

Problem D. Camino más corto

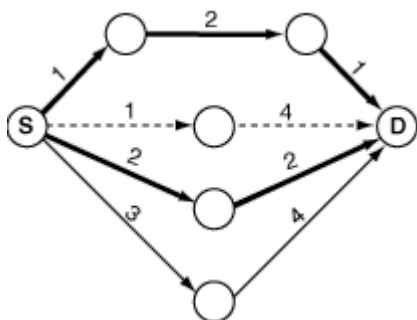
Time limit	1000 ms
Mem limit	1572864 kB
Code length Limit	50000 B
OS	Linux

Encontrar el camino más corto desde un nodo a otro dado un grafo es un problema conocido y es parte de nuestro día a día, los algoritmos que resuelven esto ya están disponibles en todas partes.

A la mayoría de la gente le encantan estas aplicaciones y las usan para hacer sus vidas más fáciles. Hoy en día todos tienen acceso a GPS y son capaces de calcular los caminos más cortos, pero por esto los caminos están congestionados y son más lentos de lo normal.

Con esto en mente, estás desarrollando una aplicación que te permita calcular no el menor camino, pero el casi mejor camino. Defines el casi mejor camino como el camino más corto que no ocupe ninguna arista que sea parte de algún camino más corto. Esto es equivalente al camino más corto en un grafo donde se eliminan todas las aristas que son parte de algún camino más corto.

Por ejemplo, supongamos que la figura de abajo representa el mapa dado. El punto de partida está marcado con una S y el destino con una D. Las líneas en negrita son parte de un camino más corto (en este caso hay dos caminos más cortos, cada uno con tamaño 4). Entonces, el casi mejor camino es el que está en la línea punteada (tamaño 5) ya que no tiene ninguna arista que pertenezca a un menor camino. Notar que podría haber más de una respuesta o incluso no haber respuesta.



Entrada

El input contiene varios casos de prueba. La primera línea de un caso de prueba contiene dos enteros N ($2 \leq N \leq 500$) y M ($1 \leq M \leq 10^4$), separados por un solo espacio, indicando el número de puntos en el mapa y el número de rutas unidireccionales que existen respectivamente. Cada nodo

es identificado con un número entre 0 y $N - 1$. La segunda línea contiene dos enteros S y D , separados por un solo espacio, indicando respectivamente la partida y el destino. ($S \neq D$; $0 \leq S, D < N$).

Cada una de las siguientes M líneas contiene tres enteros U , V y P ($U \neq V$; $0 \leq U, V < N$; $1 \leq P \leq 10^3$), separados por un solo espacio, indicando la existencia de un camino que va de U a V con distancia P . Hay a lo más una ruta desde un punto U a otro punto V , pero nota que la existencia de una ruta desde U a V no implica que haya una ruta desde V a U , y, si tal ruta existiera, puede tener distinta distancia. El fin del input es indicado por una sola línea que contiene dos ceros.

Salida

Por cada caso de prueba en el input, tu programa debe imprimir una sola línea que va a ser -1 si no es posible cumplir los requisitos, o un entero que represente la distancia del casi mejor camino encontrado.

Ejemplo

Input	Output
7 9 0 6 0 1 1 0 2 1 0 3 2 0 4 3 1 5 2 2 6 4 3 6 2 4 6 4 5 6 1 4 6 0 2 0 1 1 1 2 1 1 3 1 3 2 1 2 0 3 3 0 2 6 8 0 1 0 1 1 0 2 2 0 3 3 2 5 3 3 4 2 4 1 1 5 1 1 3 0 1 0 0	5 -1 6