



EJERCICIO 2 – JOGGING

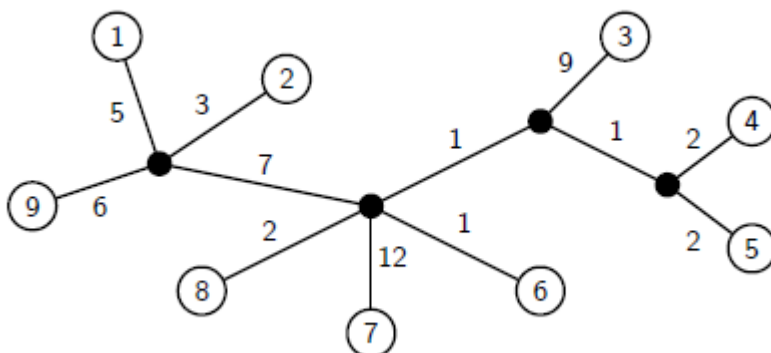
Todas las mañanas, Marcos sale a correr por el barrio. Las casas del barrio, numeradas de 1 a n , están conectadas por un conjunto de caminos con la propiedad que, entre dos casas cualesquiera, existe un único camino. La distribución de las casas conforma un árbol con n hojas (cada hoja es una casa) y una cantidad de hasta $n-1$ nodos internos que representan intersecciones (donde el camino se divide o se junta).

Se trata de ayudar a Marcos a planificar una ruta para correr desde una casa del barrio a otra, con la condición de que la ruta lleve lo máximo posible, esto es, no exista ningún par de casas diferentes tal que la ruta entre ellas le lleve más tiempo. El tiempo de una ruta se calcula considerando los metros recorridos, el tiempo que demora en correr un metro y el tiempo que demora en cada intersección.

Dada una matriz de distancias (en metros) entre cada par de casas, el número r de segundos que demora en correr un metro, y el número t de segundos que lleva cruzar una intersección de la ruta, determinar el tiempo máximo posible.

Ejemplo:

Una posible vecindad de 9 casas (representadas con nodos externos blancos) con intersecciones marcadas en nodos internos negros sería:



Los **únicos** datos que se disponen son: la matriz de distancias entre cada par de casas, el valor r y el valor t . Dado $r=1$ y $t=5$, el camino entre las casas (1 y 2) demora $8r + t = 13$; entre (3 y 9) demora: $(9+1+7+6)*r + 3*t$, o sea, $23+15 = 38$ segundos (que sería el valor máximo). La respuesta sería 38.

Archivo de Entrada:

El archivo de entrada contiene múltiples casos. Cada caso empieza con una línea con 3 enteros: el total de casas n ($1 \leq n \leq 50$), el número r de segundos que lleva correr 1 metro ($1 \leq r \leq 10$) y el número t de segundos que lleva atravesar cada intersección ($1 \leq t \leq 100$).

Las siguientes n líneas del caso contienen las distancias. Por cada nodo i ($1 \leq i \leq n$), hay una línea con n valores, cada uno representa la distancia d_{ij} ($1 \leq d_{ij} \leq 1000$) a cada una de las j casas ($1 \leq j \leq n$).

Se asume que $d_{ii} = 0$ y que $d_{ij} = d_{ji}$ para todo i, j . La matriz corresponde a un árbol válido, ninguna casa coincide con un nodo interno, cada nodo interno tiene grado ≥ 3 (esto es, sobre él inciden al menos 3 aristas) y todas las aristas del árbol tiene valor positivo.

La entrada termina con una línea con un único valor 0, que no debe ser procesada.

Ejemplo de entrada

9 1 5
0 8 22 16 16 13 24 14 11
8 0 20 14 14 11 22 12 9
22 20 0 12 12 11 22 12 23
16 14 12 0 4 5 16 6 17
16 14 12 4 0 5 16 6 17
13 11 11 5 5 0 13 3 14
24 22 22 16 16 13 0 14 25
14 12 12 6 6 3 14 0 15
11 9 23 17 17 14 25 15 0
0

Archivo de Salida:

Por cada uno de los casos, imprimir una línea con "Caso x:", siendo x el número de caso, empezando desde 1. Indicar el tiempo máximo de ese caso.

Ejemplo de salida

Caso 1: 38