

Compte rendu du projet C-WIRE

Dans le cadre de la matière Informatique 3, nous avons dû réaliser un projet intitulé C-WIRE. La première étape a été de constituer nos groupes. Naturellement, étant donné que nous étions dans la même classe en pre ing 1, nous avons décidé de travailler ensemble. Une fois notre groupe formé, nous avons pris connaissance du sujet.

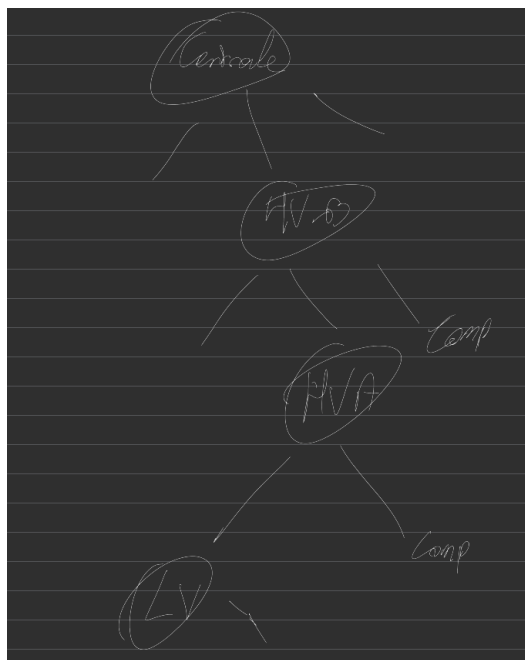
Honnêtement, au début, nous étions complètement perdus. Nous ne comprenions ni les attentes ni comment réaliser ce programme. C'était un moment délicat, nous ne savions pas par où commencer. Heureusement, le fait que Francisco habite à proximité de l'école a été un atout. Cela nous a permis de travailler ensemble presque tout le temps.

Notre première mission a été de décortiquer le cahier des charges. Après plusieurs heures de discussions et de brainstorming, Francisco a su bien comprendre le fonctionnement global du programme. Il nous a réalisé un beau dessin pour nous expliquer ce dernier.

Répartition des tâches

Pour commencer nous avons repartit grossièrement les tache et nous avons réalisé une to do list

- Francisco a pris en charge la structure initiale du projet et a dessiné les schémas explicatifs qui ont servi de base à notre compréhension commune.
- Jilani a géré l'écriture du script Shell, notamment la validation des arguments, la vérification des fichiers nécessaires et le traitement des données avec `awk`.
- Corentin s'est plutôt penché sur le code en C, en particulier la gestion des AVL



Organisation fichier :

- CSV dans dossier "input"
- Programme C dans dossier "codeC"
- Graphiques dans dossier "graphs" sur le disque dur
- Fichier temporaire dans dossier "tmp"
- Résultats dans dossier "test"
- Script shell à la racine

4 décembre TODO :

- Fichier de sortie avec nom
- Première ligne indiquant le nom des colonnes !
- Cas lv all ?
- Vérification et compilation du programme C
- Durée du traitement à afficher avec la fonction time (faire la différence)
- Libération des données alloué dynamiquement (arbre AVL notamment)
- Création du graphique avec GnuPlot
- Attention langue utilisée !!!!

10 décembre TODO :

- Trier par capacité croissante
- Cas lv all, mettre dans un fichier et regarder l'ordre
- Vérifier la libération de donnée
- GnuPlot

11 décembre TODO :

- Vérifier libération de donnée
- Langue utilisée
- GnuPlot

21 décembre TODO :

- GnuPlot

22 décembre :

- Fin

Planning de réalisation

1. **Semaine 1** : Familiarisation avec le Shell et création du script.
2. **Semaine 2** : Programmation en C, mise en place structures et outil AVL.
3. **Semaine 3** : Réalisation « lv_all_min_max » et optimisation de la mémoire.
4. **Semaine 4** : réalisation du bonus graphique avec gnuplot et finalisation du projet.

Les étapes du projet

1. Apprendre à travailler avec le Shell

Notre première épreuve technique a été de nous familiariser avec le Shell. Ce n'était pas simple, mais Jilani s'est en grande partie occupé de cette tâche. Grâce aux travaux dirigés et à des recherches sur internet, il a en grande partie géré la partie Shell.

Nous avons donc commencé par écrire le script Shell lors de la première semaine. Voici les principales étapes :

1. Validation des arguments : Nous avons vérifié que les arguments passés au programme étaient valides et que l'option « -h » était bien gérée.
2. Vérification des fichiers requis : Le programme s'assure que tous les fichiers nécessaires, comme l'exécutable, sont présents.
3. Traitement du fichier CSV : Nous avons utilisé awk pour trier les données du fichier CSV. Le awk fut une très bonne découverte pour nous. Grâce aux conseils de camarades d'autres classes nous sommes partie sur cette stratégie là pour le tri et ça a été une très bonne idée. Jilani s'est plongé dans ce langage, qui ressemble qui ressemble plus ou moins à du C et a réussi à bien faire fonctionner cela.

Après cette phase, le script gère la création de deux fichiers : un pour les stations et un autre pour les consommateurs. À partir de là, la partie C prend le relais.

2. Le code C

La deuxième semaine a été consacrée à la programmation en C. Corentin a pris les rênes pour cette partie et a divisé le code en quatre modules principaux :

1. `Avltree_utils` : Ce module gère tout ce qui concerne les arbres AVL, notamment leur création et leur modification.
2. `Avltree_usage` : Il s'occupe de la lecture des arbres et de l'écriture des résultats dans les fichiers.
3. `utils` : Ce module regroupe des fonctions pratiques comme `toUpperCase`, qui convertit des caractères en majuscules.

3. Surmonter les difficultés et optimiser

Lors de la troisième semaine, nous avons abordé le cas « `lv_all_min_max` », qui nous posait beaucoup de problèmes. Après plusieurs échecs et l'abandon temporaire du bonus, nous avons demandé de l'aide à nos camarades pour essayer de trouver une meilleure approche.

Finalement, cette partie du projet est une fierté pour nous. Corentin a eu l'idée d'utiliser le concept de tas pour optimiser la mémoire au mieux, et cette stratégie s'est révélée très efficace.

4. Ajouter une dimension graphique

La dernière semaine a été dédiée à l'implémentation du graphique. Nous avons exploré `gnuplot` grâce à des tutoriels et des ressources en ligne. Malgré la difficulté de compréhension, nous avons réussi à intégrer la fonction graphique qui permet de visualiser les résultats (`min_max`). Cette étape nous a aussi aidés à mieux comprendre les valeurs absolues nécessaires pour le « min max » que nous n'avions pas bien compris.

Limitations fonctionnelles

- **Gestion des doublons dans le fichier « min max »** : Si le programme récupère moins de 20 utilisateurs, des doublons peuvent apparaître.
- **Optimisation du tri** : Nous ne sommes pas certains d'avoir utilisé la méthode la plus optimale.
- **Bonus graphique** : Bien que fonctionnel, le graphique pourrait être davantage enrichi pour offrir une meilleure lisibilité des données.
- **Point ou point-virgule** : concernant les fichiers de sorti il y avait écrit dans le sujet qu'il fallait que les données dans le fichier soit séparé par des " : " cependant nous ne trouvions pas cela très logique. Nous avons appliqué cela mais si par exemple le csv s'ouvre sur Excel il y a 3 colonnes mais tout se met dans la même colonne.

Aussi dans le dossier [présentation des résultats](#) vous pourrez trouver des exemples de ce que le programme nous renvoie et du graphique.

Conclusion

En conclusion, ce projet nous a appris énormément de chose sur le plan technique. Travailler chez Francisco a vraiment été un réel atout, car cela nous a permis de réaliser ensemble environ 90 % du code.

Il y a bien sûr des aspects à améliorer, mais ce projet que nous pensions impossible au début s'est transformé à notre sens en une réussite. Plus nous avançons, plus nous prenions confiance en nos compétences. Nous sommes fiers d'avoir relevé ce défi et de pouvoir présenter un travail abouti.

Jilani Mohamed, Corentin Pellerin, Francisco Tornos Lloret