

INFO–F–203 — Algorithmique 2

Manipulation de graphes : CPS - GPS pour les étudiants

Année académique 2012–2013

Le problème

Un étudiant venant d'arriver à l'université s'est retrouvé devant un des cercles folkloriques de l'ULB. Malheureusement le cercle était fermé et il n'a pas pu y entrer. Il voudrait donc aller découvrir les autres cercles et il ne sait pas par où commencer. Heureusement, il a l'application *CPS* (Cercle Positioning System) dans son smartphone. Cette application lui permet d'indiquer sa position actuelle et lui donne la liste des Cercles les plus proches ainsi que les chemins (avec le temps approximatif) pour y arriver.

Objectif du projet

On vous demande de produire un algorithme efficace pour CPS, qui reçoit en entrée le graphe décrivant les positions des cercles de l'ULB et la position initiale dans ce graphe, et qui produit :

1. la séquence de déplacement la moins coûteuse partant de la position initiale vers chaque cercle ;
2. une séquence de déplacement la moins chère possible partant de la position initiale et visitant tous les cercles exactement une fois ; vu que l'algorithme doit être exécuté sur un smartphone, il doit être rapide et ne renverra donc pas nécessairement la meilleure solution mais une solution approximative.

Détails techniques

Ces algorithmes devront être implémentés en *Java* en suivant au maximum le paradigme Orienté Objet (OO). De plus, les deux algorithmes demandés n'étant pas très différents, il est possible de bien concevoir votre code afin de minimiser le code redondant en pensant aux responsabilités de chaque classe et de chaque méthode et à l'utilité de l'héritage et de la composition. *Un conseil : ne faites jamais de copier-coller.*

Vous pouvez utiliser les structures de données des bibliothèques standard fournies par Java.

Les données du graphe décrivant la ville seront transmises à votre programme via la lecture d'un seul fichier (dont le nom sera passé en argument de votre programme). Le fichier contiendra la description du graphe, composée :

- du nombre de nœuds du graphe (un entier) ;
- du nom de chaque nœud (une chaîne de caractères alphanumériques uniquement), suivi d'un retour à la ligne ;
- d'une liste de triplets "NoeudDeDépart NoeudD'Arrivée Temps" représentant chaque arc du graphe. Chaque nœud est identifié par son nom et le temps est un réel (en minutes).

Voici un exemple de fichier :

```
20
CI
CP
Agro
CdS
CePHA
CD
CM
CPSY
C$
CGEO
CIJ
...
CI CePHA 1
CI CGEO 6.5
CGEO CI 4
CdS CP 5
CI CP 17
CD CP 6.5
...
```

La liste des arcs se finit lorsque le fichier se termine. *Le graphe n'est pas forcément un graphe complet.*

Effectuez le *parsing* dans une ou plusieurs méthodes (pas dans le main) et placez-les dans les classes qui vous paraissent les plus à même d'effectuer ce travail (pensez aussi aux méthodes statiques). Vous pouvez supposer que le fichier d'entrée ne comporte pas d'erreur.

Votre programme doit recevoir le nom du fichier et la position de départ en argument et écrire sur l'output standard la solution de la manière suivante :

```
~>./projet cercles-ULB.txt CI
Plus courts chemins :
CI -> CePHA : 1min
CI -> Agro -> CP : 25min
...
Chemin passant par tous les cercles :
CI -> Agro -> ... -> CP : 350min
```

Utilisez les exceptions dans vos algorithmes en cas de problèmes (et évitez les retours de booléens). Votre code source sera évidemment accompagné d'un *makefile* permettant de compiler facilement votre projet.

Le projet peut être réalisé individuellement ou par groupe de deux (ce que nous recommandons).

Tout cela sera naturellement présenté dans un *rapport* à la présentation soignée. Pour rappel, un rapport est une *texte suivi en français correct* qui *décrit* votre travail. Cela va sans dire, le rapport fait partie *intégrale* de votre travail et une partie importante de la note finale portera sur le rapport (y compris sa présentation).

Consignes pour la remise du projet

À respecter scrupuleusement !

1. Les modalités pour la remise du rapport sont :

- Date : **le 21 Décembre 2012** (vous pouvez ne pas remettre votre projet dans le cas de fin du monde).
- Lieu : **au Secrétariat « étudiants » du Département d’Informatique, local 2N8.104.**
- Heure : **Avant 16h.**

Le secrétariat ferme à 16h. **Après 16h**, les projets sont considérés comme **en retard**, et vous perdez **la moitié des points** sur votre note finale (plus un point par jour de retard). Les projets en retard doivent être déposés **dans la caisse** prévue à cet effet près du secrétariat.

2. Les modalités pour la remise du code sont :

- Tous les fichiers dans **une seule archive** au format **ZIP** (ni tar.gz, ni rar, ni autre chose !).
- L’archive doit porter **le nom** AG2.NomEtudiant1-NomEtudiant2.zip, si deux étudiants ont réalisé le projet ; AG2.NomEtudiant.zip **dans le cas contraire.**
- L’archive doit être envoyée à lporrett@ulb.ac.be et aussi à nveshchi@ulb.ac.be.
- Le **sujet** du mail doit être **exactement** Algo2-Projet2012.

Si vous ne respectez pas ces critères, nous considérerons que vous n’avez pas rendu de code.

Bon travail !