



Projet **CY-Trucks**

PRÉ-ING-2

Table des matières

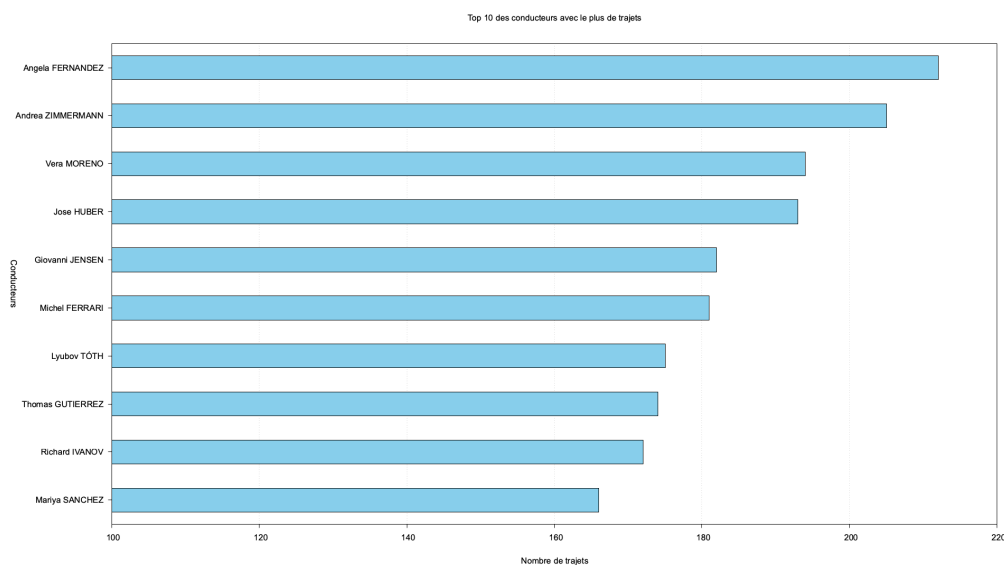
Présentation des options et des résultats	3
-[D1] : Conducteurs avec le Plus de Trajets:	3
-[D2] : Conducteurs et la Plus Grande Distance:	3
-[L] : Les 10 Trajets les Plus Longs:	4
-[T] : Les 10 Villes les Plus Traversées:	4
-[S] : Statistiques sur les Étapes:	5
Limites et bug	5
Répartition du projet	6
Planing de réalisation	6

Présentation des options et des résultats

Vous retrouverez tous les résultats présentés ci-dessous dans le dossier "demo" de notre projet, dans le sous-dossier "graphique". De plus, le dossier "temporaire" contient tous les fichiers nécessaires à la génération des graphiques. Ils ont été générés avec le fichier data.csv fourni par M. Grignon. Bien entendu, tous ces fichiers sont régénérables à partir de notre code.

-[D1] : Conducteurs avec le Plus de Trajets:

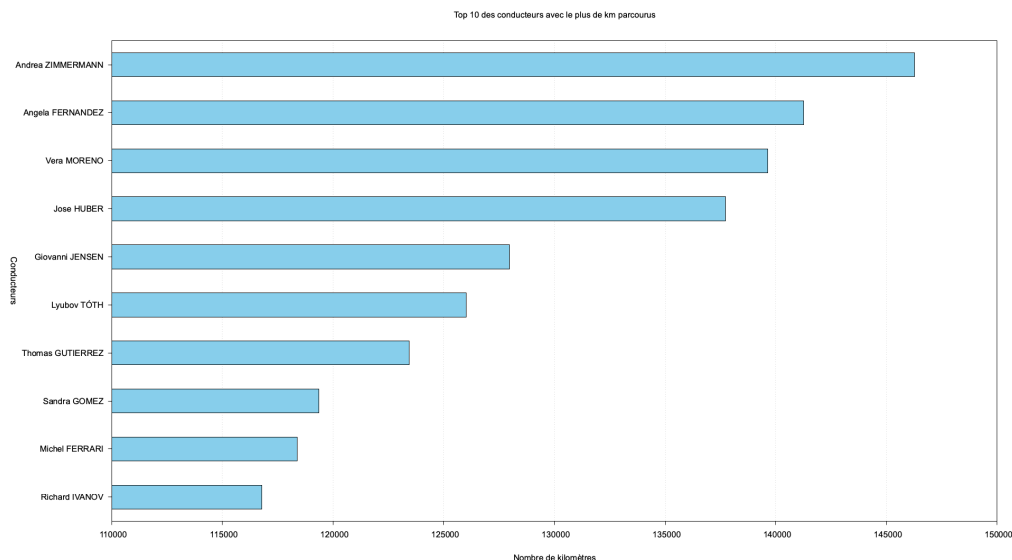
Ce traitement vise à extraire la liste des conducteurs avec le nombre de trajets différents qu'ils ont effectués. Cette liste est ensuite triée par ordre décroissant en fonction du nombre de trajets, et seuls les 10 premiers conducteurs sont conservés. Voici le résultat qu'on peut obtenir :



Le temps d'exécution sur notre machine est d'environ 6 secondes.

-[D2] : Conducteurs et la Plus Grande Distance:

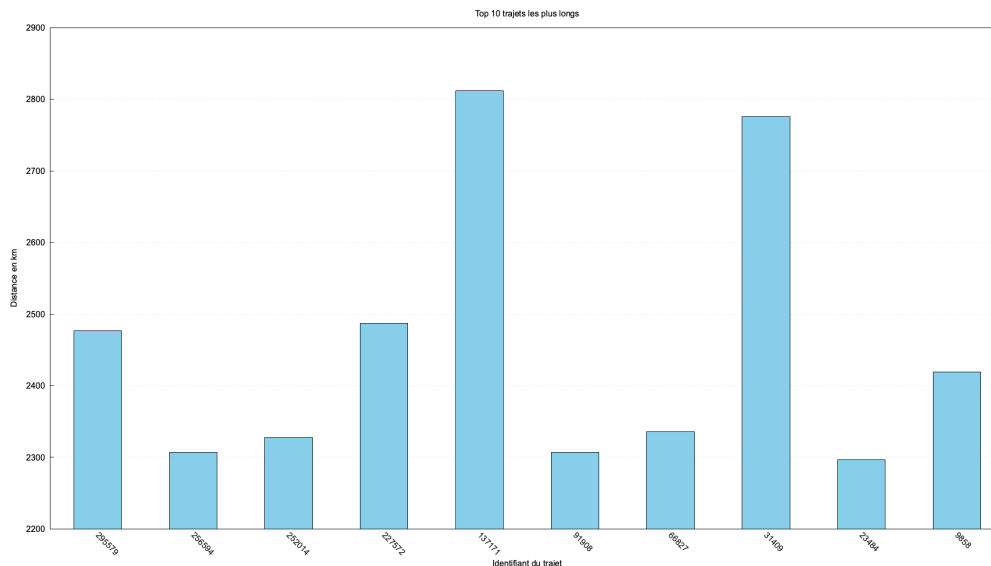
Ce traitement a pour objectif de calculer la distance totale parcourue par chaque conducteur en effectuant la somme des distances de toutes les étapes associées à ces conducteurs. Parmi toutes ces distances, seules les 10 plus grandes seront conservées et classées par ordre décroissant. Voici le résultat qu'on peut obtenir :



Le temps d'exécution sur notre machine est d'environ 6 secondes.

-[L] : Les 10 Trajets les Plus Longs:

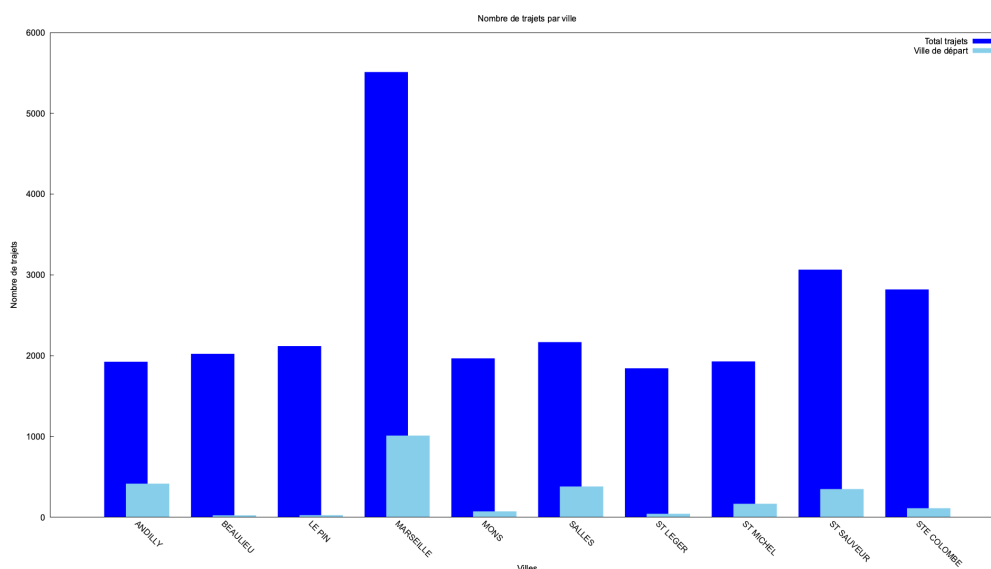
Ce traitement vise à obtenir la distance totale de chaque trajet, calculée comme la somme des distances de chaque étape pour ces trajets spécifiques. Parmi toutes les distances calculées, seules les 10 distances les plus importantes seront conservées. Ces distances seront ordonnées par numéro croissant d'identifiant de trajet. Voici le résultat qu'on peut obtenir :



Le temps d'exécution sur notre machine est d'environ 8 secondes.

-[T] : Les 10 Villes les Plus Traversées:

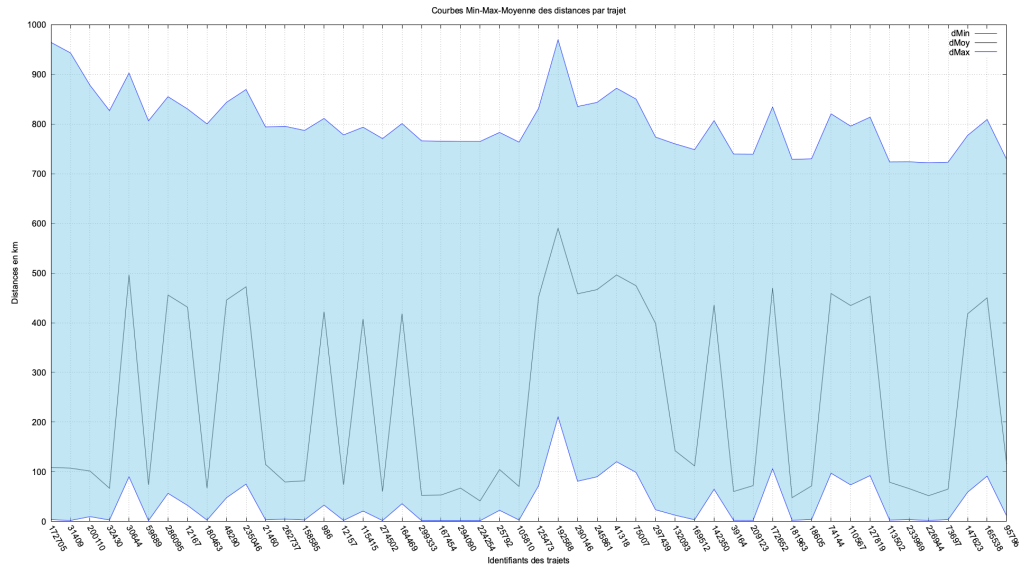
Ce traitement vise à analyser les trajets en comptant le nombre de fois où chaque ville est soit traversée par un trajet, soit utilisée comme ville de départ d'un trajet. Une ville peut apparaître deux fois dans les données : une fois comme ville de départ et une fois comme ville d'arrivée. Le traitement sélectionne les 10 villes les plus fréquentes par ordre alphabétique. Voici le résultat qu'on peut obtenir :



Le temps d'exécution sur notre machine est d'environ 4 secondes.

-[S] : Statistiques sur les Étapes:

Ce traitement a pour objectif d'analyser les différentes distances des étapes pour chaque trajet. Il récupère les distances minimales, maximales et moyennes des étapes pour chaque trajet. Le résultat est représenté sous forme d'un graphique de type courbes min-max-moyenne. Voici le résultat qu'on peut obtenir :



Le temps d'exécution sur notre machine est d'environ 6 secondes.

Limites et bug

Non-contrôle précis du temps d'exécution : Les options telles que -d1, -d2 et -l peuvent varier considérablement en temps d'exécution sur différents ordinateurs, par exemple de 2 à 20 secondes. Le programme ne fournit pas de mécanisme précis pour contrôler ou prédire ces variations de temps d'exécution. Toutefois, comme précisé ci-dessus, sur notre machine les temps d'exécutions sont respectés.

Gestion de la mémoire non optimale : Le programme en langage C prend en compte la mémoire de manière non optimale. Il est sensible à la taille des fichiers qu'on lui demande de traiter, ce qui peut entraîner des problèmes de performance et de stabilité pour des fichiers volumineux.

Répartition du projet

Personne	Tache assigné
Eve	Faire option -s en C et son graphique
Thomas	Faire option -t en C et son graphique
Luc	Faire option -d1 -d2 et -l en Shell et les graphiques
Tous ensembles	Requis obligatoire dans le Shell, le PDF de présentation, organisation du git

Planing de réalisation

Semaine	Tache réaliser
04/12/2023	- Lecture du projet décomposition des taches principales - Création du git et des principaux fichiers
11/12/2023	- Ajout de la fonction de traitement d1 , d2 et l dans le Shell - Test de plusieurs commandes pour la rapidité
18/12/2023	- Recherche pour créer les graphiques -d1, -d2, -l. Première itération non concluante - Ajout graphique - d1 , - d2 et - l - Graphique - d1 et - d2 fonctionnel mais pas la bonne forme.
25/12/2023	- Début rédaction/mise en forme du README - Ajout de commentaire dans le code - Ajout graphique -s et -t
01/01/2024	Aucune modification
08/01/2024	Aucune modification
15/01/2024	- Modification de gitignore pour simplifier les commit - Finition des graphiques - d1 et - d2
22/01/2024	- Ajout de la fonction de traitement - s en C - Ajout de l'option s dans le Shell - Organisation et uniformisation
29/01/2024	- Ajout de la fonction de traitement - t en C - Ajout de l'option t dans le Shell - Rédaction du PDF d'explication - Finition de louter éléments qui serai passé au travers

A noté: Dans notre configuration de travail, il est important de noter que la personne ayant effectué les commits sur Git n'est pas nécessairement celle qui est assignée à la tâche. Cette situation découle de notre utilisation de dossiers partagés ainsi que d'une maîtrise limitée des commandes Git par certains membres de l'équipe.