

## Semaine 1 : Planification et Mise en Place

### Objectifs :

- Analyser le script Shell et le code C existants.
- Définir les interactions entre le script et le programme en C.

### Tâches :

- Étudier le fichier main.c pour identifier les fonctions principales :
  - o Parsing des fichiers CSV.
  - o Gestion de l'arbre AVL.
  - o Génération des fichiers de sortie.
- Étudier le script Shell (c-wire\_v25.dat) :
  - o Comprendre ses fonctionnalités.
  - o Identifier les dépendances et les interactions possibles avec le code en C.
- Configurer l'environnement sur Replit pour exécuter à la fois le programme en C et le script Shell.
- Définir les responsabilités :
  - o Melek : Code en C (AVL et logique principale).
  - o Liam : Analyse et amélioration du script Shell.
  - o Assaad : Automatisation avec le Makefile et tests globaux.

## Semaine 2 : Développement des Composants de Base

### Objectifs :

- Implémenter et tester les fonctionnalités principales en C et Shell.

### Tâches :

- Programme en C :
  - o Ajouter des tests unitaires pour les fonctions AVL.
  - o Valider le parsing des fichiers CSV avec différents jeux de données.
- Script Shell :
  - o Adapter ou améliorer le script Shell pour traiter les fichiers générés par le programme C.
  - o Ajouter des fonctionnalités telles que :
    - Vérification de l'existence des fichiers d'entrée.
    - Archivage ou backup automatique des fichiers de sortie.
- Intégration :
  - o Établir une communication entre le programme en C et le script Shell.
  - o Ajouter une commande dans le Makefile pour exécuter le script Shell après la compilation.

## Semaine 3 : Fonctionnalités Avancées et Intégration

### Objectifs :

- Intégrer les scripts Shell et le code C dans un workflow automatisé.

### Tâches :

- Programme en C :
  - o Ajouter une gestion avancée des erreurs (par exemple, si le fichier CSV est corrompu).
  - o Implémenter des options avancées en ligne de commande pour plus de flexibilité.

- Script Shell :
  - o Automatiser l'exécution du programme en C avec des paramètres dynamiques.
  - o Ajouter des logs pour tracer les étapes du traitement (entrée, sortie, erreurs).
- Automatisation globale :
  - o Tester le pipeline complet (entrée Shell -> programme C -> sortie Shell).
  - o Configurer le Makefile pour inclure toutes les étapes dans une seule commande.

#### Semaine 4 : Finalisation et Livraison

##### Objectifs :

- Finaliser le projet et livrer une solution complète.

##### Tâches :

- Tests finaux :
  - o Tester le programme et le script Shell avec des fichiers CSV de tailles variées.
  - o Simuler des scénarios d'erreurs pour valider la robustesse.
- Documentation :
  - o Rédiger un guide utilisateur expliquant :
    - L'exécution du programme en C.
    - Le fonctionnement du script Shell.
  - o Créer une documentation technique pour décrire :
    - La structure du programme en C.
    - Les étapes automatisées par le script Shell.
    - Le Makefile et ses commandes.
- Présentation et livraison :
  - o Organiser une démonstration pour présenter le pipeline complet.
  - o Livrer tous les fichiers nécessaires (code source, script Shell, fichiers de test, documentation).