

## Contando manzanas...

Contribución de Natalia Pérez

## Descripción del problema

La Organización Argentina de Ciclismo desea efectuar una competencia en una ciudad caracterizada por su apoyo al deporte. La idea es realizar una carrera cuyo recorrido encierre la mayor cantidad de manzanas. Además, por motivos organizacionales, también se desea establecer la largada y la meta en un mismo punto.

La circulación en el pueblo se realiza por amplias avenidas con algunos árboles que separan ambas manos, con lo cual ambas manos son utilizables para la carrera. No obstante, la Organización dispuso que, si bien todas las calles podrían recorrerse en doble sentido, no se encuentra permitido girar en U para controlar que ningún participante haga trampa y retorne antes del final de la cuadra. Tampoco está prohibido que la solución pase dos veces por una misma esquina a condición de que, por razones obvias, el trazado elegido para la carrera no se cruce a sí mismo.

Por otra parte, muchas de sus avenidas no son aptas para formar parte del recorrido. Se ha recolectado información acerca de las cuadras aptas las cuales se describen identificando el par de esquinas que las delimitan. Por economía de la descripción cuadras consecutivas de una misma calle cuyas esquinas intermedias no dan lugar a bifurcaciones se encierran en una sola descripción dando las esquinas extremas. Finalmente se desea que el recorrido propuesto encierre a la mayor cantidad de cuadras y si hubiera más de una elección para ello que se proponga la de mayor longitud.

El intendente confeccionó un recorrido por las avenidas aptas cerrando un circuito, pero como no tiene garantía de ser de máxima superficie, necesita tu ayuda. Es por esto que se te pide que escribas una función

**recorrido**( **EO**, **NS**, **E**, **Esquinas**, **C**, **Conexiones**, **perimetro**, **manzanas**) que calcule el perímetro del recorrido que encierra la mayor cantidad de manzanas. Sus parámetros son:

**EO**: un entero que indica la cantidad de avenidas con sentido este-oeste de la ciudad ( $2 \leq \mathbf{EO} \leq 30.000$ )

**NS**: un entero que indica la cantidad de calles con sentido norte-sur de la ciudad ( $2 \leq \mathbf{NS} \leq 30.000$ )

**E**: un entero que indica la cantidad de esquinas utilizadas en la descripción ( $4 \leq \mathbf{E} \leq 200.000$ )

**Esquinas**: una matriz de enteros de tamaño  $\mathbf{E} \times 2$  con las ubicaciones (eo,ns) de las esquinas utilizadas. La coordenada **eo** designa la calle este-oeste ( $1 \leq \mathbf{eo} \leq \mathbf{EO}$ ) y la **ns** la calle norte-sur ( $1 \leq \mathbf{ns} \leq \mathbf{NS}$ ). Las esquinas se listan consecutivamente de 1 a **E**.

**C**: un entero que indica la cantidad de conexiones entre las esquinas ( $4 \leq \mathbf{C} \leq 250.000$ )

**Conexiones**: una matriz de enteros de tamaño  $\mathbf{C} \times 2$  con **e1** y **e2** ( $1 \leq \mathbf{e1}, \mathbf{e2} \leq \mathbf{E}$ ) de dos esquinas de una misma avenida. Las cuadras comprendidas pueden ser transitadas.

**perímetro**: un entero en el cual retornar el largo del recorrido que encierra la mayor cantidad de manzanas.

**manzanas**: un entero en el cual retornar la cantidad de manzanas abarcadas por el recorrido.

## Ejemplo

Si se presenta el siguiente caso:

EO=4; NS=5;

E=11; Esquinas (transpuesta):

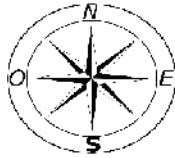
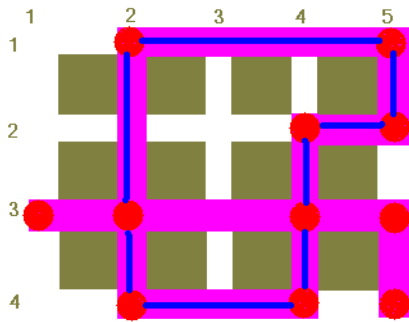
2	3	1	4	3	1	3	4	2	3	4
4	5	2	2	4	5	1	5	5	2	4

C=12; Conexiones (transpuesta)

5	10	1	7	8	2	11	6	4	9	5	1
11	3	9	10	2	5	4	3	10	6	10	5

La solución deberá ser:

perimetro=12; manzanas=7



**Evaluador local**

El evaluador local lee la entrada por stdin en el siguiente formato:

- Una línea con los enteros EO, NS
- Una línea con el entero E
- E líneas con dos enteros cada una indicando las coordenadas eo y ns respectivamente de una de las esquinas de la descripción.
- Una línea con el entero C
- C líneas con los números de las esquinas de una misma calle.

El evaluador reproducirá en una línea la cantidad de esquinas y de conexiones leídas y los valores de perímetro y manzanas.

El archivo **recorrido.in** suministrado contiene:

```
#este es el ejemplo
4 5 11
2 4
3 5
1 2
4 2
3 4
1 5
3 1
4 5
2 5
3 2
4 4
12
5 11
10 3
1 9
7 10
8 2
2 5
11 4
6 3
4 10
9 6
5 10
1 5
```

De aportarlo como entrada a un programa correcto devolvería en pantalla:

Archivo con 11 esquinas y 12 conexiones da perímetro 12 y manzanas 7.

**Subtareas**

Habrà casos de prueba por un total de 20 puntos en los cuales no habrá cuadras que de ser utilizadas forzarían giros en U.

Habrà casos de prueba por un total de 20 puntos en los cuales desde una cuadra apta se pueda llegar a cualquier otra apta usando solamente cuadras aptas.

Habrà casos de prueba por un total de 60 puntos donde el mejor recorrido no pasará dos veces por una misma esquina. Esto puntos incluye los casos anteriores.

Habrà casos de prueba por un total de 30 puntos donde EO y NS no superarán a 1000.

**Puntaje parcial**

Por tener correcto la superficie se recibirá 60 puntos y 40 por el perímetro.