

Rapport du projet CY Trucks

Nous avons commencé par étudier en détail de sujet afin d'en dégager les différentes exigences fonctionnelles et techniques et ainsi veiller à la bonne compréhension du sujet et à la bonne répartition des tâches au sein du groupe.

Au niveau de la répartition des tâches au sein du groupe nous avons procédé comme suivant : Au niveau du traitement de données des cinq options, Tom REBOUR et Lilas GASMI se sont chargés de l'option -d1, -d2 et -l tandis que Florian CHAPIN s'est occupé de toute la partie C y compris le makefile et de la partie graphique de chaque option. Pour le reste tel que l'option -h et la vérification des arguments (If,Else if...) dans le shell, ce sont Lilas et Tom qui l'ont fait.

Au niveau du planning de réalisation, les options -d1, -d2 et -l ont été réalisées durant les vacances de Noël en même temps que le script shell qui vérifie les conditions au bon déroulement du programme. Au retour des vacances, nous avons implémenté la partie graphique pour ces options. Lors de la première semaine de l'année en présentiel nous avons réalisé l'option -s avec sa partie puis enfin nous nous sommes occupés de la partie t, de sa partie graphique et du makefile lors de la semaine après les partiels. Enfin, le readme et ce document ont été réalisés lors de cette semaine également.

Au niveau des limitations fonctionnelles, tout ce qui a été demandé est fonctionnel, testé sur ordinateur personnel et sur ordinateur de l'école. Nous avons cependant remarqué un petit détail pour l'option -s. En effet, il y a un léger arrondissement des valeurs ainsi les valeurs sont légèrement différents à quelques dixièmes près.

Maintenant en ce qui concerne les fichiers dans le dossier demo nous allons les présenter par option.

Pour l'option -d1, il y a le fichier data_d1.csv qui résulte de la commande awk accompagnée d'un sort et d'un head n-10. Ce fichier contient donc le nombre de trajets par conducteur. Il y a également les images résultats_d1tmp.png qui est produite par le programme gnuplot puis l'image finale résultats_d1.png qu'on obtient à l'aide de la commande convert qui tourne l'image de 90 degrés vers la droite.

Pour l'option -d2, il y a le fichier data_d2.csv qui résulte de la commande awk accompagnée d'un sort et d'un head n-10. Ce fichier contient donc le nombre de trajets par conducteur. Il y a également les images résultats_d2tmp.png qui est produite par le programme gnuplot puis l'image finale résultats_d2.png qu'on obtient à l'aide de la commande convert qui tourne l'image de 90 degrés vers la droite.

Pour l'option -l, il y a le fichier data_l.csv qui résulte de la commande awk accompagnée d'un sort, d'un head n-10 puis d'un autre sort. Ce fichier contient donc le nombre de trajets par conducteur. Il y a également les images résultats_l.png qui est produite par le programme gnuplot.

Pour l'option -s, il y a le fichier data_s.txt que l'on a généré à partir du fichier data.csv. Ce fichier contient uniquement les colonnes qui contiennent les informations dont nous avons besoin pour le traitement de cette option tout en changeant le séparateur entre les colonnes pour faciliter la récupération des données par le programme C. L'exécution du programme C correspondant à cette option va créer un fichier de données data_s.csv qui contiendra toutes les données correspondantes au traitement de l'option. Enfin, on obtient l'image correspondante résultats_s.png avec le programme gnuplot.

Enfin Pour l'option -t, il y a le fichier data_t.txt que l'on a généré à partir du fichier data.csv. Ce fichier contient uniquement les colonnes qui contiennent les informations dont nous avons besoin pour le traitement de cette option tout

en changeant le séparateur entre les colonnes pour faciliter la récupération des données par le programme C mais également le séparateur pour remplacer les espaces pour les noms composés de ville par des tirets. L'exécution du programme C correspondant à cette option va créer un fichier de données data_t_temp.csv qui contiendra toutes les données correspondantes au traitement de l'option. Avant de générer le graphique, on enlève les tirets pour les noms composés de ville pour remettre des espaces ce qui correspond à leur nom initial. Enfin, on obtient l'image correspondante résultats_t.png avec le programme gnuplot.

De plus dans le dossier demo, il y a deux fichiers temporaires manquants : data_s.txt et data_t.txt dû à leur volume trop grand. Il faudra donc exécuter le programme une fois pour les obtenir. De plus, dans le dossier demo il y a également les images résultats_d1temp.png et résultats_d2temp.png qui ne sont pas reproductibles par le programme directement pour que les résultats affichés dans images soient plus agréables à voir. Si jamais il est réellement nécessaire de les voir apparaître dans le dossier images, alors nous vous invitons à mettre les commandes « rm images/Résultats_d1tmp.png » (ligne 84) et « rm images/Résultats_d2tmp.png » (ligne 118) en commentaire (ajouter # devant).

Nous avons testé notre programme sur les ordinateurs de cy tech ainsi nous avons pu donc noter les différents temps de traitement :

- option -d1 : 7 secondes
- option -d2 : 3 secondes
- option -l : 6 secondes
- option -s : 8 secondes
- option -t : 24 secondes

Ainsi, il est donc possible de se faire une idée quand aux temps de traitement que l'on obtiendra lorsque que l'on exécutera le programme.

De plus, pour les options -s et -t, il y a eu une collaboration avec le groupe de Matias Vinkovic en Mi5.