

Google Paris Hash Code

FR

Énoncé du problème pour le round principal (2014-04-05)

Routage Street View



1. Introduction

Les photographies Street View qui sont disponibles dans Google Maps sont réalisées en utilisant des véhicules spécialisés appelés "[Street View cars](#)". Ces véhicules transportent plusieurs appareils photographiques qui prennent des photographies au fur et à mesure des déplacements des véhicules dans une ville.

La réalisation des photographies d'une ville pose un problème d'optimisation : la flotte de véhicules est disponible pour une période de temps limitée, et on souhaite couvrir autant de rues de la ville que possible.

2. Tâche

Étant donnée une description des rues de la ville et un nombre de véhicules disponibles pour une période de temps, votre tâche est de planifier le mouvement des véhicules pour maximiser la longueurs totales des rues couvertes au moins une fois.

3. Description du problème

3.1. Ville

La ville est représentée par un graphe. Les nœuds représentent les **intersections** et sont connectés par des arêtes représentant les **rues**. Le graphe est une représentation réaliste mais idéalisée d'une certaine ville, les intersections sont associées avec des coordonnées géographiques concrètes.

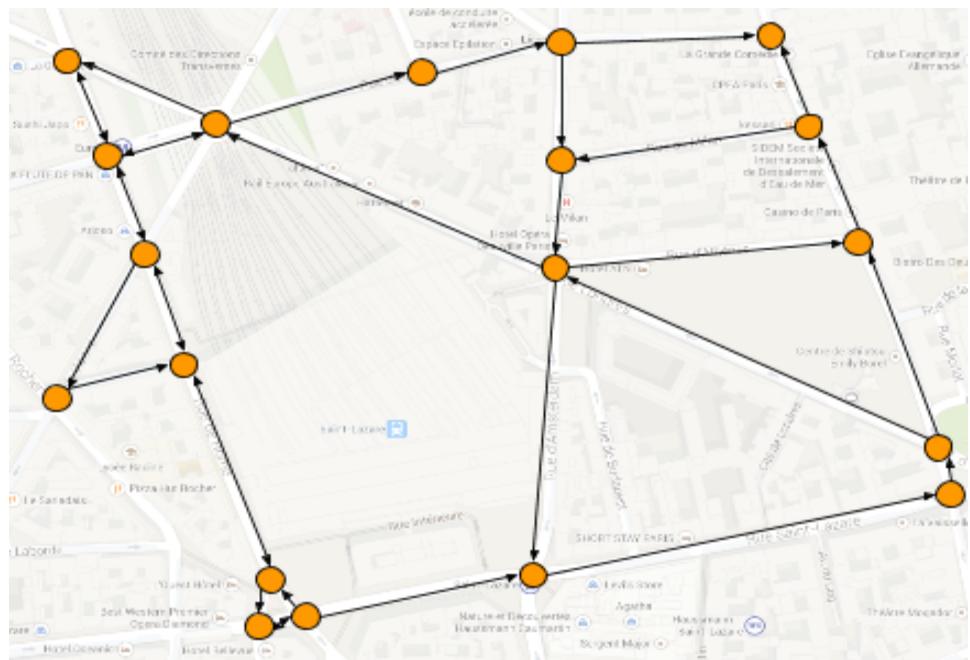


Figure 1: Une partie de la ville

Les **rues** sont modélisées par des segments droits connectant deux intersections. Chaque rue a trois propriétés :

- **direction** - chaque rue peut être à sens unique ou à double sens.
 - **longueur** - la distance en mètres qu'un véhicule StreetView couvre en parcourant cette rue. Cette distance contribue au score, il correspond à la longueur réelle de la rue (y compris les éventuelles courbes).
 - **coût (temps)** - la durée en secondes nécessaire à un véhicule StreetView pour parcourir la rue.

Chaque paire d'intersections est connectée par au plus une rue. Chaque rue connecte deux intersections distinctes. Le graphe n'est pas nécessairement planaire (à cause des ponts et des tunnels).

Les intersections et les rues sont identifiées en utilisant leur indice (en commençant par l'indice 0) correspondant à l'ordre dans lequel elles apparaissent dans le fichier d'entrée (voir ci-dessous).

3.2. Déplacement des véhicules

Votre équipe gère une flotte de **N** véhicules, tous situés à l'intersection **S** au début du jeu.

Les équipes doivent planifier le mouvement des véhicules pendant T secondes - il s'agit du temps virtuel pour le déplacement des véhicules sur la carte, il est indépendant du temps de la compétition. Les équipes ont toute la durée de la compétition pour fournir un itinéraire qui couvre le mouvement des véhicules pour les T secondes virtuelles.

Tous les déplacements planifiés dans les itinéraires doivent se terminer dans les T secondes (ou moins) – un véhicule ne peut être en train de parcourir une rue lorsque ce temps se termine.

Le score d'une équipe est la longueur totale de toutes les rues qui ont été parcourues au moins une fois par une voiture de leur flotte. Parcourir une rue qui a déjà été parcourue (y compris une rue à double sens dans la direction opposée) n'augmente pas le score.

4. Donnée d'entrée

La donnée d'entrée est fournie sous la forme d'un fichier texte contenant seulement des caractères ASCII et dont les lignes sont terminées par des caractères de fin de ligne de style UNIX (un simple caractère '\n' à la fin de chaque ligne).

Le fichier se compose de :

- une ligne contenant les entiers naturels suivants séparés par des espaces simples :
 - **N** : le nombre d'intersections de la ville,
 - **M** : le nombre de rues de la ville,
 - **T** : le temps virtuel autorisé pour le parcours des véhicules,
 - **C** : le nombre de véhicules de la flotte,
 - **S** : l'indice de l'intersection à laquelle les véhicules sont initialement positionnés.
- **N** lignes décrivant les intersections. La i-ème ($0 \leq i < N$) de ces lignes contient les nombres décimaux suivants séparés par des espaces simples et décrivant la i-ème intersection de la ville :
 - **lat**, et **long**, ($-90 \leq lat_i \leq 90$, $-180 \leq long_i \leq 180$) : les coordonnées de l'intersection en degrés décimaux
- **M** lignes décrivant les rues. La j-ème ($0 \leq j < M$) de ces lignes contient les entiers naturels suivants séparés par des espaces simples et décrivant la j-ème rue de la ville :
 - **A_j** et **B_j** ($0 \leq A_j, B_j < N$) ($A_j \neq B_j$) : les indices des intersections connectées par la rue.
 - **D_j** est soit 1 ou 2. Si **D_j** est égal à 1, la j-ème rue est à sens unique et peut être parcourue de l'intersection **A_j** vers l'intersection **B_j**. Sinon (pour **D_j** égal à 2) la rue peut être traversée dans les deux directions.
 - **C_j** : le coût de parcourir la rue, c'est-à-dire le temps en secondes dont un véhicule a besoin pour parcourir la rue.
 - **L_j** : la longueur, en mètres, de la rue. C'est le score attribué lorsque la rue est traversée pour la première fois.

Exemple de fichier d'entrée

```
3 2 3000 2 0    // 3 intersections, 2 rues, 3000 s, 2 véhicules, partant de 0.  
48.8582 2.2945 // Coordonnées de la première intersection.  
50.0 3.09       // Coordonnées de la seconde intersection.  
51.424242 3.02 // Coordonnées de la troisième intersection.  
0 1 1 30 250    // Rue de la première à la deuxième intersection.  
1 2 2 45 200    // Rue de la deuxième à la troisième intersection.
```

5. Soumissions

5.1. Format du fichier

Une soumission doit être décrite par un fichier texte ASCII, avec des fins de ligne au style UNIX ou Windows.

Le fichier doit commencer par une ligne contenant un entier naturel **C** représentant le nombre de véhicules dans la flotte. Les itinéraires de chaque véhicule doivent ensuite être décrits dans le format indiqué ci-dessous.

L'itinéraire pour le i -ème véhicule ($0 \leq i < C$) doit commencer par une ligne contenant un entier naturel V_i ($1 \leq V_i \leq 10^6$) représentant le nombre d'intersections visitées par le véhicule, y compris l'intersection de départ S . Ensuite, l'itinéraire doit contenir V_i lignes décrivant les intersections dans l'ordre dans lequel elles sont visitées par le véhicule. Chaque ligne doit contenir un seul entier naturel correspondant à l'indice (indexé à 0) de l'intersection.

Exemple de fichier de soumission

```
2      // Deux véhicules dans la flotte.  
1      // Le premier véhicule reste à l'intersection de départ :  
0      // - l'intersection de départ.  
3      // Le second véhicule visite 3 intersections :  
0      // - l'intersection de départ,  
1      // - le véhicule se rend de l'intersection 0 à l'intersection 1,  
2      // - le véhicule se rend de l'intersection 1 à l'intersection 2.
```

5.2. Validation

Une solution doit satisfaire les critères suivants pour être acceptée :

- le format du fichier doit respecter la description ci-dessus,
- le nombre de véhicules de la flotte C doit être égal au nombre de véhicules dans le fichier d'entrée,
- la première intersection de chaque itinéraire doit être l'intersection de départ S indiquée dans le fichier d'entrée,
- pour chaque paire de jonctions consécutives dans un itinéraire, une rue doit exister entre ces deux intersections dans le fichier d'entrée,
 - si la rue est à sens unique, elle doit être parcourue dans la bonne direction,
- la durée totale pour chaque itinéraire doit être égale ou inférieure à T ,

5.3. Score

Le score d'une solution est la longueur totale des rues qui sont visitées au moins une fois par au moins un véhicule de la flotte.

Les équipes peuvent soumettre plusieurs solutions - la meilleure solution valide de chaque équipe sera utilisée pour le classement des équipes.

Les équipes seront classées selon leur meilleur score. Dans le cas où deux équipes auraient le même score, l'équipe qui avait atteint ce score en premier sera classée en premier. La soumission répétée d'une même meilleure solution n'affecte pas le classement des équipes.