

AMARCHA Mohamed
DABOUSSI Akram
D'HERIN Arthur
MAKOUNDIKA KIDZOUNOU Ifrel Rinel
NTYAM Kevin
ROUSSEAU Lenaïc
SCHWARTZMANN Victor

Phase de réimplantation d'un éditeur de liens : **Journal de bord**

13/12/2024 :

Dans un premier temps, nous avons réalisé les fonctions utiles pour l'étape 1 soit la lecture de fichier ELF : Nous avons notamment écrit des fonctions de lecture sur 32 bits ou 16 bits, en little endian ou big endian, et une fonction pour écrire le tableau e_ident de la structure Elf32_Ehdr. Nous avons ensuite écrit la fonction de lecture de l'en-tête de fichier. Deux approches différentes ont été implémentées.

Nous avons ensuite commencé l'étape 2 soit la lecture de la table de section.
En attendant d'avoir accès à notre dépôt Gitlab, nous avons créé un dépôt Git sur Github.
Nous avons également commencé l'infrastructure de tests.

16/12/2024 :

Nous avons tout d'abord corrigé des erreurs dans l'étape 1, la lecture du tableau e_ident étant erronée : nous faisons uniquement une lecture de certaines parties du tableau plutôt qu'une lecture complète.
De premiers tests ont été effectués à l'aide du fichier example1.o, et semblent montrer un bon fonctionnement de notre code.

17/12/2024 :

Nous avons décidé de réaliser une extension de l'étape 1 : nous accordons la possibilité d'afficher l'en-tête de fichiers ELF en 64 bits et/ou en little endian.
Nous faisons également en sorte qu'il soit possible de comparer à l'aide de diff notre version de readelf avec la vraie implémentation.

Nous nous sommes ensuite réparti les tâches de la manière suivante :

- Arthur et Rinel : tests
- Mohamed : étape 2
- Akram : étape 3
- Victor : étape 4

Nous avons une infrastructure de tests relativement établie.

18/12/2024 :

Nous avons dans un premier temps factorisé le fichier entete.c.
Les étapes 4 et 5 ont été réorganisées de la manière suivante :

- Lenaïc : étape 4
- Victor : étape 5

L'infrastructure de tests est améliorée.

De premières versions des étapes 2 et 3 ont été implémentées.

19/12/2024 :

Nous avons modifié la gestion des options grâce à la fonction `getopt_long`.

Nous avons également amélioré la lecture du contenu des sections.

L'infrastructure de tests continue d'être améliorée, et de premières versions des étapes 4 et 5 ont été implémentées.

20/12/2024 :

Nous avons dans un premier temps corrigé des libérations mémoire, ainsi que l'affichage des étapes 2 et 3.

Après avoir pris en compte les remarques faites lors de notre audit de code, nous avons également discuté de la création d'une structure `ELF32_FILE` contenant toutes les structures utilisées actuellement dans les fonctions de lecture et d'affichage.

Une première version du descriptif de la structure du code a commencé à être écrite.

Interruption pédagogique :

Lors de l'interruption pédagogique, nous avons commencé par factoriser la phase 1 du projet afin de pouvoir intégrer la structure `ELF32_FILE` contenant toutes les structures de `elf.h` dont nous nous servons. La structure du code a également été modifiée afin de séparer les fonctions de lecture/écriture et les fonctions d'affichage.

Nous avons également eu du mal à répartir les tâches convenablement pendant l'interruption pédagogique, notamment dû au fait que nous travaillions en distanciel.

Nous avons également discuté de la compréhension de chacun de la phase 2.

De premières implémentations des étapes 6, 7 et 8 ont été effectuées.

Nous avons repris l'écriture du descriptif de la structure de code afin de correspondre à la nouvelle organisation des fonctions du projet suite à la factorisation du code.

Après réflexion, nous avons décidé de changer de méthodologie pour les étapes de la phase 2 : au lieu de modifier les informations du fichier dans un buffer contenant toutes les informations, nous modifions plutôt les données dans la structure avant de réécrire le fichier. Nous avons également implémenté une première version de l'étape 9, et nous avons également débuté le lien avec le simulateur ARM.

6/01/2025 :

Nous avons écrit la description des tests et le mode d'emploi.

7/01/2025 :

Nous avons commencé les diapositives pour la soutenance.

Des bugs ont été corrigés et de nouveaux tests ont été ajoutés.

8/01/2025 :

Nous avons corrigé des bugs, notamment liés à l'étape 6, et ajouté de nouveaux tests. Nous avons également réparti nos tâches pour la soutenance, et avons préparé nos parties de notre côté.