

# PROJET INFORMATIQUE

## Théorie des graphes Java



I.	Présentation .....	2
1.	Contexte et enjeux .....	2
2.	Objectifs du projet .....	3
II.	Organisation et planning .....	5
III.	Cahier des charges détaillé .....	5
1.	Choix des graphes .....	5
2.	Progression pas à pas .....	6
3.	Problématiques à étudier .....	7
4.	Conception et réalisation du programme Java .....	11
IV.	Travail à réaliser, rendus et modalités d'évaluation .....	12



## I. Présentation

### 1. Contexte et enjeux

La gestion des déchets constitue un enjeu majeur pour les collectivités territoriales, tant sur le plan environnemental qu'économique et logistique. L'évolution des modes de vie et de consommation a donné lieu à un doublement des quantités de déchets générés par les ménages en 40 ans. A cela s'ajoutent les déchets issus des bureaux, commerces, parcs et jardins ...

Le premier défi est bien sur la sensibilisation des citoyens à l'importance du tri, du recyclage et de la réduction des déchets.

Le deuxième défi est la complexité du problème de la gestion quotidienne des déchets qui est en réalité tout un processus comprenant la collecte, le transport, le tri, la sélection puis l'élimination ou la valorisation des déchets.

La collecte et le transport nécessitent déjà à eux seuls toute une organisation :

- la planification des jours et des itinéraires de ramassage, aux pieds des habitations ou dans des points de collecte.
- le contrôle et la distribution géographique des points de collectes (conteneurs).
- le contrôle des quantités collectées et la prise en compte des contraintes liées au nombre et à la disponibilité des camions, aux charges maximales des véhicules, à la capacité d'absorption par les usines de tris et de traitements ...

Optimiser les tournées de ramassage permet de réduire les coûts de transport, limiter les émissions de CO<sub>2</sub> et améliorer l'efficacité du service public.

## 2. Objectifs du projet

**Dans le cadre de ce projet, vous allez vous concentrer sur l'organisation de la collecte et du transport des déchets.**

Vous réfléchirez en particulier aux problématiques suivantes :

- Comment optimiser les itinéraires de ramassage aux pieds des habitations ou dans des points de collectes ?
- Comment planifier les jours/horaires de passage dans les différents quartiers pour tenir compte de la quantité de déchets à collecter et des contraintes liées au nombre et à la disponibilité des camions, aux charges maximales des véhicules ?

**Les objectifs de ce projet sont multiples :**

- En termes de théorie des graphes :
  - Réfléchir et modéliser les problématiques sous-formes de graphes.
  - Appliquer les notions et les algorithmes vus en cours à une thématique concrète de société.
  - Se documenter et effectuer des recherches sur les méthodes de résolution et les algorithmes permettant de trouver des solutions réalistes et adaptées à une situation réelle.
  - Mettre en évidence les difficultés de résolution liées à la complexité des algorithmes et aux multiples contraintes à prendre en compte.
- En termes de conception et de programmation orientée objet :
  - Appliquer les démarches de conception et de modélisation objet vues en cours de POO sur une étude de cas concrète et conséquente.
  - Renforcer la maîtrise du langage Java.

**Dans ce projet, les aspects réflexion, recherche et analyse sont primordiaux. Après une analyse approfondie du cahier des**

**charges, vous devrez réfléchir aux problématiques en termes de graphes, puis rechercher et comparer les méthodes de résolution possibles.**

**Le projet devra ensuite aboutir à la conception et à la réalisation d'un programme orienté objet en langage Java. Le programme sera multi-utilisateur. Il devra proposer plusieurs fonctionnalités à deux catégories d'utilisateurs : les collectivités et l'entreprise en charge de la collecte et du transport des déchets.**

Le programme devra par exemple permettre à une collectivité de soumettre le plan de la commune ou du territoire à couvrir, de signaler les modifications de circulation liées à d'éventuels travaux, d'indiquer des contraintes horaires à respecter, de consulter les quantités de déchets récoltés ... Il devra également permettre à l'entreprise chargée de la collecte de calculer les itinéraires de ramassage, de planifier les jours de collectes et de programmer les camions

...

**Votre projet comprendra plusieurs parties que vous devrez exposer dans un rapport compte-rendu et lors de soutenances orales :**

- Vos réflexions, recherches et analyses sur les différentes méthodes de résolution et algorithmes.
- Votre démarche de conception et de modélisation UML.
- Votre programme Java commenté et documenté.
- Votre analyse des résultats.

## II. Organisation et planning

- Le projet se conduit en équipe de trois ou quatre étudiants d'un même groupe de TD.
- Les étudiants doivent d'inscrire sur la plateforme projet la semaine du 03/11/2025.
- Le projet démarre la semaine du 10/11/2025 (distribution du sujet) et se termine le 06/12/2025 (rendu final déposé par chaque équipe).
- Des soutenances auront lieu la semaine du 08/12/2025.

Semaine du 03/11/2025	Semaine du 10/11/2025	06/12/2025	Semaine du 08/12/2025
Constitution des équipes	Distribution du sujet	Rendu final	Soutenances

## III. Cahier des charges détaillé

### 1. Choix des graphes

Pour le ramassage des déchets au pieds des habitations, le graphe de base est le réseau routier correspond au plan d'une commune. C'est un multigraphie pondéré par les distances entre les intersections ou carrefours. L'orientation du graphe sera soumise à différentes hypothèses comme expliqué un peu plus loin.

Vous devez commencer par travailler sur de petits graphes fictifs d'au plus quelques dizaines de sommets représentant des petits villages.

Vous pourrez ensuite travailler sur des graphes d'ordre plus élevé et récupérer des graphes réels correspondant à de véritables plans de communes.

Concernant les adresses des habitations et des différents bâtiments, vous pouvez simplifier en indiquant uniquement la section de rue (arc ou arête) sur lesquels ils se trouvent.

Vous pourrez ensuite être plus précis en adoptant par exemple la numérotation « à l'américaine », où les numéros correspondent à la distance depuis une des extrémités de la rue.

## **2. Progression pas à pas**

Les algorithmes formels de théorie des graphes s'appliquent rarement tels quels à des cas réels et concrets. Pour résoudre les problématiques de ce sujet, vous devrez adapter les algorithmes vus en cours, en construire de nouveaux en combinant plusieurs, en découvrir de nouveaux en effectuant des recherches et en vous documentant.

La prise en considération des contraintes réelles rend la résolution des problématiques plus difficiles et peut exiger l'utilisation d'algorithmes complexes faisant appel à des heuristiques et à des méthodes d'optimisation combinatoire afin d'obtenir des solutions satisfaisantes dans des temps de calcul raisonnables.

**Vous procéderez donc pas à pas, en partant d'hypothèses simples (même si peu vraisemblables), pour aller vers des hypothèses plus réalistes et proches des situations réellement rencontrées.**

**Vous ne pourrez probablement pas résoudre toutes les problématiques proposées avec tous les jeux d'hypothèses. Avancez progressivement dans les niveaux de difficulté. Ne proposez pas une résolution si vous n'êtes pas capable de la comprendre et de l'expliquer. L'objectif n'est pas de tout faire, mais de bien faire.**

## Hypothèses sur l'orientation des rues :



**HO1** : Toutes les rues sont à double sens et les camions ramassent d'un côté et de l'autre de la rue lors d'un seul passage. (Graphe non orienté)

**HO2** : Certaines rues peuvent être à sens unique. Les rues à double sens possèdent au moins deux voies et les camions ne ramassent que du côté de la voie sur laquelle ils passent. (Graphe orienté)

**HO3** : Certaines rues peuvent être à sens unique, d'autres à double sens. Dans les rues à double sens ne possédant qu'une seule voie, les camions ramassent des deux côtés lors d'un seul passage, mais dans les rues possédant plusieurs voies, ils ne ramassent que du côté de la voie sur laquelle ils passent. (Graphe mixte)

## 3. Problématiques à étudier

### a. Thème 1 : Optimiser le ramassage aux pieds des habitations.

Pour les trois problématiques ci-dessous, on suppose qu'un seul camion intervient. Il part toujours du centre de traitement et y revient à la fin de la collecte. Il faut trouver l'itinéraire qui minimise les distances parcourues.

#### Problématique 1 : Organiser la collecte des encombrants.

Dans la plupart des communes, les particuliers peuvent demander l'enlèvement à domicile de leurs encombrants (le service est souvent gratuit).



**Hypothèse 1 :** Un seul ramassage à la fois. Il faut déterminer l'itinéraire le plus court permettant de se rendre chez un particulier.

**Hypothèse 2 :** La mairie regroupe les demandes et propose des dates de tournées de ramassages. Une tournée est limitée à une dizaine de ramassages. Il faut calculer l'itinéraire le plus court permettant de passer chez chaque particulier de la liste en une seule tournée.

**Problématique 2 : Organiser la collecte des poubelles aux pieds des habitations.**



Le camion doit ramasser toutes les poubelles de la commune (chaque poubelle de chaque côté de chaque rue).

- Vous traiterez d'abord le cas idéal (mais fort peu réaliste) où tous les sommets sont de degrés pairs.
- Vous traiterez ensuite le cas un peu moins trivial (mais toujours fort peu réaliste) où deux sommets seulement sont de degrés impairs.
- Vous vous attaquerez ensuite au cas général (aucune contrainte sur la parité du degré des sommets).

Réfléchissez aux différentes méthodes de résolution possibles en fonction des hypothèses sur l'orientation du graphe.

**b. Thème 2 : Optimiser les ramassages des points de collecte.**



Vous supposerez dans un premier temps que le graphe est non orienté (hypothèse H01). Puis vous réfléchirez à la possibilité ou non d'étendre les approches aux autres hypothèses.

On suppose qu'un seul camion intervient.

Il faut trouver les itinéraires qui permettent de passer par chaque point de collecte en minimisant la distance totale parcourue. C'est le problème du voyageur de commerce qui est NP-complet. Nous vous proposons de tester et de comparer deux méthodes de résolution qui ne garantissent pas d'obtenir la solution optimale mais qui tentent de s'en approcher :

### **Approche 1 : Approche par plus proche voisin**

L'idée est de partir du centre de traitement, puis de visiter à chaque étape le point le plus proche non encore visité.

### **Approche 2 : Approche MST (arbre couvrant de poids minimum)**

- Créer le graphe complet dont les sommets sont les points de collecte et dont les arcs représentent les plus courts chemins entre les sommets.
- Trouver un arbre couvrant de poids minimum.
- Faire un parcours préfixe de l'arbre obtenu en partant du centre de traitement. Cela permet d'obtenir un ordre de visite des points de collecte.
- Si on suit le parcours obtenu, il faudra parcourir chaque arête deux fois. Donc la longueur totale de la tournée sera deux fois le poids total de l'arbre couvrant. Mais on peut appliquer du « shortcutting », c'est à dire supprimer les visites redondantes : Si on doit passer de X à Y puis à Z mais que Y a déjà été visité, on peut aller directement de X à Z par le plus court chemin.

### **Prise en compte des capacités**

Le parcours obtenu avec l'approche MST doit être découpé en plusieurs tournées respectant la capacité de collecte maximale par tournée.

#### Exemple :

Notons D le dépôt,  $P_i$  les points de collecte et  $c_i$  leurs contenances, C la charge maximale du camion.

Il faut découper le parcours obtenu en plusieurs chemins tels que la somme des contenances des points parcourus reste inférieure à C.

Supposons qu'il y ait 5 points à collecter et que le parcours obtenu soit :

**D → P<sub>1</sub> (c<sub>1</sub>=4) → P<sub>3</sub> (c<sub>3</sub>=3) → P<sub>2</sub> (c<sub>2</sub>=3) → P<sub>4</sub> (c<sub>3</sub>=3) → P<sub>5</sub> (c<sub>5</sub>=5) → D**

Si C=10,

alors il faut le découper en deux tournées :

**T1 : D → P<sub>1</sub> (c<sub>1</sub>=4) → P<sub>3</sub> (c<sub>3</sub>=3) → P<sub>2</sub> (c<sub>2</sub>=3) → D**

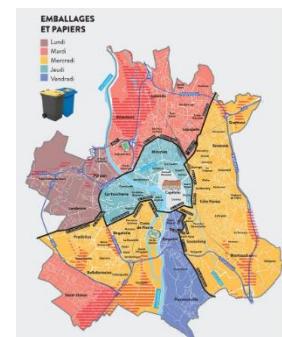
**et**

**T2 : D → P<sub>4</sub> (c<sub>3</sub>=3) → P<sub>5</sub> (c<sub>5</sub>=5) → D**

### c. Thème 3 : Planifier les jours de passage dans les différents quartiers.



En général les grandes municipalités divisent leurs communes en secteurs géographiques équilibrés dans lesquels les ramassages pourront avoir lieu à des créneaux différents. L'objectif est de réduire les phénomènes d'encombrement et de tenir compte des contraintes de capacités de collecte liées au nombre de camions et à leurs charges maximales.



Comment planifier les créneaux de collecte sur les différents secteurs ?

**Hypothèse 1 :** Deux secteurs voisins (géographiquement contigus) ne doivent pas être collectés le même jour. Aucune contrainte sur les capacités.

**Hypothèse 2 :** Deux secteurs voisins ne doivent pas être collectés le même jour. Pour éviter les nuisances il ne peut pas y avoir plus d'une tournée de ramassage par jour pour un

même secteur. Et il faut tenir compte de la quantité de déchets produites et des capacités de collecte :

- Chaque secteur  $s$  a une quantité estimée de déchets  $q_s$ .
- Un camion a une charge maximale  $C$ .
- Le nombre de camions utilisables simultanément est limité à  $N$ .
- La somme des quantités collectées dans les secteurs planifiés sur un même créneau ne doit pas dépasser la capacité de collecte.

## 4. Conception objet et réalisation du programme Java

Les algorithmes obtenus grâce à vos analyses et à vos recherches seront implémentés dans un programme Java.

Votre programme devra proposer plusieurs fonctionnalités aux deux catégories d'utilisateurs : les collectivités et l'entreprise en charge de la collecte et du transport des déchets.

### IHM - Visualisation des graphes

L'IHM de base devra être réalisée en mode console. A vous de prévoir des affichages de résultats lisibles et compréhensibles.

On ne vous demande pas de programmer une IHM permettant de visualiser les plans et d'y visualiser les parcours.

Trouvez des outils de visualisation existants pour nous présenter vos graphes et vos résultats lors des soutenances.

## IV. Travail à réaliser, rendus et modalités d'évaluation

- **Analyse du Cahier des charges**

Etudiez attentivement le CDC et n'hésitez pas à utiliser des tableaux pour représenter toutes les situations qui devront être étudiées.

- **Modélisation sous forme de graphes. Préparation des graphes de tests.**

Réfléchissez aux informations indispensables à stocker pour permettre la résolution des différentes problématiques.

Inventez des graphes fictifs de petites tailles pour pouvoir tester les différentes hypothèses.

- **Etude des problématiques, recherche de solutions**

Pour chaque problématique :

- Indiquez à quels problèmes de théorie des graphes elle se rattache (problèmes de coloration, recherche de plus court chemin, recherche de cycles ...).
- Expliquez brièvement puis récapituler sous forme de tableaux, les méthodes de résolution ou algorithmes que vous avez trouvés ou construits en fonction des différents jeux d'hypothèses adoptés.

- **Etude des algorithmes**

- Expliquez le fonctionnement des algorithmes retenus (en français sous forme de logigrammes).
- Etudiez et comparez leurs complexités. Précisez s'ils sont optimaux ou s'ils donnent des solutions approchées.

- **Conception du programme Java**

- En fonction des solutions que vous avez pu trouver, présentez les fonctionnalités de votre programme sous forme d'un diagramme de cas d'utilisation.

- Réfléchissez à l'architecture du programme et proposez des diagrammes de classes généraux puis détaillés.

- **Réalisation du programme et tests**

Implémentez le programme et testez-le sur vos petits graphes.

- **Présentation et analyse des résultats obtenus**

Analysez et comparez les résultats obtenus selon les différentes approches pour les différents jeux d'hypothèses.

- **Obtenir et travailler avec des graphes réalistes.**

- **Approfondir le sujet et aller plus loin.**

Si vous avez eu le temps d'aborder tous les points du CDC, vous pourrez approfondir le sujet, prendre en compte de nouvelles contraintes, et nous présenter vos recherches voire implémenter des fonctionnalités supplémentaires dans votre programme.

**Lors du rendu final (au plus tard le 06/12/2025) vous déposerez votre code et un compte rendu contenant :**

- Le planning suivi et la répartition des tâches effectuées par chaque membre de l'équipe.
- Votre démarche d'analyse et de conception telle que décrite ci-dessus.

**Des soutenances auront lieu la semaine du 08/12/2025.**

**L'évaluation du projet tiendra compte de votre travail collectif et individuel.**

**Travail collectif :**

- Qualité des rendus et présentations.
- Pertinence des recherches, des analyses et des explications.

- Niveaux de difficulté des solutions proposés qui doivent être clairement expliquées. Une méthode de résolution ou un algorithme non compris ne sera pas pris en compte.
- Pertinence des graphes de tests proposés.
- Qualité de la conception et de l'architecture du programme.
- Clarté de la présentation des résultats et pertinence de leur analyse.
- Recherche de graphes réalistes représentant de vraies communes.

**Travail individuel :**

- Implication dans le projet, importances et difficultés des tâches réalisées.
- Qualité des réponses individuelles fournies au jury lors des soutenances.

**Bon projet !**

**Julienne Palasi et toute l'équipe enseignante**