

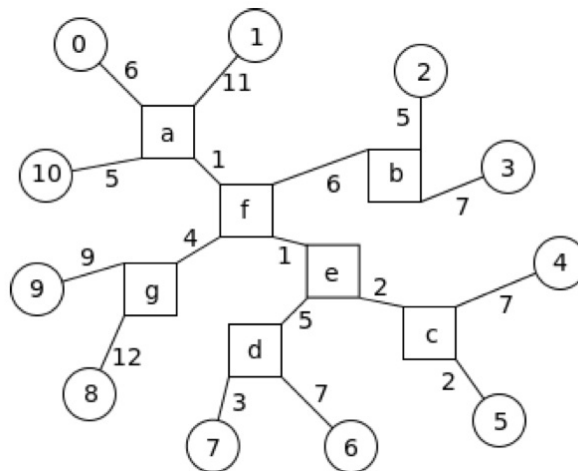
# Pueblitos

Hay  $N$  pueblitos en Kazajistán, numerados de  $0$  a  $N - 1$ . También hay un número desconocido de ciudades grandes. Tanto los pueblitos como las ciudades grandes de Kazajistán se conocen como *asentamientos*.

Todos los asentamientos de Kazajistán están conectados por una única red de carreteras bidireccionales. Cada carretera conecta dos asentamientos distintos y cada par de asentamientos está directamente conectado por a lo más una carretera. Para cada par de asentamientos  $a$  y  $b$ , existe una única manera para ir de  $a$  a  $b$  utilizando las carreteras, asumiendo que cada carretera se utiliza a lo más una vez.

Se sabe que cada pueblito está directamente conectado a un único otro asentamiento, y cada ciudad grande está conectada a tres o más asentamientos.

La siguiente figura muestra una red de **11** pueblitos y **7** ciudades grandes. Los pueblitos se muestran como círculos etiquetados con enteros y las ciudades grandes se muestran como cuadrados etiquetados con letras.



Cada carretera tiene una longitud entera positiva. La distancia entre dos asentamientos es la suma mínima de las longitudes de las carreteras que se deben atravesar para ir de un asentamiento al otro.

Para cada ciudad grande  $C$ , podemos definir  $r(C)$  como la distancia de  $C$  al pueblito más alejado de dicha ciudad. Una ciudad grande  $C$  es un *centro* si la distancia  $r(C)$  es la más pequeña entre todas las ciudades grandes. La distancia entre un centro y el pueblito más alejado del centro se denotará como  $R$ . Es decir,  $R$  es el valor más pequeño de todos los valores  $r(C_i)$  para cualquier  $C_i$ .

En el ejemplo de arriba, el pueblito más alejado de la ciudad grande  $a$  es el pueblito  $8$ , y la distancia entre ellos es  $r(a) = 1 + 4 + 12 = 17$ . Para la ciudad grande  $g$  también tenemos que  $r(g) = 17$ . (Uno de los pueblitos que está más alejado de  $g$  es el pueblito  $6$ ). El único centro en el ejemplo de

arriba es la ciudad grande  $f$ , con  $r(f) = 16$ . Por lo tanto, en el ejemplo de arriba,  $R$  es 16.

Si se elimina un centro, la red se divide en múltiples piezas conectadas. Un centro está *balanceado* si cada una de las piezas sobrantes después de haber quitado el centro contiene a lo más  $\lfloor N/2 \rfloor$  pueblitos. (Hay que enfatizar que no estamos considerando las ciudades grandes). Nota que  $\lfloor x \rfloor$  denota el entero más grande que es menor o igual a  $x$ .

En nuestro ejemplo, la ciudad grande  $f$  es un centro. Si eliminamos la ciudad grande  $f$ , la red se dividirá en cuatro piezas conectadas. Estas cuatro piezas consisten en los siguientes conjuntos de pueblitos:  $\{0, 1, 10\}$ ,  $\{2, 3\}$ ,  $\{4, 5, 6, 7\}$  y  $\{8, 9\}$ . Ninguna de las piezas tiene más de  $\lfloor 11/2 \rfloor = 5$  pueblitos, por lo tanto la ciudad grande  $f$  es un centro balanceado.

## Tarea

Inicialmente, la única información que tienes acerca de la red de asentamientos y carreteras es el número  $N$  de pueblitos. No conoces el número de ciudades grandes. Tampoco sabes nada acerca de la disposición de carreteras del país. Sólo puedes obtener información nueva haciendo preguntas acerca de las distancias entre pares de pueblitos.

Tu tarea es determinar:

- En todas las subtareas: la distancia  $R$ .
- En las subtareas 3 a 6: determinar si existe un centro balanceado en la red.

Necesitas implementar la función `hubDistance`. El evaluador calificará múltiples casos en una sola ejecución. El número de casos por ejecución es a lo más 100. Para cada caso de prueba, el evaluador llamará a tu función `hubDistance` exactamente una vez. Asegúrate que tu función inicializa todas las variables necesarias cada vez que se llama.

- `hubDistance(N, sub)`
  - $N$ : el número de pueblitos.
  - `sub`: el número de subtarea (explicado en la sección Subtareas).
  - Si `sub` es 1 o 2, la función puede regresar ya sea  $R$  o  $-R$ .
  - Si `sub` es mayor a 2, si existe un centro balanceado entonces la función debe regresar  $R$ , de lo contrario debe regresar  $-R$ .

Tu función `hubDistance` puede obtener información de la red de carreteras llamando a la función del evaluador `getDistance(i, j)`. Esta función regresa la distancia entre los pueblitos  $i$  y  $j$ . Nota que si  $i$  y  $j$  son iguales, la función regresará 0. También regresará 0 si los argumentos son inválidos.

## Subtareas

En cada caso de prueba:

- $N$  es un entero entre 6 y 110, inclusive.
- La distancia entre dos pueblitos distintos es entre 1 y 1,000,000, inclusive.

El número de preguntas que tu programa puede hacer es limitado. El límite varía por subtarea, y se indica en la tabla de abajo. Si tu programa intenta exceder el límite del número de preguntas, será

terminado y se asumirá que dio una respuesta incorrecta.

subtarea	puntos	número de preguntas	encontrar centros balanceados	restricciones adicionales
1	13	$\frac{N(N-1)}{2}$	NO	ninguna
2	12	$\lceil \frac{7N}{2} \rceil$	NO	ninguna
3	13	$\frac{N(N-1)}{2}$	SÍ	ninguna
4	10	$\lceil \frac{7N}{2} \rceil$	SÍ	cada ciudad grande está conectada a <i>exactamente</i> tres asentamientos
5	13	$5N$	SÍ	ninguna
6	39	$\lceil \frac{7N}{2} \rceil$	SÍ	ninguna

Nota que  $\lceil x \rceil$  denota el entero más pequeño que es mayor o igual a  $x$ .

## Evaluador de ejemplo

Nota que el número de subtarea es parte de la entrada. El evaluador de ejemplo cambia su comportamiento de acuerdo al número de subtarea.

El evaluador de ejemplo lee la entrada del archivo `towns.in` en el siguiente formato:

- línea 1: El número subtarea y el número de casos de prueba.
- línea 2:  $N_1$ , el número de pueblitos en el primer caso de prueba.
- siguientes  $N_1$  líneas: El  $j$ -ésimo número ( $1 \leq j \leq N_1$ ) en la  $i$ -ésima de estas líneas ( $1 \leq i \leq N_1$ ) es la distancia entre los pueblitos  $i - 1$  y  $j - 1$ .
- El resto de los casos tendrán el mismo formato que el primer caso de prueba.

Para cada caso de prueba, el evaluador de ejemplo imprime el valor de retorno de `hubDistance` y el número de llamadas en líneas separadas.

El archivo de entrada que corresponde al ejemplo de arriba es:

```

1 1
11
0 17 18 20 17 12 20 16 23 20 11
17 0 23 25 22 17 25 21 28 25 16
18 23 0 12 21 16 24 20 27 24 17
20 25 12 0 23 18 26 22 29 26 19
17 22 21 23 0 9 21 17 26 23 16
12 17 16 18 9 0 16 12 21 18 11
20 25 24 26 21 16 0 10 29 26 19
16 21 20 22 17 12 10 0 25 22 15
23 28 27 29 26 21 29 25 0 21 22
20 25 24 26 23 18 26 22 21 0 19
11 16 17 19 16 11 19 15 22 19 0

```

El formato es bastante diferente a especificar la lista de carreteras. Nota que tienes permiso de modificar los evaluadores de ejemplo para que utilicen un formato de entrada distinto.