

Description du code

Descriptif de la structure du code développé

(fonction et fichier correspondant)

Chaque fichier source correspond à une partie du sujet. Nous avons cependant en plus notre fichier principal nommé « main.c » dans lequel nous appelons les différentes fonctions développées afin de manipuler un fichier ELF.

Liste des fonctionnalités implémentées et manquantes

Header.c permet de gérer l'en-tête du fichier ELF et comporte deux fonctions principales :

- *Elf32_Ehdr recupEnTete(FILE* f)* : permet de récupérer l'en-Tete d'un fichier Elf et de stocker le résultat dans une structure Elf32_Ehdr, structure fourni par elf.h.
- *void afficheEnTete(Elf32_Ehdr header_elf)* : Permet d'afficher l'en-Tete d'un fichier Elf.

TableauSections.c permet de gérer les en-têtes de sections

- *void recupTabsection(FILE* f, Elf32_Ehdr header_elf, Elf32_Shdr *t)* : permet de récupérer les en-Tetes de section d'un fichier Elf et de stocker le résultat dans un tableau de structure Elf32_Shdr, structure fourni par elf.h.
- *afficheTabsection(Elf32_Shdr *section_elf, Elf32_Ehdr header_elf, FILE* elf)* : Permet d'afficher les en-têtes de sections d'un fichier elf.

ContenuSection.c permet de gérer le contenu d'une section

- *void afficheContenuSection(Elf32_Shdr* tabEnTeteSection, int numberSection, FILE* elf)* : Permet d'afficher le contenu du section passé en paramètre.

TableauSymboles.c permet de gérer la table des symboles

- *int getTableauSymboles(FILE* file, Elf32_Ehdr header_elf, Elf32_Shdr *section_elf, Elf32_Sym **symbole_elf)* : permet de récupérer, grace à un passage par adresse, la table des symboles d'un fichier elf.
- *void afficheSymbole(FILE* elf, Elf32_Ehdr header_elf, Elf32_Sym *symboltable, Elf32_Shdr *section_elf, int size)* : Permet d'afficher la table des symboles d'un fichier ELF passé en paramètre.

TableReimplantation.c permet de gérer la table de réimplantation

- *TableReimplantation* recupTabReimplantation(FILE* file, Elf32_Ehdr header_elf, Elf32_Shdr *section_elf)* : Cette fonction permet de récupérer la table de réimplantation d'un fichier elf.
- *void afficheTabReimplantation(Elf32_Sym *SymbolTable, FILE* elf, Elf32_Ehdr header_elf, Elf32_Shdr *SectionTable, TableReimplantation* pointeur_reimplantation, int longueurTab)* : Cette fonction permet d'afficher la table de réimplantation d'un fichier elf

Les fonctionnalités qui vont nous manquer sont toute celle de la 2ème phase de développement que nous avons malheureusement à peine eu le temps de regarder. Nous n'avons seulement développé qu'une fonction permettant de copier des structures d'en-têtes de fichier, de section et d'en-têtes de sections en éliminant toutes les sections inutile.

Nous aurions également voulu rajouter un niveau d'abstraction dans notre main comme nous l'on suggérer les professeurs lors de l'audition de code, ainsi qu'une structure regroupant tous les différents éléments de notre phase de développement.

Liste et descriptions des tests effectués :

La plupart des tests effectués ont été effectués à partir des fichiers d'exemples fournies avec le sujet. Nous avons ainsi pu comparer les résultats de notre programme avec la commande *readelf*.

Ce sont les seuls tests que nous avons effectués étant donné que nous n'avons malheureusement pas réalisé la 2ème partie du sujet.

Répartition des tâches :

Nous étions quatre membres dans le groupe. Nous nous sommes donc répartis les tâches en binômes. Le plus souvent, les couples étaient Lucas/Abdul-Dramane et Jules/Valentin. Chaque binôme travaillait sur une partie différente.

Il arrivait de temps en temps que nous « cassions les binômes » si l'un des deux étaient bloqués pour venir en aide à l'autre.

Les journées se terminaient généralement par une série de test, du débogage et par la rédaction du journal de bord.