Rutas Baratas X81287_es

Hemos recopilado abundante información sobre las carreteras locales y alojamientos de una cierta región que queremos visitar. Nuestro plan es ir de una ciudad A a otra ciudad B, gastando la menor cantidad de dinero posible. Para toda carretera que conecta dos ciudades u y v sabemos el coste $\omega(u,v) = \omega(v,u)$ de viajar por dicha carretera (peajes, gasolina, comidas durante el viaje, ...). Cada vez que viajamos de una ciudad u a una de sus vecinas v debemos parar en v y hacer noche; sabemos los costes $\omega'(v)$ de pernoctar para todas las ciudades v (el coste añadido por A y B a nuestra ruta es 0, ya que son los puntos de origen y de destino). Todos los costes, de vértices y de aristas, son no negativos. Por lo tanto el coste de la ruta

$$P = [A, v_1, \ldots, v_n, B]$$

es

$$\text{coste}(P) = \omega(A, v_1) + \omega(v_1, v_2) + \ldots + \omega(v_n, B) + \omega'(v_1) + \ldots + \omega'(v_n).$$

Escribe un programa en C++ que, dados un garfo no dirigido con pesos no negativos en vértices y en aristas, y dos vértices A y B, devuelve el coste de la ruta más barata para ir de A a B, o una indicación de que no existe tal ruta.

Entrada

Todos los datos de entrada son enteros no negativos. La entrada comienza con dos enteros $2 \le n \le 10000$ y m, $0 \le m \le 20n$. A continuación, viene una secuencia de n enteros no negativos $\omega'(0), \ldots, \omega'(n-1)$, los pesos $\omega'(u)$ de los n vértices del grafo. Luego viene una secuencia con las m aristas del grafo en forma de tripletas $\langle u, v, \omega(u,v)\rangle$. Los vértices u y v son enteros en el rango $\{0, \ldots, n-1\}$ y los pesos $\omega(u,v)$ son enteros no negativos. Puede asumirse que no hay aristas paralelas diferentes uniendo un mismo par de vértices y que no hay ninguna arista que une a un vértice consigo mismo. Finalmente, la entrada contiene una secuencia de pares $\langle A_i, B_i\rangle$, donde los A_i's y los B_i's denotan vértices del grafo ($0 \le A_i, B_i < n$).

Salida

Para cada par $\langle A_i, B_i\rangle$ de la entrada, el programa escribe el coste δ de la ruta más barata entre A_i y B_i con el formato `c(`A_i`,`B_i`) = `δ. Si no hay rutas entre A_i y B_i el programa escribe `c(`A_i`,`B_i`) = +oo`. Cada línea de la salida termina con un salto de línea (`endl`).

Ejemplo de entrada

```
6 8
3 6 10 15 5 2
0 1 2   1 2 7   2 3 2
0 2 1   1 3 4   2 4 8
3 4 2   3 0 5
0 4
1 4
2 4
3 1
4 1
2 5
2 2
```

Ejemplo de salida

```
c(0,4) = 19
c(1,4) = 21
c(2,4) = 8
c(3,1) = 4
c(4,1) = 21
c(2,5) = +oo
c(2,2) = 0
```

Información del problema

Autor : Conrado Martinez
Traductor : Conrado Martinez
Generación : 2018-11-28 18:43:54