la población de zorros en la reserva natural ha estado recuperándose día a día gracias a las medidas de conservación en curso. Cada día llega un nuevo zorro. La bióloga Simona está observando el proceso de recuperación y le interesa saber el número de familias distintas que forman los zorros en cualquier momento dado. Simona sabe que cada zorro i tiene un territorio de caza que puede representarse mediante un segmento $[L_i, R_i]$ con $L_i < R_i$. Estos territorios pueden solaparse o incluso estar contenidos el uno dentro del otro. A partir de sus estudios, Simona sabe que dos zorros i y j son *parientes directos* si uno de sus territorios de caza está anidado dentro del otro (ya sea $L_i \leq L_j < R_j \leq R_i$ o $L_j \leq L_i < R_i \leq R_j$). Dos zorros pertenecen a la misma *familia* si y solo si están relacionados directamente o si están conectados a través de una cadena de zorros relacionados directamente.¹

El zorro i ($0 \leq i \leq N - 1$) llega el día i y permanece en la reserva desde entonces, manteniendo el mismo territorio de caza $[L_i, R_i]$ para siempre. La llegada de cada zorro puede cambiar o no las relaciones familiares. Después de cada día, Simona quiere saber el número de familias de zorros después de que haya llegado el zorro i .

Entrada

La primera línea de la entrada contiene un único número entero N , el número de días. Las siguientes N líneas contienen dos números enteros cada una, L_i y R_i , que describen el territorio de caza del zorro i .

Salida

Imprime N líneas. La línea i (para $0 \leq i \leq N - 1$) debe contener un único número entero, el número de familias de zorros que existían después de que llegara el zorro i .

Restricciones

- $1 \leq N \leq 100\,000$.
- $0 \leq L_i < R_i \leq 200\,000$.
- Ningún par (L_i, R_i) aparecerá más de una vez.

Puntuación

Tu programa será probado en varios casos de prueba agrupados en subtareas. Para obtener la puntuación de una subtarea, debes resolver correctamente todas las pruebas que contiene.

- **Subtask 0 [0 puntos]:** Ejemplos.
- **Subtask 1 [10 puntos]:** $N \leq 100$.
- **Subtask 2 [15 puntos]:** $N \leq 2000$.
- **Subtask 3 [16 puntos]:** $R_i - L_i \leq 2$.
- **Subtask 4 [23 puntos]:** $L_i < L_{i+1}$.
- **Subtask 5 [36 puntos]:** Sin restricciones adicionales.

¹Formalmente, dos zorros a y b están en la misma familia si y solo si existe una secuencia de zorros c_0, c_1, \dots, c_{m-1} tal que $a = c_0$ y $b = c_{m-1}$, y c_i está relacionado directamente con c_{i+1} para todo $0 \leq i < m - 1$.

Ejemplos de entrada/salida

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

Explicación

El primer ejemplo cumple con las restricciones de las subtareas 1, 2 y 5. El segundo ejemplo cumple con las restricciones de las subtareas 1, 2, 3 y 5. El tercer ejemplo cumple con las restricciones de las subtareas 1, 2, 4 y 5.

Primer Ejemplo. Después de que llega el primer zorro, hay una familia. Después de que llega el segundo zorro, hay dos familias, ya que $[1, 4]$ y $[3, 6]$ se solapan pero ninguno de los territorios contiene al otro. Luego llega el zorro con territorio $[3, 4]$: está contenido tanto en $[1, 4]$ como en $[3, 6]$, por lo que estas dos familias se fusionan y el número de familias es ahora 1. Finalmente, el zorro con territorio $[6, 7]$ no contiene ningún territorio anterior y no está contenido dentro de ninguno de ellos, por lo que forma una nueva familia y el número de familias es ahora 2.

