Introducción

El primer semáforo del mundo se remonta a 1868. Se instaló en Londres para controlar

trá forco para ... vehículos tirados por caballos! Hoy en día, las luces de tráfico se pueden encontrar en la calle.

intersecciones en casi todas las ciudades del mundo, lo que hace que sea más seguro para los vehículos circular

a través de ellos.

Los semáforos tienen al menos dos estados y usan un color (generalmente rojo) para indicar "alto",

y otro (generalmente verde) para indicar que los autos pueden pasar. El primero

las luces de tráfico se controlaban manualmente. Hoy en día son automáticos, lo que significa que

Deben diseñarse y cronometrarse cuidadosamente para optimizar el viaje en general.

tiempo para todos los participantes en tráco.

Tarea

Dada la descripción de un plano de la ciudad y los caminos planificados para todos los automóviles en esa ciudad,

optimizar el horario de las luces de tráco para minimizar la cantidad total de tiempo

tráco, y ayudar a tantos coches como sea posible a llegar a su destino antes de una determinada

plazo.

Descripción del problema

Plano de la ciudad

El plan de la ciudad consta de calles e intersecciones de un solo sentido. Cada calle:

● se identifica por un nombre único,

● conduce de una intersección a otra,

● no contiene intersecciones intermedias (si es necesario cruzar dos calles

fuera de una intersección, se utiliza un puente o túnel),

● tiene una cantidad fija de tiempo L que tarda un automóvil en llegar desde el comienzo de la

calle hasta el final. Si se necesitan L segundos para atravesar una calle y un automóvil

entra en el tiempo T llegará al final de la calle precisamente en T + L,

independientemente de cuántos coches haya en la calle.

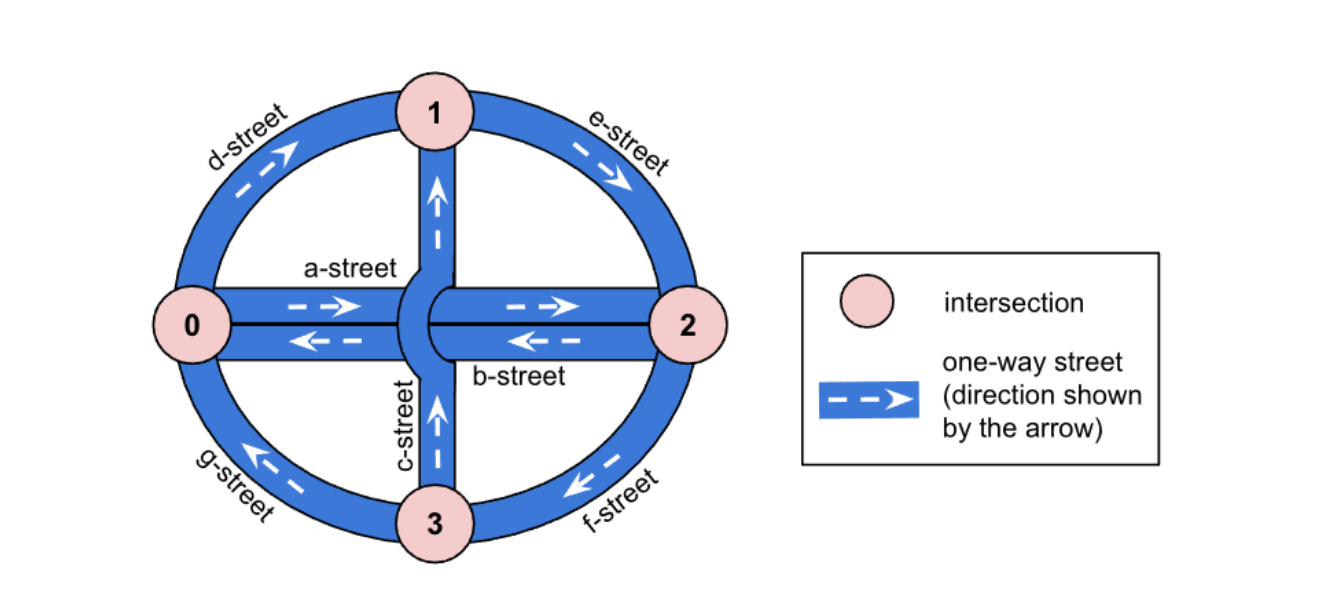


Figura 1. Plano de una ciudad con 4 intersecciones (0, 1, 2 y 3) y 7 calles (a, b, ...,

g-street). Las intersecciones 0 y 2 están conectadas directamente en ambos sentidos a través de una calle

y b-street. c-street usa un puente sobre a-street y b-street y no

se cruzan con esas dos calles.

Tenga en cuenta que si bien las calles son unidireccionales, es posible tener dos

calles que conectan dos intersecciones en direcciones opuestas (ver intersecciones 0 y

2 y a-street y b-street en la Figura 1). Sin embargo, nunca habrá dos calles

conectando dos intersecciones en la misma dirección.

Cada intersección:

● tiene un ID entero único (por ejemplo, 0, 1, 2 ...),

● tiene al menos una calle que entra y al menos una calle que sale

de ella.

Luces de traƻc

Hay un semáforo al final de cada calle (justo antes de la intersección). Cada

El semáforo tiene dos estados: una luz verde indica que los automóviles de esa calle pueden

cruce la intersección y diríjase hacia cualquier otra calle, mientras que una luz roja indica

que los autos de esa calle deben detenerse. A lo sumo, un semáforo estará verde en

cada intersección en un momento dado. Mientras que la luz es verde para una calle entrante,

solo los autos de esta calle podrán ingresar a la intersección (y moverse a cualquier

calle de salida), todos los demás coches tienen que esperar.

Haciendo fila

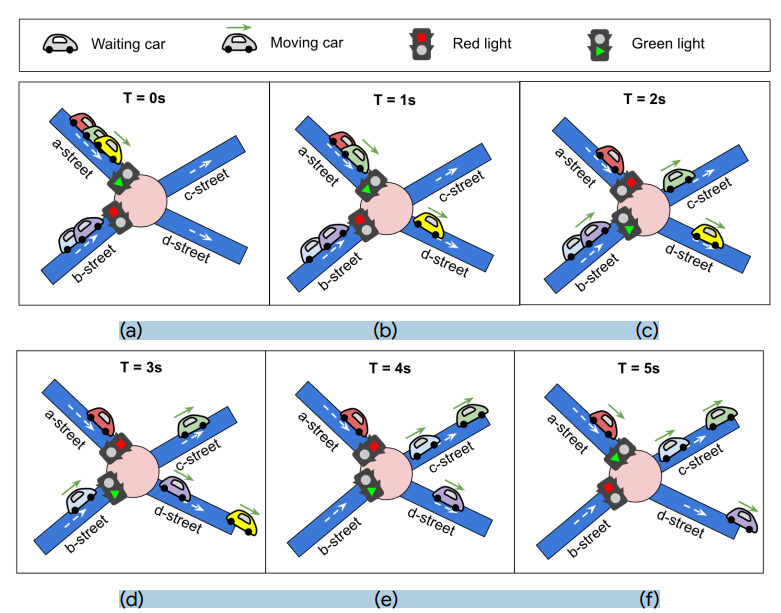
Cuando el semáforo al final de una calle está en rojo, los autos que llegan hacen cola esperando el

luz para ponerse verde. Cuando la luz está en verde, un automóvil puede cruzar la intersección.

cada segundo. Esto significa que si una luz verde para una calle determinada dura Ti

segundos, entonces solo los primeros autos Ti de esa calle continuarán su viaje (ver

Figura 2). Otros deberán esperar la siguiente luz verde.



(a B C)

(d) (e) (f)

Figura 2. Considere una intersección con dos calles entrantes (una calle con 3 carros esperando

y calle b con 2 carros esperando), y dos calles salientes (calle c y calle d). Allí

Hay dos semáforos, uno al final de una calle con T1 = 2s y otro al final de la

b-street con T

2 = 3 s. (a) Inicialmente, el semáforo de una calle es verde y el primero (amarillo)

el coche está en movimiento. (b) En T = 1s, el siguiente automóvil (verde) de una calle se mueve. (c) En T =

2s, a-street tiene un semáforo en rojo y el último automóvil (rojo) que espera allí debe detenerse. Al mismo

tiempo, el semáforo de b-street se pone verde y el primer automóvil (púrpura) en la cola

Moviente. (d), (e) En T = 3s y T = 4s, los dos autos (violeta y azul) que estaban inicialmente en

b-street ya ha pasado el semáforo y ha continuado su camino. (f) En T = 5s, la luz

en una calle se vuelve verde y el automóvil (rojo) que estaba esperando allí ahora puede moverse.

Calendario

Para cada intersección podemos establecer un horario de semáforos. El horario del traƻc light

determina el orden y la duración de la luz verde para las calles entrantes del

intersección y se repite hasta el final de la simulación. El horario es una lista

de pares: calle de entrada y duración. Cada calle puede aparecer como máximo una vez en

el horario. El horario puede ignorar algunas de las calles entrantes; esas

nunca obtenga luz verde.

El horario del semáforo está controlado por sus presentaciones. No tienes que hacerlo

especificar el horario de todas las luces de tráco. Por defecto todas las luces en todas las intersecciones

están en rojo (sí, los coches atascados allí tendrán que esperar hasta el final de la simulación).

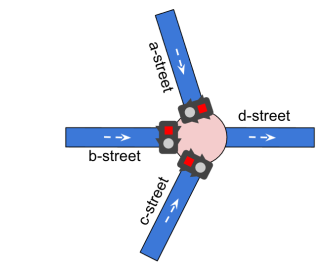
Ejemplos de horarios de luz de tráfico

En el siguiente ejemplo, una intersección tiene 3 calles que conducen a ella (a, by

c-street) y uno que sale de la intersección (d-street). Consideramos tres diferentes

ejemplos de horarios para estas luces de tráco:

Sin horario de luz de tráfico (predeterminado)

Figura 4 (a). Si no se establece un horario de luz de tráco para un

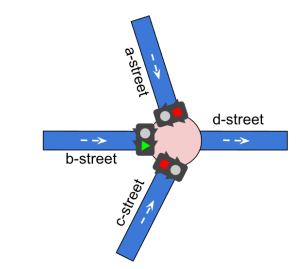
intersección, todas las luces de tráfico permanecen rojas

a lo largo de la simulación. Cualquier coche que sea

programado para pasar por la calle a, b o c será

bloqueado y no llegar a su destino.

Horario que cubre solo una calle

Figura 4 (b). En este ejemplo, la luz de tráfico

El horario cubre solo una calle entrante

(calle b). En este caso b-street siempre tiene un verde

ligero. ΑCualquier coche que venga de b-street siempre irá

a través de la intersección directamente, mientras que cualquier automóvil

programado para pasar por una calle o

c-street se bloqueará y no llegará a su

destino.

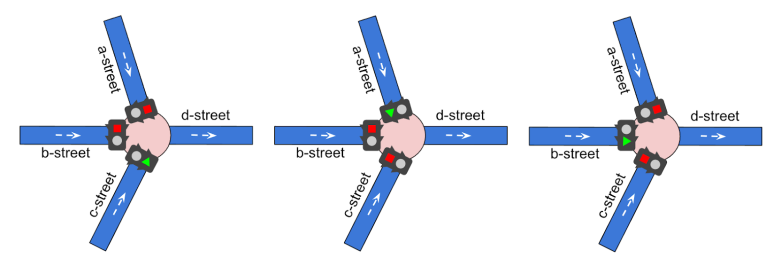


Figura 4 (c). En este ejemplo, el horario del semáforo cubre todas las calles entrantes

(C-street, luego a-street, luego b-street). Para cada calle podemos definir un Ti diferente,

lo que significa una duración diferente durante la cual esa luz de tráfico permanece verde.

Carros

Cada automóvil está descrito por el camino (una secuencia de calles) que va a conducir.

a través de. Las rutas están definidas por los conjuntos de datos de entrada y no se pueden modificar. En el

conjuntos de datos de entrada garantizamos que cada automóvil puede pasar por una sola intersección en

la mayoría una vez.

Inicialmente, todos los coches se quedan al final de la primera calle en su camino, esperando el green.

luz (en caso de que la luz de trá isco esté roja), o lista para moverse (si es verde). Si dos coches se quedan

al final de la misma calle, el automóvil que aparece primero en el archivo de entrada va primero. Cada carro

luego sigue un camino dado, mientras obedece a las luces de tráco, y necesita llegar al

final de la última calle de ese camino.

Los coches hacen cola al final de cada calle. El primer coche de la cola puede cruzar

la intersección inmediatamente después de que la luz se ponga verde. No hay demora mientras un auto

pasa por una intersección. Autos después que cruzan la intersección uno después

otro, un coche por segundo.

Cuando un automóvil entra en la última calle de su camino, completa su recorrido hasta el final de

la calle y luego se retira inmediatamente de ella. Esto significa que el coche no

no hace cola al final de la última calle de su camino y no entra en el

intersección al final de la misma.

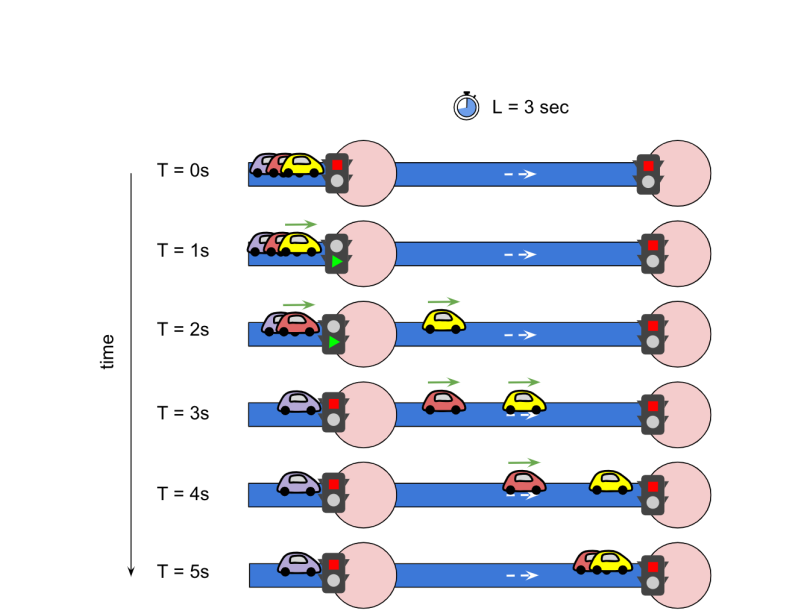


Figura 3. Esta ǁgura muestra el estado de tres coches exactamente en T segundos, con la simulación

empezando por T = 0s y terminando en T = 5s. La calle tiene L = 3 s, lo que significa que cualquier automóvil necesita 3

segundos para ir desde el principio hasta el final. La luz se vuelve verde en la intersección de la izquierda en

T = 1s y se vuelve rojo nuevamente 2 segundos después. Los coches hacen cola al final de la calle

siendo el amarillo el primero. En T = 1s, el primer automóvil (amarillo) se pone inmediatamente en movimiento

y llega al final de la calle 3 segundos después, en T = 4s. El segundo coche (rojo) tiene que

espere 1 segundo antes de que se ponga en movimiento y llegue al final de la calle 3 segundos después, en

T = 5 s. El tercer automóvil (violeta) no tuvo tiempo suficiente para entrar a la calle, ya que el semáforo

ya se puso rojo. Tenga en cuenta que esta ǁgura solo muestra dos calles para simplificar: cuando un tráƻco

la luz se muestra en rojo, esto significa que otra en la misma intersección ha girado

verde.

Conjunto de datos de entrada

Formato de archivo

Cada conjunto de datos de entrada se proporciona en un archivo de texto sin formato. El archivo contiene solo ASCII

caracteres con líneas que terminan con un solo carácter '\ n' (también llamado línea "estilo UNIX"

finales). Cuando se dan varios números en una línea, están separados por un

espacio simple entre cada dos números.

● La primera línea contiene cinco números:

un número entero D (1 ≤ D ≤ 104) - la duración de la simulación, en segundos,

○ un entero I (2 ≤ I ≤ 105) - el número de intersecciones (con ID desde 0

a I-1),

○ un entero S (2 ≤ S ≤ 105) - el número de calles,

○ un número entero V (1 ≤ V ≤ 103) - el número de coches,

○ un entero F (1 ≤ F ≤ 103): los puntos de bonificación por cada automóvil que alcance

su destino antes del tiempo D.

● Las siguientes líneas S contienen descripciones de calles. Cada línea contiene:

○ dos enteros B y E (0 ≤ B <I, 0 ≤ E <I) - las intersecciones en la staǈ

y el final de la calle, respectivamente,

○ el nombre de la calle (una cadena que consta de entre 3 y 30 minúsculas

Caracteres ASCII a-z y el carácter -),

○ un entero L (1 ≤ L ≤ D): el tiempo que tarda un automóvil en llegar desde el

desde el principio hasta el final de esa calle.

● Las siguientes líneas V describen los caminos de cada automóvil. Cada línea contiene:

○ un entero P (2 ≤ P ≤ 103) - el número de calles que el automóvil quiere

viaje,

○ seguido de P nombres de las calles: el coche está al final de la

Primera calle (es decir, espera la luz verde para pasar a la siguiente calle)

y sigue el camino hasta el final de la última calle. El camino de un coche es

siempre válido, es decir, las calles estarán conectadas por intersecciones.

Ejemplo

6 4 5 2 1000

2 0 rue-de-londres 1

0 1 rue-d-amsterdam 1

3 1 rue-d-athenes 1

2 3 rue-de-rome 2

1 2 rue-de-moscou 3

4 rue-de-londres rue-d-amsterdam

rue-de-moscou rue-de-rome

3 rue-d-athenes rue-de-moscou

rue-de-londres

La simulación dura 6 segundos, hay 4

intersecciones, 5 calles y 2 autos; y un carro

obtiene 1000 puntos por llegar al destino

a tiempo.

Street rue-de-londres comienza en la intersección 2,

termina en 0, y tarda L = 1 segundos en pasar de

desde el principio hasta el final.

La calle rue-d-amsterdam comienza en la intersección 0

termina en 1 y tiene L = 1.

La calle rue-d-athenes comienza en la intersección 3,

termina en 1 y tiene L = 1.

La calle rue-de-rome comienza en la intersección 2,

termina en 3 y tiene L = 2.

La calle rue-de-moscou comienza en la intersección 1,

termina en 2 y tiene L = 3.

El primer coche empieza al final de

rue-de-londres y luego sigue el camino indicado.

El segundo coche comienza al final de

rue-d-athenes y luego sigue el camino indicado.

Tenga en cuenta que el archivo de entrada no contiene líneas en blanco. Líneas y líneas en blanco

los envoltorios en el ejemplo anterior se agregan para mayor claridad.

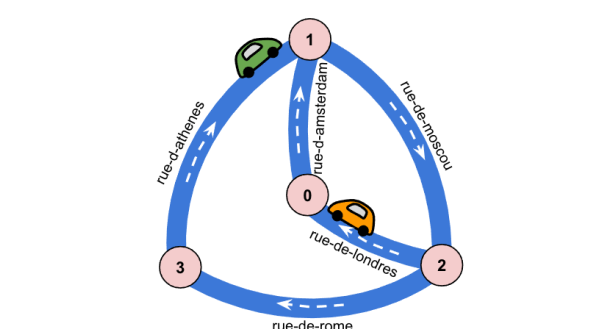


Figura 5. Las calles y las intersecciones, como se indica en el conjunto de datos de entrada de ejemplo, como

así como los dos coches en sus posiciones iniciales.

Envíos

Su envío describe los horarios de luces de tráfico que desea establecer para

intersecciones.

Formato de archivo

La primera línea debe contener un solo entero A (0 ≤ A ≤ I), el número de intersecciones

para el que especifica el horario.

Luego, el archivo de envío debe describir los horarios de intersección, en cualquier orden.

Cada horario debe estar descrito por varias líneas:

● la primera línea que contiene un solo entero i (0 ≤ i <I) - el ID del

intersección,

● la segunda línea que contiene un solo entero Ei (0 <Ei) - el número de

calles entrantes (de la intersección i) cubiertas por este horario,

● E

i líneas que describen el orden y la duración de las luces verdes. Cada una de esas líneas

debe contener (separados por un solo espacio):

○ el nombre de la calle,

○ un número entero T (1 ≤ T ≤ D) - por cuánto tiempo cada calle tendrá un verde

ligero.

Cada intersección solo se puede enumerar una vez en el archivo de envío. Y cada calle

El nombre solo puede aparecer una vez por horario. Si el nombre de la calle no está presente en

conǁguración de intersección significa que su luz de tráfico siempre es roja. Si una intersección

la configuración no está presente en el archivo de presentación, entonces todas sus luces de tráco están

siempre rojo.

Ejemplo

3 1 2 r

ue-d-athenes 2

rue-d-amsterdam 1

0 1 r

ue-de-londres 2

2 1 r

ue-de-moscou 1

Hay 3 intersecciones con horarios de semáforos:

En la intersección 1, los semáforos están en verde para

2 calles entrantes diferentes, a saber

rue-d-athenes durante 2 segundos, luego verde durante

rue-d-amsterdam durante 1 segundo, luego nuevamente rue-d-athenes.

En la intersección 0, los semáforos están en verde para

1 calle de entrada solamente, a saber

rue-de-londres durante 2 segundos por ciclo (siempre es verde

para rue-de-londres).

En la intersección 2, los semáforos están en verde para

1 calle de entrada solamente, a saber

rue-de-moscou durante 1 segundo por ciclo (siempre es verde para

rue-de-moscou).

Tenga en cuenta que el archivo de entrada no contiene líneas en blanco. Líneas y líneas en blanco

los envoltorios en el ejemplo anterior se agregan para mayor claridad.

Puntuación

Se otorga una puntuación por cada automóvil que finaliza su camino antes del final del

simulación. Para un automóvil que termina su camino en el momento T, la puntuación otorgada es (F) puntos

(Premio fijo por terminar el camino) y adicionalmente (D - T): un punto por cada

segundo a la izquierda cuando el coche terminó el camino.

En otras palabras: si un automóvil termina en el tiempo T, puntúa

● F + (D - T) puntos si T ≤ D,

● o 0 puntos en caso contrario.

La puntuación de la solución es la suma de las puntuaciones de todos los automóviles.

Ejemplo

Por ejemplo, en el ejemplo anterior, el primer automóvil:

● T = 0s: cruza inmediatamente la intersección 0, ya que el semáforo siempre hay

verde,

● T = 1s: un segundo después, ha pasado por rue-d-amsterdam. Sin embargo,

la luz de trá isco es roja (en cuanto a la intersección 1, la presentación ha establecido el

duración para que la luz de rue-d-athenes sea verde durante 2 segundos),

● T = 2s: el automóvil ahora cruza la intersección y continúa hacia rue-demoscou,

● T = 5 s: el coche ha llegado al final de la rue-de-moscou, encuentra una luz verde en

intersección 2, la cruza y continúa hasta rue-de-rome.

Este primer coche habría llegado al final de la rue-de-rome en T = 7s, pero ya pasó

la fecha límite de la ejecución (definida en el conjunto de datos de entrada). Lo que significa que 0 puntos son

asignado a este coche.

El segundo carro:

● T = 0s: encuentra un semáforo en verde en la intersección 1 y continúa hacia rue-de-moscou,

● T = 3s: llega al final de rue-de-moscou, encuentra una luz verde en la intersección

2 y no hay autos esperando, por lo que inmediatamente cruza la intersección y se dirige

hacia rue-de-londres,

● T = 4s: el coche llega al final de rue-de-londres, que es su destino.

Entonces, el segundo automóvil termina antes de la fecha límite y obtiene una puntuación de 1000 + (6 - 4) =

1002 puntos.

La puntuación final de esta presentación es 1002.

Tenga en cuenta que hay varios conjuntos de datos que representan instancias separadas del

problema. El puntaje final de su equipo será la suma de sus mejores puntajes para

los conjuntos de datos individuales.