Contando Kaikas

Contribución de Agustín Santiago Gutiérrez

Descripción del problema

Kanako, Iwao y Kaori son tres amigos que aman los juegos. Les encanta inventar sus propios juegos. Recientemente, inventaron un nuevo juego llamado Kaika. Es un juego de 3 jugadores excesivamente complicado, que involucra realizar pasos de baile muy difíciles mientras se comunican entre los 3 jugadores mediante señas con banderas.

El lugar de juego será en la plaza de la ciudad, que tiene **N** claros y **M** senderos. Cada sendero conecta exactamente dos claros, y es un camino **en línea recta** desde un claro hasta el otro. **Dos senderos jamás se cruzan entre sí**, exceptuando en los claros que están en los extremos de los senderos.

Para jugar Kaika, los amigos Kanako, Iwao y Kaori deben pararse cada uno en uno de los **N** claros. Cada uno de los tres se para en un claro distinto. Dado que es necesario tener una línea de visión directa y sin obstáculos entre cada par de jugadores, los tres claros elegidos deben estar directamente conectados entre sí mediante senderos. Es decir, debe haber un sendero entre Kanako e Iwao, un sendero entre Kanako y Kaori, y un sendero entre Iwao y Kaori.

A los tres les encanta jugar Kaika, y planean jugar una y otra vez. Para maximizar la diversión, cada día jugarán utilizando un *Kaika* diferente. Un Kaika no es más que una elección del claro donde estará Kanako, el claro donde estará Iwao, y el claro donde estará Kaori. Dos Kaikas se consideran diferentes si al menos uno de los tres se para en un claro diferente.

Por más que a los amigos les encantaría jugar Kaika por toda la eternidad, solamente hay un número limitado de Kaikas posibles. Esto los pone muy tristes.

Debes escribir una función que dadas las posiciones $(\mathbf{x}, \mathbf{y})$ de cada uno de los **N** claros, y los **M** pares de claros que están conectados mediante senderos, calcule por cuántos días podrán jugar al juego. Es decir, se debe computar la cantidad de Kaikas diferentes.

Descripción de la función

Debes implementar la función `kaikas(x,y,a,b)`, que recibe:

- `x,y`: Arreglos de longitud **N**. El claro **i** se encuentra ubicado en `(x[i],y[i])`, $\mathbf{0 \le i < N}$
- `a,b`: Arreglos de longitud **M**. El sendero **i** conecta entre sí los claros **a[i]** y **b[i]**, $\mathbf{0 \le i < M}$

La función retorna un entero con la cantidad de Kaikas.

Evaluador

El evaluador local lee de la entrada estándar con el siguiente formato:

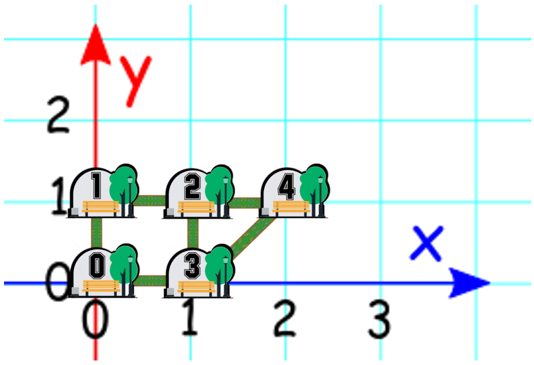
- En la primera línea, dos enteros **N** y **M**, la cantidad de claros y de senderos, respectivamente.
- Luego, **N** líneas, cada una con dos enteros $\mathbf{x_i}$ y $\mathbf{y_i}$
- Finalmente, **M** líneas más, cada una con dos enteros $\mathbf{a_i}$ y $\mathbf{b_i}$

El evaluador local escribe a la salida estándar el resultado de la función `kaikas`.

Cotas

- $\mathbf{1 \le N, M \le 500.000}$
- $\mathbf{0 \le x_i, y_i \le 10^9}$
- $\mathbf{0 \le a_i, b_i < N}$

Ejemplo



Si se invoca al evaluador con la siguiente entrada:

```
5 6
0 0
0 1
1 1
1 0
2 1
0 1
1 2
2 3
3 0
4 3
4 2
```

Para un programa correcto, la salida será:

```
6
```

Subtareas

1. $\mathbf{N} \leq \mathbf{100}$ (14 puntos)
2. $\mathbf{N} \leq \mathbf{2.000}$ (16 puntos)
3. $\mathbf{N} \leq \mathbf{60.000}$ (17 puntos)
4. En cada claro, $\mathbf{x_i} \cdot \mathbf{y_i} = \mathbf{0}$ (7 puntos)
5. Se garantiza que para cada Kaika, el triángulo con vértices en los 3 claros del Kaika no contiene ningún claro en su interior (19 puntos)
6. Sin más restricción (27 puntos)