

Présentation du projet CY-Truck

WEISS Zachary
MANCEAU Alexandre

1. Répartition des tâches

Il nous fallait d'abord comprendre et analyser la structure finale du programme. À la vue des capacités d'analyse et de traitements de données que le bash permet, il nous sembla être une bonne idée de commencer par celui-ci. Zachary a ainsi commencé à créer la structure mère du programme bash (vérification des dossiers, ...) tandis que Alexandre s'attela aux traitements.

Pour la partie C, Zachary s'est occupé de la modularité principalement, tandis que Alexandre se concentrait sur les traitements t et s, sous les conseils et aides de Zachary. Zachary s'est occupé de la création de toute la partie gnuplot en parallèle.

2. Planning du groupe

Nous utilisons le réseau social 'Discord' pour pouvoir se contacter. Nous faisons des appels hebdomadaires (x2) pour voir où chacun en était au niveau de ses tâches, puis nous pouvions les mettre en commun.

Si un de nous deux avait un problème, nous pouvions nous aider mutuellement, tout en restant en appel.

Nous avons décidé, dans un premier temps, de nous concentrer sur toute la partie bash du projet, que ça soit dans la structure ou bien les traitements. Le choix d'effectuer les traitements d1, d2 et l en bash fut opté.

Une fois cela fait, nous devons nous concentrer sur toute la partie C : en passant par la réalisation du programme en lui-même jusqu'à la modulation du programme. Pour ce qui est de la modulation, un regard sur le projet de préing 1 nous permis de comprendre immédiatement comment procéder.

En ce qui concerne la création des algorithmes de traitements, pour créer des formats AVL, nous avons, grâce aux cours fournis (et Internet), une base solide pour pouvoir aboutir sur des AVL performants et fiables.

3. Limites fonctionnelles du programme

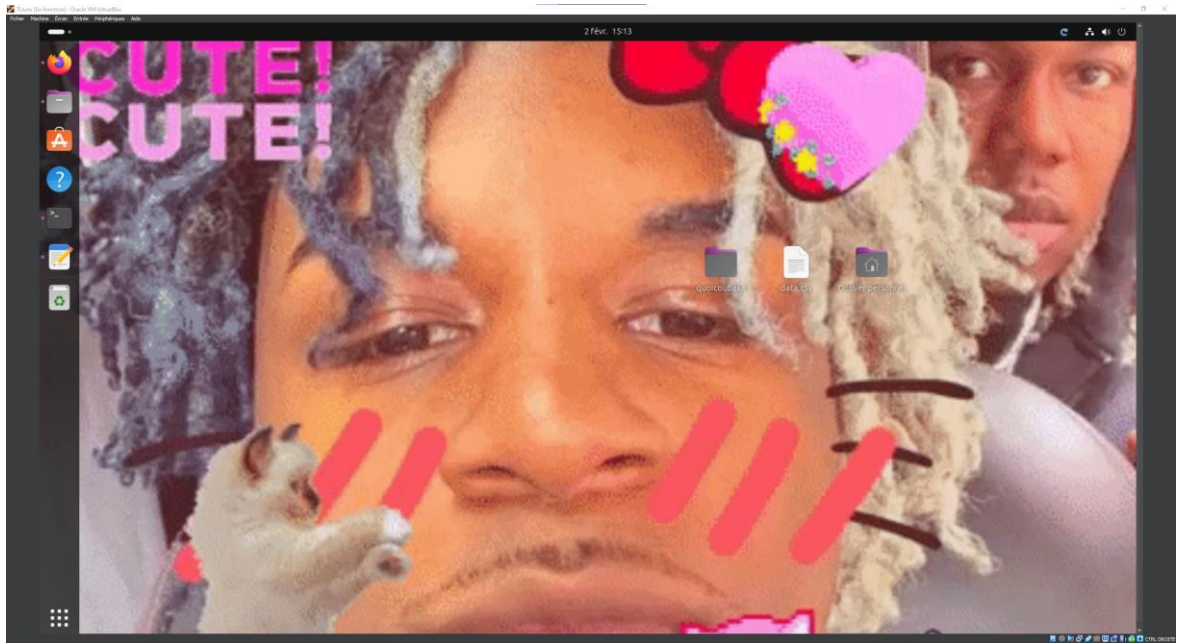
Le programme, dans sa partie bash, ne semble contenir de défauts majeurs. Tant dans l'exécution des traitements, la vérification de chaque élément ou encore la création des courbes et histogrammes par gnuplot.

Il semble néanmoins important de noter la nécessité de la bibliothèque de commande 'Imagemagick', notamment la commande 'convert' pour pouvoir effectuer une rotation des histogrammes lors des traitements d1 et d2, car gnuplot ne peut effectuer d'histogrammes horizontaux (la création d'une fonction de déplacement de chaque pixel de l'image fut envisagée, mais ne put aboutir).

Dans ce qui concerne la partie C, la partie s n'est pas fonctionnelle, mais des fonctions et applications existent (simplement pas finis). La partie C du traitement t quant à elle fonctionne.

4.Présentation du dossier 'démon'

Pour l'exécution du programme, il nous fallait nécessairement un environnement linux. Ainsi, Zachary a installé sur son pc le logiciel 'Virtual Box', accompagné de la dernière iso 'Ubuntu' pour pouvoir exécuter dans de réels conditions les programmes créés sur le site 'Replit' (photo de la machine virtuel ci-dessous) :



Dans les temps d'exécution du programme, toutes les conditions sont respectées :

```
Dossier images opérationnel
Dossier demo opérationnel
Dossier temp en nettoyage
Dossier temp vidé
-----
Nombre total d'arguments : 3
Liste des arguments :
-d1
-d2
-l
-----
Traitement d1 en cours
Tracage en cours : -d1
Tracage enregistré sous : images/d1_20-17-2024-34_test.png
Temps d'exécution : 3 seconde.s
-----
Traitement d2 en cours
Tracage en cours : -d2
Tracage enregistré sous : images/d2_20-17-2024-37_test.png
Temps d'exécution : 1 seconde.s
-----
Traitement l en cours
Tracage en cours : l
Tracage enregistré sous : images/l_20-17-2024-38_test.png
Temps d'exécution : 4 seconde.s
```

De plus, nous avons pu faire les graphiques suivant par gnuplot (les graphiques d1, d2 et l sont effectués depuis le 'main.sh', tandis que les graphiques de s et t sont effectués à partir des données retrouvables sur Teams) :

