

Projet C-Wire

Répartition des tâches

Nom	Date de début	Date de fin	Responsable
Réunion découverte sujet	19/11/2024	19/11/2024	Emma/Amine/Marwa
Réunion répartition des tâches	20/11/2024	20/11/2024	Emma/Amine/Marwa
Partie Shell	22/11/2024	28/11/2024	Emma/Marwa
Partie C	28/11/2024	15/12/2024	Emma/Amine/Marwa
Gnuplot	15/12/2024	?? /12/2024	Emma/Marwa
Readme	15/12/2024	17/12/2024	Amine/Marwa
Rédaction pdf	28/11/2024	20/12/2024	Emma/Marwa

Détail des tâches :

- **Partie Shell :**

Vérification présence de l'exécutable : Marwa

Vérification/Création/Vidage des dossiers temp/graphs : Emma / Marwa

Écriture de l'aide (aide.txt) : Emma / Marwa

Codage des traitement :

hvb_comp (traitement HVB avec consommateur « comp ») : Marwa

hva_comp (traitement HVA avec consommateur « comp ») : Marwa

lv_comp/indiv/all (traitement LV pour différents consommateurs) : Marwa

Temps d'exécution pour chaque traitement : Marwa

- **Partie C :**

Fichier avl.h :

Déclaration de la structure Station : Emma

Déclaration des prototypes des fonctions AVL : Emma

Fichier avl.c :

Ecriture des fonctions de base pour l'AVL :

Fonction max (trouver le maximum entre deux valeurs) : Amine

Fonction hauteur (calcul de la hauteur d'un nœud) : Amine

Fonction equilibrerAVL (calcul du facteur d'équilibre) : Emma

Fonction de rotations : Emma

Insertion équilibrée dans l'arbre AVL (insérer) : Marwa

Parcours infixe pour afficher les données : Amine

Libération de la mémoire allouée : Marwa

Fichier main.c :

Gestion des arguments en ligne de commande : Emma

Lecture et traitement des données depuis un fichier CSV : Emma

Appel de la fonction insérer pour chaque ligne du fichier : Marwa

Affichage des résultats (parcours infixe) : Amine

Gestion des erreurs liées aux fichiers : Marwa

Libération de la mémoire à la fin du programme : Marwa

- Partie Gnuplot :

Graphique lv_minmax : Emma / Marwa

- Autres tâches :

Ecriture du makfile : Marwa

Ecriture du readme : Amine/Marwa

Rendu du projet sur Github : Marwa

Rédaction du PDF : Emma / Marwa

Descriptif du code :

- Traitement -hvb_comp

Donne les informations sur les stations HVB pour les consommateurs « comp » triées par ordre croissant de capacité.

Après avoir testé sur les ordinateurs les plus performants que nous avons, le temps d'exécution de ce traitement est de 11 secondes.

```
emma@Cookie:~/Projet-CY-Wire$ ./c-wire.sh data/c-wire_v25.dat hvb comp
Le dossier temporaire existe. Vidage en cours...
Le dossier a été vidé avec succès.
Compilation de c-wire avec Makefile dans ./progC...
make: Nothing to be done for 'build'.
Compilation réussie.
Traitement en cours pour le type de station 'hvb' et consommateur 'comp'...
Résultat intermédiaire : temp/hvb_comp.csv
Résultat final : temp/hvb_comp_somme.csv
Temps d'exécution du traitement : 11 secondes
Le fichier ./temp/hvb_comp.csv a été généré avec succès dans le dossier temp.
Temps d'exécution du graphique : 11 secondes
```

- Traitement -hva_comp

Donne les informations sur les stations HVA pour les consommateurs « comp » triées par ordre croissant de capacité.

Après avoir testé sur les ordinateurs les plus performants que nous avons, le temps d'exécution de ce traitement est de 12 secondes.

```

• emma@Cookie:~/Projet-CY-Wire$ ./c-wire.sh data/c-wire_v25.dat hva comp
Le dossier temporaire existe. Vidage en cours...
Le dossier a été vidé avec succès.
Compilation de c-wire avec Makefile dans ./progC...
make: Nothing to be done for 'build'.
Compilation reussie.
Traitement en cours pour le type de station 'hva' et consommateur 'comp'...
Résultat intermédiaire : temp/hva_comp.csv
Résultat final : temp/hva_comp_somme.csv
Temps d'execution du traitement : 12 secondes
Le fichier ./temp/hva_comp.csv a ete genere avec succès dans le dossier temp.
Temps d'execution du graphique : 12 secondes

```

- Traitement -lv_comp

Donne les informations sur les stations LV pour les consommateurs « comp » triées par ordre croissant de capacité.

Après avoir testé sur les ordinateurs les plus performants que nous avons, le temps d'exécution de ce traitement est de 17 secondes.

```

• emma@Cookie:~/Projet-CY-Wire$ ./c-wire.sh data/c-wire_v25.dat lv comp
Le dossier temporaire existe. Vidage en cours...
Le dossier a été vidé avec succès.
Compilation de c-wire avec Makefile dans ./progC...
make: Nothing to be done for 'build'.
Compilation reussie.
Traitement en cours pour le type de station 'lv' et consommateur 'comp'...
Résultat intermédiaire : temp/lv_comp.csv
Résultat final : temp/lv_comp_somme.csv
Temps d'execution du traitement : 17 secondes
Le fichier ./temp/lv_comp.csv a ete genere avec succès dans le dossier temp.
Temps d'execution du graphique : 17 secondes

```

- Traitement -lv_indiv

Donne les informations sur les stations LV pour les consommateurs « indiv » triées par ordre croissant de capacité.

Après avoir testé sur les ordinateurs les plus performants que nous avons, le temps d'exécution de ce traitement est de 27 secondes.

```
emma@Cookie:~/Projet-CY-Wire$ ./c-wire.sh data/c-wire_v25.dat lv indiv
Le dossier temporaire existe. Vidage en cours...
Le dossier a été vidé avec succès.
Compilation de c-wire avec Makefile dans ./progC...
make: Nothing to be done for 'build'.
Compilation reussie.
Traitement en cours pour le type de station 'lv' et consommateur 'indiv'...
Résultat intermédiaire : temp/lv_indiv.csv
Résultat final : temp/lv_indiv_somme.csv
Temps d'execution du traitement : 27 secondes
Le fichier ./temp/lv_indiv.csv a ete genere avec succès dans le dossier temp.
Temps d'execution du graphique : 27 secondes
```

- Traitement -lv_all

Donne les informations sur les stations LV pour les consommateurs « all » triées par ordre croissant de capacité.

Après avoir testé sur les ordinateurs les plus performants que nous avons, le temps d'exécution de ce traitement est de 37 secondes, le temps d'exécution du graphique est de 1 seconde.

```
emma@Cookie:~/Projet-CY-Wire$ ./c-wire.sh data/c-wire_v25.dat lv all
Le dossier temporaire existe. Vidage en cours...
Le dossier a été vidé avec succès.
Compilation de c-wire avec Makefile dans ./progC...
make: Nothing to be done for 'build'.
Compilation reussie.
Traitement en cours pour le type de station 'lv' et consommateur 'all'...
Résultat intermédiaire par capacité croissante : temp/lv_all.csv
Résultat final : temp/lv_all_somme.csv
Fichier lv_all_minmax.csv généré avec succès dans le dossier temp.
Temps d'execution du traitement : 31 secondes
Le fichier ./temp/lv_all.csv a ete genere avec succès dans le dossier temp.
Création des graphiques...
Graphique généré : ./graphs/graphique_lvminmax.png
Temps d'execution du graphique : 1 secondes
```

Pour le graphique, le code a été fait dans la partie script shell en utilisant le fichier de sortie créé par le programme C (traitement_s.png).

(Voir annexe pour avoir un exemple du graphique obtenu).

Le dossier « temp » vous permettra de voir les résultats obtenus pour chaque traitement avec le tri effectué (.csv)

Le dossier « graph » vous permettra de voir le graphique généré (.png).

Annexe :