
El Segon Camí**X77211_ca**

Sabem calcular el camí de cost mínim entre dos vèrtexos qualsevol (si existeix), en un graf dirigit etiquetat amb nombres positius, gràcies a l'**algorisme de Dijkstra**, que tots hauríem de conèixer.

En aquest problema volem calcular el cost del **segon** camí de cost mínim, de manera que aquest camí no comparteixi *cap* aresta amb el camí de cost mínim original, que s'ha calculat sense cap restricció.

Així doncs, el problema és, donats:

- un graf G dirigit i etiquetat (amb nombres enters positius),
- dos vèrtexos s i e ,

Feu una funció **segon.millor.camí** (G, s, e) que calculi el cost del camí de cost mínim a G entre s i e , i que no passa per *cap* aresta del conjunt d'arestes E_{min} (on E_{min} és el conjunt d'arestes del camí de cost mínim a G entre s i e , calculat sense cap restricció).

Precondició

G és un graf dirigit i etiquetat amb nombres enters positius. Si N és el nombre de vèrtexos de G , $0 \leq s < N$ i $0 \leq e < N$

Entrada

L'entrada al programa serà el graf G i els vèrtexos s i e .

Primer tenim un nombre n , que és el nombre de vèrtexos del graf (anomenats, per tant, amb els nombres $0 \dots n - 1$), seguit d'un nombre m , que és el nombre d'arestes del graf. Tot seguit tenim m grups de tres nombres u, v, w , significant l'aresta entre u i v amb etiqueta w . És clar que $0 \leq u, v < n$, i que $w > 0$. Finalment tenim dos nombres s i e , que són els vèrtexos inicial i destinació respectivament, i $0 \leq s, e < n$.

Vegeu els exemples que formen el joc de proves públic.

Observacions

Heu de baixar-vos el fitxer **code.py** (icona de la serp). Aquest fitxer és un programa amb **tot** el que cal per executar els jocs de prova públics. Només falta, clar, la funció que us demana l'enunciat. Aquest fitxer l'heu de completar amb el codi que falta, i això, **tot**, és el que heu d'enviar al Jutge com a solució.

Dins el fitxer **code.py** teniu una versió modificada de l'algorisme de Dijkstra. La modificació està pensada per fer senzilla la solució a aquest problema. Està convenientment comentat, però entendre i saber utilitzar aquesta versió modificada de Dijkstra forma part de la solució d'aquest problema.

També podeu trobar la funció per llegir grafs que ja heu fet servir diversos cops a les sessions de laboratori. No cal, per tant, que vosaltres us amoïneu per llegir l'entrada. Us ho donem fet.

L'eficiència i la qualitat de la solució es tindran en compte a la correcció manual.

Exemple d'entrada 1

```
5 7
0 1 2
0 2 5
1 2 1
1 3 4
2 3 2
2 4 3
3 4 2
0 4
```

Exemple d'entrada 2

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

Exemple d'entrada 3

```
6 10
1 0 6
1 5 15
3 4 3
3 1 8
4 0 20
0 5 5
0 2 1
5 1 10
4 1 2
2 3 4
3 5
```

Exemple d'entrada 4

```
2 1
0 1 1000
1 0
```

Exemple d'entrada 5

```
3 3
0 2 100
0 1 40
1 2 60
0 2
```

Exemple de sortida 1

9

Exemple de sortida 2

-1

Exemple de sortida 3

23

Exemple de sortida 4

-1

Exemple de sortida 5

100

Informació del problema

Autor : Jordi Delgado

Generació : 2025-04-02 09:19:28

© *Jutge.org*, 2006–2025.

<https://jutge.org>