



Informatique 3,
CY Cergy Paris Université

Projet d'Informatique

Maths-Informatique, MI-2

Marc-Antoine ABALE
Lucien BOYER
Eliot Durand de gevigney

Sous la direction de Monsieur

Année universitaire 2024-2025

Romuald GRIGNON

Dans le but de réaliser notre projet d'informatique, qui consiste à créer un programme permettant de faire la synthèse de données d'un système de distribution d'électricité, nous avons dû trouver la meilleure manière de collaborer.

Premièrement, nous avons décidé de communiquer à travers un groupe WhatsApp afin de nous organiser pour le projet, après que chacun ait pris connaissance du sujet. Nous nous sommes réunis pour essayer de le comprendre et surtout de nous faire une idée claire de ce qui nous était demandé. Ensuite, nous avons séparé le travail. Nous avons aussi fait à plusieurs reprises des réunions sur Discord afin d'avancer sur le projet en dehors des heures de cours.

Étant donné que la partie qui était nouvelle pour nous était le script shell, nous avons décidé de mettre deux personnes dessus et de laisser Eliot sur la partie programmation en C, qui nous semblait plus courte et plus abordable.

En effet, ayant déjà une idée des différentes fonctions à utiliser, nous avons décidé que c'était la meilleure décision pour avancer de manière efficace. (Il est important de noter qu'au début du projet, nous avons eu recours à l'IA à travers ChatGPT pour nous familiariser avec le script shell, ce qui nous a permis de mieux comprendre et de pouvoir avancer beaucoup plus rapidement sur la suite du projet).

Pendant qu'Eliot s'occupait de programmer la partie où l'on doit calculer la somme des consommations pour une station HV-B, HV-A ou un poste LV, en utilisant un arbre AVL, Lucien et Marc-Antoine s'occupaient de s'assurer que toutes les options obligatoires étaient présentes et que leurs valeurs étaient cohérentes, en veillant à ce qu'un message d'erreur soit affiché si ce n'était pas le cas. Mais nous nous sommes aussi occupés de vérifier la présence des dossiers tmp ainsi que celle de l'exécutable C sur le disque dur, tout en nous assurant de lancer la compilation dans le cas contraire. Lorsque Eliot termina sa partie, il nous rejoignit pour nous aider sur la partie traitement des données. Tout ceci s'est déroulé durant la première semaine. C'est cette semaine-là que nous avons commencé à travailler sur Replit.

La deuxième semaine était dédiée aux tests. L'objectif était de s'assurer que le traitement des données et l'affichage des résultats se déroulaient bien, et cela pour les HV-B, HV-A et les postes LV. Ce qui nous a aussi permis de vérifier que le programme C fonctionnait correctement.

C'est la troisième et dernière semaine que nous nous sommes remis à travailler en local et nous nous sommes occupés de faire un fichier Makefile pour pouvoir compiler correctement notre projet, calculer la différence entre la capacité totale et la consommation totale, et trier le fichier en fonction de la valeur croissante de cette différence.

Concernant le rapport, c'est Marc-Antoine qui s'en est occupé. Eliot s'est occupé du Readme. Lucien s'est occupé de continuer d'avancer sur le shell et a réalisé le makefile. De manière générale, tout le monde a essayé de participer à la conception du script shell lors de la semaine. Puis nous avons ajouté le graphique représentant les 10 postes LV les plus chargés, et les 10 postes LV les moins chargés comme bonus. Nous avons remplacé la commande awk par grep lors d'un cours d'informatique pour des questions de rapidité de notre programme et cela sur les conseils de notre professeur.

Les limitations fonctionnelles que nous pouvons pointer sont que si notre application devait traiter un fichier beaucoup plus volumineux alors elle aurait sûrement plus de difficultés à le traiter, ce qui se traduirait par un temps d'exécution beaucoup plus long.

Une autre limitation que nous avons découvert est que dans le C la fonction qui vérifie si les données d'entrée sont négatives ne marche pas car tout les moins "-" sont transformés en des 0 par la commande "tr" dans le shell. On peut considérer cette erreur comme minime grave car la donnée sera juste renvoyée mais positivement. Nous n'avons pas trouvé de solution satisfaisante avant que le temps imparti pour le projet se termine.

Néanmoins toutes les fonctionnalités du cahier des charges sont respectées ainsi que les sécurités de notre programme (notamment lors de la rentrée des arguments par l'utilisateur). De plus nous avons réalisé la fonctionnalité bonus qui est de faire un graphique à partir du fichier lv_all_minmax. (on a pas réussi à mettre la couleur correspondante sur les barres du graphique : rouge pour négatif et vert pour positif).

