

Projet « Graphes et compression de données »

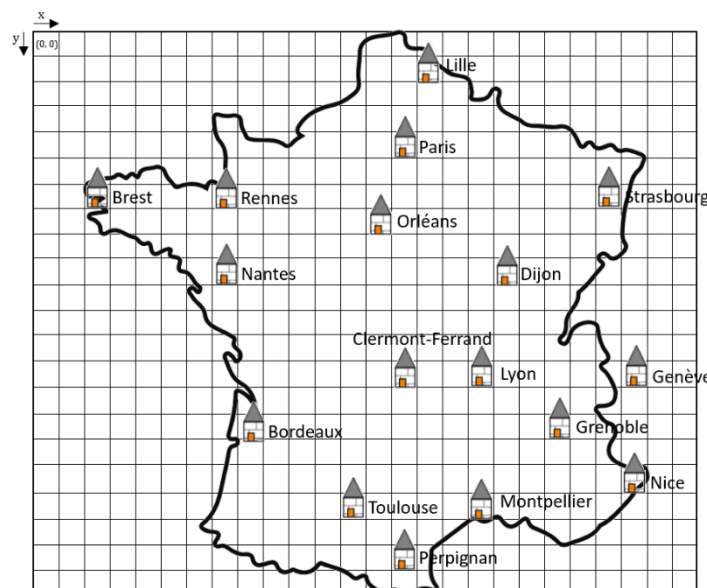
Un voyageur de commerce

Ahmad AUDI / Mouna EL ATTAR

Ce projet a pour objectif de réaliser une application basée sur la théorie de graphe et la compression de données. Il permet aussi d'approfondir vos connaissances pratiques en programmation C avancée.

Consignes

- Le **projet** doit être rendu par **mail** (audiahmad@hotmail.com) sous la forme d'une **archive compressée** contenant l'**ensemble** des **fichiers** nécessaires à la **compilation** et à l'**utilisation** du **projet**. Le nom du fichier ainsi que l'objet du mail doivent contenir vos noms, vos prénoms et votre numéro de la classe.
- Les **critères d'évaluation** seront **informés** au début de la première séance du projet.
- Les **élèves** doivent se mettre en **binôme**.
- Le **code source** doit être fourni en **C** et utilisera les bibliothèques usuelles. Votre code doit pouvoir être **compilé**, sans **erreur** et doit fonctionner sans erreur de segmentation. Il est indispensable de fournir un **Makefile**.
- Vous pouvez utiliser n'importe quelle **structure de données** (tableaux, pointeurs, listes chaînées simples, etc ...)

Sujet :

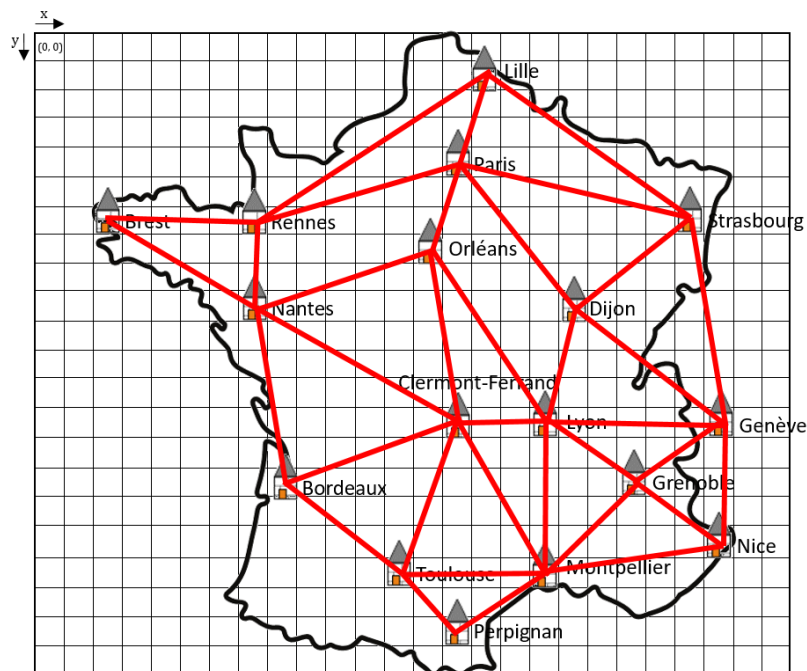
Un **voyageur** souhaite visiter **toutes les villes** mentionnées ci-dessus sur la **carte**. On considère que chaque case de la carte représente une distance de **10km**. Sachant que le prix de l'essence est en fonction de la distance parcourue, votre mission principale sera d'aider le voyageur à minimiser ses dépenses d'essence.

Pour résoudre ce problème, vous disposez d'un fichier **carte.txt** qui représente la numérisation conventionnée de la carte (Cf. les explications qui seront données lors de la première séance).

- 1- Implantez un programme qui permet de lire la carte à partir du fichier **carte.txt** donné. Les villes sont distinguées par des numéros allant de 1 jusqu'à 17. Les valeurs dans les autres cases de la carte valent 0.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Lille	Paris	Brest	Rennes	Strasbourg	Orléans	Nantes	Dijon	Clermont-Ferrand	Lyon	Genève	Bordeaux	Grenoble	Nice	Toulouse	Montpellier	Perpignan

- 2- Implantez une fonction qui détermine les distances entre les villes du graphe ci-dessous. Vous pouvez utiliser un autre fichier contenant les liaisons entre les villes.



- 3- Modélisez le graphe sous la forme d'une matrice de poids (distances).
- 4- Implantez un algorithme de la théorie des graphes qui permet de déterminer l'arbre couvrant de poids minimum.
- 5- Implantez une fonction qui permet de calculer la somme des poids des arêtes de l'arbre couvrant de poids minimum.
- 6- Implantez un algorithme permettant de compresser la solution optimale obtenue.

- 7- Implantez un algorithme permettant de décompresser la solution optimale obtenue (après compression).
- 8- Implantez un algorithme permettant de compresser le fichier **carte.txt** donnée
- 9- Implantez un algorithme permettant de décompresser le fichier **carte.txt** (après compression).
- 10- Utilisez une interface graphique qui permet de visualiser les différentes étapes du projet.