

Creando un emporio

Contribución de Agustín Santiago Gutiérrez

Descripción del problema

Un acaudalado empresario ha puesto sus ojos en el pequeño pueblo de Líteltaun con vistas a transformarlo en un emporio: una ciudad floreciente a la que todos concurren para el comercio desde cualquier lugar del mundo.

En el pueblo existen P puntos de interés numerados de 0 a $P-1$ y R rutas, numeradas de 0 a $R-1$, que conectan dos de ellos de manera bidireccional. Es posible viajar entre dos puntos de interés cualesquiera utilizando las rutas del pueblo.

El empresario decidió crear paseos comerciales en algunas rutas del pueblo estratégicamente elegidas. Para garantizar su éxito, quiere que luego de la instalación de los paseos comerciales, sea posible viajar entre dos puntos de interés cualesquiera utilizando únicamente rutas en las cuales se haya instalado un paseo comercial.

No todas las rutas resultan igual de atractivas para la instalación de los paseos: para la ruta i , se conoce la cantidad de personas T_i que la transitan diariamente, así como la cantidad de dinero promedio D_i que dichas personas gastarían en un paseo comercial si se instalase uno en dicha ruta. Este gasto es 100% ganancia para el empresario, que solamente pagará un costo de mantenimiento diario fijo M_i por un paseo instalado en la ruta i . Si no se instala un paseo comercial sobre una ruta, esta no genera costos ni ganancia.

Se te pide calcular la máxima ganancia diaria posible que puede obtener el empresario, y un conjunto de rutas en las que debe instalar paseos comerciales para lograrlo.

Nota: La ganancia podría ser negativa si no existe manera de que el empresario pueda crear en el pueblo un emporio que no pierda dinero.

Descripción de la función

Debes implementar la función

```
emporio(P : ENTEROS;  
a,b,t,d,m : ARREGLO[R] de ENTEROS;  
paseos : ARREGLO[] de ENTEROS  
) : ENTERO LARGO
```

que toma los siguientes parámetros:

- P : la cantidad de puntos de interés
- a y b : arreglos que indican los puntos de interés unidos por la ruta i ($0 \leq i < R$)
- t : arreglo que contiene la cantidad de personas que transitan la ruta i ($0 \leq i < R$)
- d : arreglo con la cantidad de dinero promedio que gastarían una persona en un paseo comercial instalado en la ruta i ($0 \leq i < R$)
- m : arreglo que contiene el costo de instalar un paseo comercial en la ruta i ($0 \leq i < R$)

La función deberá devolver en un ENTERO LARGO la máxima ganancia que podrá obtener el empresario y en la variable *paseos* las rutas (sin repetidos) en las que se deben instalar paseos comerciales para obtener esa ganancia.

Evaluador

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

- Una línea con dos números: la cantidad de puntos de interés **P** y la cantidad de rutas **R**
- **R** líneas, cada una con 5 enteros que representan los valores de a, b, t, d y m para cada ruta. La ruta une los puntos de interés a y b , la transitan t personas diariamente, el gasto promedio en un paseo es d y el costo de mantenimiento diario de un paseo sería m .

El evaluador devuelve una primera línea con la ganancia indicada por la función `emporio`, y luego una segunda línea con los valores del arreglo `paseos`, separados por espacios.

Puntaje

- 40% por la ganancia máxima posible
- 60% por indicar un conjunto de rutas óptimo para instalar los paseos

Cotas

$$2 \leq P \leq 100.000$$

$$1 \leq R \leq 500.000$$

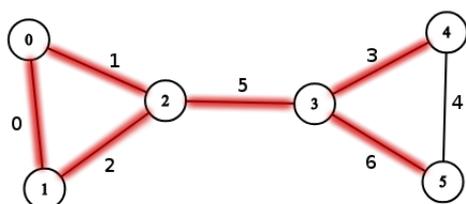
$$0 \leq a_i, b_i < P$$

$$a_i \neq b_i$$

$$0 \leq t_i, d_i \leq 10^6$$

$$0 \leq m_i \leq 10^9$$

Ejemplo



Si se invoca al evaluador con la siguiente entrada:

6	7				
0	1	3	4	10	
2	0	5	5	22	
2	1	3	1	0	
3	4	1	1	0	
5	4	1	0	1	
2	3	2	4	10	
3	5	1	5	1	

Este llamaría a la función `emporio` suministrando los datos correspondientes, y con una solución correcta podría devolver:

11					
0	2	1	3	5	6

Que corresponde al `emporio` del dibujo. Notar que no conviene construir un paseo en la ruta 4, pues el empresario perdería dinero al hacerlo. El paseo de la ruta 5 también pierde dinero, pero en este ejemplo es inevitable instalarlo si queremos crear un `emporio`.

Subtareas

Hay subtareas de 4 tipos:

1. $m_i = 0$ para todo i , es decir: No hay costo de mantenimiento.
2. $t_i = 0, m_i = 1$ para todo i , es decir: Nadie transita las rutas, y todos los paseos tienen el mismo costo de mantenimiento.
3. $t_i = 0$, es decir: Nadie transita las rutas.
4. Sin restricción adicional sobre t_i, m_i .

Además, habrá subtareas con R pequeño. La siguiente tabla resume los puntajes de las subtareas:

	Tipo 1	Tipo 2	Tipo 3	Tipo 4
$R \leq 15$	4	4	4	4
$R \leq 1.000$	5	7	9	15
$R \leq 500.000$	5	8	15	20