

TDM - Balade en ville

💡 Objectifs

A partir d'une problématique réelle, être capable :

- de faire des choix de modélisation permettant la résolution du problème par des méthodes de RO, en prenant conscience de l'impact de ces choix sur la résolution.
- de déduire le problème formel sous-jacent et sa classe de complexité.
- d'implémenter l'algorithme répondant au problème posé.

Contexte

L'objectif de ce TP est de répondre au problème d'une entreprise de livraison. Le dépôt de l'entreprise est situé au 5 de la rue Bleu sur le plan de la ville Figure 1. La ville dispose également d'un réseau de métro que l'entreprise peut utiliser pour optimiser ses livraisons. Par optimiser on entend choisir le trajet le plus court et qui ne repasse pas deux fois par la même adresse. Le trajet de livraison standard de l'entreprise passe via quatre adresses de la ville, respectivement au 10 rue Rouge, 8 rue Mauve, au 22 rue Verte et 3 rue Marron avant de revenir au dépôt.

In fine, la méthode principale que vous allez implémenter devra prendre en paramètre les réseaux de transport et les adresses du trajet et renverra le trajet le plus court pour la livraison. Mais avant cela, quelques étapes..

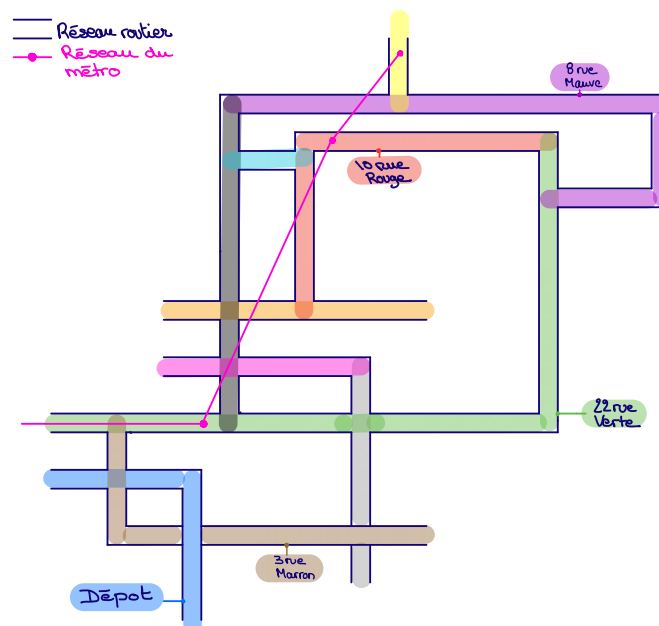


Figure 1: Réseaux de transports de la ville

Rendu attendu

Ce travail est à effectuer en **binôme**. Vous devrez déposer (**1 membre du binôme seulement**) sur Moodle avant le **9/11/2022, 7:00**, une archive nommée *nom1-nom2.tgz* contenant :

- Un document pdf (nommé *documentReponse-nom1-nom2.pdf*) répondant aux différentes questions marquées du symbole → de ce sujet.
- Un répertoire (nommé *code-nom1-nom2*) contenant votre code et un makefile. **Vous devez utiliser la librairie fournie au TP précédent.**

A partir du mercredi 9/11/2022 vous devrez évaluer les travaux qui vous auront été assignés, et ce avant le **16/11/2022, 23:00**.

Etape 1 : Modéliser, définir le problème formel, associer une classe de complexité

- Quelle modélisation à base de graphes proposez-vous pour ce problème ? Vous en préciserez les caractéristiques (orientée ?, étiquetée ? etc.) de votre (vos) graphe(s). Vous proposerez la(es) représentation(s) sagittale(s) correspondante(s) et les formalisations associées.
- Quel(s) est (sont) le(s) problème(s) formel(s) sous-jacent(s) ?
- Quelle est (sont) la (les) classe(s) de complexité associée(s) à ce(s) problème(s) ? Dans le cas où la(es) classe(s) trouvée(s) impliquera un temps d'exécution a priori peu raisonnable, justifiez pourquoi vous pensez que ça devrait bien se passer, ou abandonnez^a.

^aAttention si ça n'est pas la bonne option, vous serez fortement pénalisé par ce choix ;-)

Etape 2 : L'algorithme

L'objectif de cette étape est de trouver l'algorithme principal qui permet de répondre au problème formel et donc in fine à la problématique de l'entreprise. Pour cela vous devez faire des recherches pour identifier si cet algorithme existe déjà, et si oui dans quel mesure il faut l'adapter. Vous fournirez :

- L'état de l'art de vos recherches, notamment les références des sources qui vous ont permis de l'établir.
- Le pseudo-code de l'algorithme choisi et adapté.

Remarque : Par "trajet" on entend la liste des sommets traversés pour aller d'une origine à une destination. Si vous n'avez pas le temps de renvoyer le trajet complet proprement dit, renvoyez à minima l'ordre de passage des adresses à visiter.

Etape 3 : L'implémentation

L'objectif de cette étape est de produire le code du programme. Vous fournirez dans un répertoire (nommé *code-nom1-nom2*) :

- Un makefile qui permet de compiler et d'exécuter votre code, → si votre code ne compile pas ou si vous n'avez pas réussi à produire un résultat juste, mentionnez le dans le document pdf.
- Le code du programme qui doit contenir tout le nécessaire (y compris les librairies indispensables)
- Un répertoire *Data* qui contiendra vos données



Etape 4 : L'adaptation

- Que faut-il adapter dans votre modélisation si l'on souhaite que le trajet généré précise s'il emprunte le métro ou non ? (pas à coder)

Le métro de la ville est tombé en panne.

- Qu'implique cette panne sur votre algorithme/implémentation ?
- Faites le nécessaire pour obtenir un résultat qui tienne compte de ce changement. Vous expliquerez dans le document pdf ce que vous avez modifié, et afficherez les deux résultats (avec et sans le métro) de manière claire lors de l'exécution de votre code.