

## C. Familii de vulpi (foxfamilies)

O zonă întinsă din Alpi a fost declarată recent rezervație naturală. La început, nu existau vulpi în rezervație. Totuși, populația de vulpi din rezervație și-a revenit pe zi ce trece datorită măsurilor de conservare. În fiecare zi, sosește o nouă vulpe. Biologa Simona observă procesul de recuperare și este interesată de numărul de familii distincte pe care le formează vulpile în orice moment. Simona știe că fiecare vulpe  $i$  are un teritoriu de vânătoare care poate fi reprezentat printr-un segment  $[L_i, R_i]$ , cu  $L_i < R_i$ . Aceste teritorii se pot suprapune sau pot fi chiar incluse unul în altul. Din studiile sale, Simona știe că două vulpi  $i$  și  $j$  sunt *rude directe* dacă unul dintre teritoriile lor de vânătoare este inclus în celălalt (fie  $L_i \leq L_j < R_j \leq R_i$ , fie  $L_j \leq L_i < R_i \leq R_j$ ). Două vulpi aparțin aceleiași *familii* dacă și numai dacă sunt fie înrudite direct, fie conectate printr-un lanț de vulpi înrudite direct.<sup>1</sup>

Vulpea  $i$  ( $0 \leq i \leq N - 1$ ) sosește în ziua  $i$  și rămâne în rezervație de atunci înainte, păstrându-și același teritoriu de vânătoare  $[L_i, R_i]$  pentru totdeauna. Sosirea fiecărei vulpi poate schimba sau nu relațiile de familie. După fiecare zi, Simona vrea să știe numărul de familii de vulpi după sosirea vulpii  $i$ .

### Date de intrare

Prima linie a fișierului de intrare conține un singur număr întreg  $N$ , numărul de zile. Următoarele  $N$  linii conțin câte două numere întregi,  $L_i$  și  $R_i$ , descriind teritoriul de vânătoare al vulpii  $i$ .

### Date de ieșire

Afișați  $N$  linii. Linia  $i$  (pentru  $0 \leq i \leq N - 1$ ) trebuie să conțină un singur număr întreg, numărul de familii de vulpi care există după ce a sosit vulpea  $i$ .

### Constrângeri

- $1 \leq N \leq 100\,000$ .
- $0 \leq L_i < R_i \leq 200\,000$ .
- Nicio pereche  $(L_i, R_i)$  nu va apărea de mai multe ori.

### Punctaj

Programul tău va fi testat pe mai multe seturi de date de test grupate în subtask-uri. Pentru a obține punctajul pe un subtask, trebuie să rezolvi corect toate testele din acesta.

- **Subtask-ul 0 [ 0 puncte]:** Exemple.
- **Subtask-ul 1 [10 puncte]:**  $N \leq 100$ .
- **Subtask-ul 2 [15 puncte]:**  $N \leq 2000$ .
- **Subtask-ul 3 [16 puncte]:**  $R_i - L_i \leq 2$ .
- **Subtask-ul 4 [23 puncte]:**  $L_i < L_{i+1}$ .
- **Subtask-ul 5 [36 puncte]:** Fără restricții suplimentare.

<sup>1</sup>Formal, două vulpi  $a$  și  $b$  fac parte din aceeași familie dacă și numai dacă există un șir de vulpi  $c_0, c_1, \dots, c_{m-1}$  astfel încât  $a = c_0$  și  $b = c_{m-1}$ , iar  $c_i$  este înrudită direct cu  $c_{i+1}$  pentru orice  $0 \leq i < m - 1$ .

## Exemple de intrare/ieșire

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	

## Explicație

Primul exemplu respectă restricțiile subtask-urilor 1, 2 și 5. Al doilea exemplu respectă restricțiile subtask-urilor 1, 2, 3 și 5. Al treilea exemplu respectă restricțiile subtask-urilor 1, 2, 4 și 5.

**Primul exemplu.** După sosirea primei vulpi, există o singură familie. După sosirea celei de-a doua vulpi, există două familii, deoarece  $[1, 4]$  și  $[3, 6]$  se suprapun, dar niciun teritoriu nu îl conține pe celălalt. Apoi sosește vulpea cu teritoriul  $[3, 4]$ : aceasta este conținută atât în  $[1, 4]$ , cât și în  $[3, 6]$ , deci aceste două familii se unesc și numărul de familii devine 1. În final, vulpea cu teritoriul  $[6, 7]$  nu conține niciun teritoriu anterior și nu este conținută în niciunul dintre ele, așa că formează o familie nouă și numărul de familii devine 2.

