

Alumbrado Público

Laboratorio Semana 12

1 Introducción

La crisis económica afecta a todos, incluso las finanzas públicas. Para reducir los costos de operación, El Gobierno ha decidido optimizar el alumbrado público. En su diseño original, todas las calles se iluminaban toda la noche, lo cual cuesta 1 Unidad Tributaria por metro por día. Para ahorrar dinero, deciden ya no iluminar cada calle, sino apagar el alumbrado de algunas calles. Para asegurarse de que los habitantes se sientan seguros, deben optimizar la iluminación de tal manera que, luego de apagar algunas calles en la noche, seguirá habiendo al menos un camino iluminado de cada sector a cada otro.

¿Cuál es la cantidad máxima de dinero diario que El Gobierno puede ahorrar, sin hacer que sus habitantes se sientan inseguros?

2 Requerimientos del programa

El programa se debe poder ejecutar desde la consola con el siguiente comando:

```
>java Alumbrado <instancia>
```

donde `<instancia>` es el nombre del archivo que contiene el grafo

2.1 Archivo de entrada

Cada caso comienza con dos números m y n : el número de sectores y el número de calles, respectivamente. La entrada se termina por $m, n = 0$. En caso contrario, $1 \leq m \leq 200\,000$ y $m - 1 \leq n < 200\,000$.

Luego siguen n tripletas x, y, z especificando que habrá una calle bidireccional entre x y y con longitud de z metros ($0 \leq x, y < m$ y $x \neq y$), especificando el grafo del caso de prueba. La longitud total de todas las calles en cada caso de prueba es menor que 2^{31} .

2.2 Entradas de ejemplo

```
7 11
0 1 7
0 3 5
1 2 8
1 3 9
1 4 7
2 4 5
3 4 15
3 5 6
4 5 8
4 6 9
5 6 11
0 0
```

Respuesta esperada: 51 UT ahorradas

```
3 2
0 1 2000000
1 2 1
0 0
```

Respuesta esperada: 0 UT ahorradas

3 Requerimientos de la entrega

Debe entregar, a través del Moodle antes del viernes, 14 de noviembre a las 11:55 pm. un archivo comprimido que debe contener los archivos fuentes del código de su solución. Sólo deberá efectuar una entrega por grupo.

4 Evaluación

De ser escogido para evaluación, se asignarán:

- 5 puntos por código
 - 2 puntos por su algoritmo de árbol mínimo cobertor
 - 1 punto por calcular el costo total del grafo original correctamente
 - 1 punto por calcular el costo del árbol mínimo cobertor correctamente
 - 1 punto por su lectura de archivo
- 2 puntos por ejecución
 - 1 punto por calcular el ahorro correcto en los ejemplos mostrados
 - 1 punto por calcular el ahorro correcto en un grafo de 200 calles
- 3 punto por documentación

El programa debe correr sin errores.