

Hem recollit abundant informació sobre les carreteres locals i allotjaments d'una regió que ens proposem recórrer. El nostre objectiu és anar de la ciutat A a la ciutat B i ens agradaria fer el mínim de despeses. Per a cada carretera que connecta dues ciutats u i v sabem el cost $\omega(u, v) = \omega(v, u)$ de viatjar per aquesta carretera (peatges, gasolina, àpats durant el camí, ...). Cada cop que anem d'una ciutat u a una de les seves veïnes v haurem de fer nit; també sabem el cost $\omega'(v)$ d'aturar-se per fer nit a cada ciutat v (el cost afegit per A i B al cost de la nostra ruta és 0, doncs són els punts d'origen i de destí).

Tots els costos, tant del vèrtexos com de les arestes, són no negatius. Així doncs, el cost de la ruta

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

és

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

Escriu un programa en C++ que, donat un graf no dirigit amb pesos no negatius als vèrtexos i les arestes i dos vèrtexos A i B , calcula el cost de la ruta més barata per anar d' A a B , o bé ens indica que no existeix cap ruta.

Entrada

Totes les dades d'entrada són enters no negatius. L'entrada comença amb dos enters $2 \leq n \leq 10000$ i m , $0 \leq m \leq 20n$. A continuació, vindrà una seqüència d'enters no negatius $\omega'(0), \dots, \omega'(n-1)$ dels costos $\omega'(u)$ dels n vèrtexos del graf. Després l'entrada conté la seqüència d'arestes del graf en forma de tripletes $\langle u, v, \omega(u, v) \rangle$. Els vèrtexos u i v són enters en el rang $\{0, \dots, n-1\}$ i els costos $\omega(u, v)$ són enters no negatius. Pots assumir que no hi haurà arestes diferents connectant un mateix de vèrtexos ni cap aresta que connecta un vèrtex amb si mateix.

Finalment, l'entrada conté una seqüència de parells $\langle A_i, B_i \rangle$, on A_i i B_i són vèrtexos del graf ($0 \leq A_i, B_i < n$).

Sortida

Per a cada parell $\langle A_i, B_i \rangle$ de la seqüència d'entrada, el programa imprimeix el cost de la ruta més barata entre A_i i B_i . Si no hi ha cap ruta entre A_i i B_i el programa escriu $c(A_i, B_i) = +\infty$. Per a cada cas s'escriu una línia acabada amb un salt de línia (`endl`).

Exemple d'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
```

Exemple de sortida

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

Informació del problema

Autor : Conrado Martinez

Traductor : Conrado Martinez

Generació : 2018-11-28 18:43:50

© *Jutge.org*, 2006–2018.

<https://jutge.org>