

Semana 14 Python

L14A_16-1

[Problema](#)

[Entrada](#)

[Salida](#)

L14B_16-1

[Problema](#)

[Entrada](#)

[Salida](#)

Semana 14 Python

L14A_16-1

Tiempo límite: 1 segundo

Descripción del problema

Un grupo de ratones de laboratorio está siendo entrenado para escapar de un laberinto. El laberinto se compone de celdas, y cada celda está conectada a algunas otras celdas. Sin embargo, hay obstáculos en el paso entre las celdas y por lo tanto, hay una penalización de tiempo para superar el paso. Además, algunos pasajes permiten a los ratones ir en un solo sentido.

Supongamos que todos los ratones son entrenados y ahora, cuando se coloca en una celda arbitraria en el laberinto, toman un camino que les lleva a la celda de salida en un tiempo mínimo.

Vamos a llevar a cabo el siguiente experimento: Se coloca un ratón en una celda del laberinto, Su tarea consiste en decir el tiempo que tarda el ratón en llegar a la salida.

Entrada

La entrada comienza con un único entero positivo que indica el número de casos de prueba.

La segunda línea de entrada contiene tres números: el número de celdas en el laberinto, el id de la celda donde se coloca el ratón y el id de la celda de salida respectivamente. La tercera línea contiene el número de conexiones en el laberinto, y es seguido por M líneas, cada una especificando una conexión con tres números enteros que indica un camino entre la celda u hasta la celda v con un tiempo w.

Tenga en cuenta que cada conexión es unidireccional, es decir, los ratones no pueden viajar de v a u menos que haya otra línea que especifica ese pasaje. Observe también que el tiempo requerido para viajar en cada dirección puede ser diferente.

Semana 14 Python

Salida

Por cada caso de prueba se debe imprimir la distancia mínima que tarda el ratón en llegar a la salida. Si no es posible llegar se debe imprimir -1.

Ejemplo de Entrada	Ejemplo de Salida
2 4 0 1 8 0 1 1 0 2 1 1 0 1 1 3 1 2 0 1 2 3 1 3 1 1 3 2 1 5 2 0 6 0 2 7 0 1 2 1 3 1 3 2 2 4 1 3 4 3 6	Case #1: 1 Case #2: -1

Semana 14 Python

L14B_16-1

Tiempo límite de ejecución: 2 segundos

Problema

Villa Algoritmo es una pequeña ciudad con una interesante planeación urbana, no hay calles ni carreras bien definidas, pero si cuentan con una serie de senderos que comunican cada casa entre sí. Algunas casas tienen senderos a todas sus vecinas, otras no necesariamente a todas, también hay algunas casas muy nuevas que aún no tienen ningún sendero.

En Villa Algoritmo no hay conexión a internet y sus habitantes aún se comunican con cartas. El cartero debe repartir las cartas en bicicleta y él siempre busca terminar su trabajo lo más rápido posible. Su problema es saber cual es el camino más corto entre dos casas, es decir el camino con menor Kilometraje. ¿Puedes ayudarlo?

Entrada

La primera línea contiene un número que significa la cantidad de casos de prueba.

La primer línea de cada caso contiene 3 enteros **N** , **M** y **Q** que simbolizan el número de casas, el número de senderos y el número de consultas respectivamente.

Las siguientes **M** líneas de entrada contienen los senderos de la forma "**u v k**", que simboliza que existe un sendero directo que conecta la casa **u** con la casa **v** y la distancia que las separa es **k**.

Las siguientes **Q** líneas contiene la casa inicial **q** y la casa de destino **d**.

Restricciones/Consideraciones

Los nodos tienen valores entre 0 y **N-1**.

$$1 \leq N \leq 100$$

$$0 \leq q$$

$$0 < k < 10000$$

Salida

Para cada consulta se debe calcular la distancia mínima que existe desde el nodo origen **q** al nodo destino **d**. Si no existe camino entre los nodos imprimir -1

Semana 14 Python

Ejemplo de Entrada	Ejemplo de Salida
2 5 4 3 0 1 1 1 2 1 2 3 1 3 4 1 0 4 0 2 1 2 7 4 4 0 5 1 0 5 1 4 2 1 5 6 1 6 0 4 2 0 6 6 2	4 2 1 2 1 2 -1