

Batiendo al minotauro

Contribución de Carlos Mendioroz y Hugo Ryckeboer

Descripción del problema

En una ciudad lejana hay un laberinto (una intrincada red de pasajes que unen puntos que llamaremos descansos) y en el medio del mismo un monstruo llamado Minotauro. Cada pasaje une dos descansos para ir de uno a otro caminando una cierta cantidad de pasos.

Teseo ha decidido derrotar al Minotauro pero, por las dudas, quiere saber como salir del laberinto.

Teseo, como cuenta con muchos amigos, envía a varios al laberinto a medir cuantos pasos hay que caminar para ir de cada descanso a cualquier otro descanso, sabiendo que sólo hay un camino para ir de un descanso a otro (algo que le había confiado el arquitecto).

Mucho tiempo después, los amigos lo gran devolverle a Teseo una planilla con las distancias entre los descansos, y Teseo tiene que armar el mapa.

¿Te animas a escribir un programa **minotauro.pas**, **minotauro.cpp**, o **minotauro.c** que dada la matriz de distancias en pasos entre los descansos, calcule cuales son los pasajes que conforman el laberinto?

Datos de entrada

Se recibe un archivo **minotauro.in** con el siguiente formato:

- Primera línea: el número **n** (que indica la cantidad de descansos ($1 \leq n \leq 1000$))
- **n** líneas con **n** números **d_{ij}**, que corresponden a la distancia medida en pasos desde el descanso **i** al descanso **j**, ($0 \leq d_{ij} \leq 100\ 000\ 000$).

Datos de salida

El programa debe generar un archivo **minotauro.out** con una línea conteniendo el número **p** de pasajes que hay en el laberinto, seguido por **p** líneas conteniendo tres números separados por blancos que indican para cada pasaje el descanso de origen, el de destino y el largo del pasaje en pasos.

Nota

No hay pasos asociados a cruzar un descanso.

No hay descansos inaccesibles desde otro descanso.

Ejemplo

En el caso de que la entrada **minotauro.in** fuera:

```
6
0 12 17 16 7 5
12 0 19 18 9 7
17 19 0 23 14 12
16 18 23 0 9 11
7 9 14 9 0 2
5 7 12 11 2 0
```

La salida **minotauro.out** podría ser:

```
5
1 6 5
6 2 7
3 6 12
4 5 9
5 6 2
```