



**Université
Gustave Eiffel**

PROGRAMMATION C AVANCEE

COMPTE RENDU DE TP

TRAVAUX PRATIQUE NUMERO : 8

UYAR ORHAN

TP8

Le but de ce TP est d'élaborer un mécanisme automatique de bravement de plugins sur un programme relativement simple.

Pour ce TP on prend comme point de départ le TP3 dans le quel nous avons du code une calculatrice en Polonais inversé.

L'idée ici étant de créer des plugins pour toutes les opérations que nous avons déjà codées au préalable et rendre notre nouveau programme ainsi plus dynamique de façon à pouvoir être capable d'utiliser ces plugins externes.

Pour cela nous avons commencé par récupérer le code du TP3. Après quelque petit nettoyage et amélioration du fait de nouvelle compétence acquise en cours de route.

On commence par voir les différentes méthodes que nous avons besoin pour réaliser cette modification. À l'aide des différentes structures présentes dans le sujet, on remarque assez rapidement quelque structure que nous aurons besoin afin de manipuler ces structures.

Mais avant tout on commence par la création des plugins. Car malgré le non usage pour l'instant. Essayer de construire ces plugins permet de mieux comprendre la manipulation de celui-ci.

Après voici les différents modules que nous avons dû mettre en place.

D'abord dans le fichier `plugin/plugin.c` on a mis en place toute la structure et méthode qui vont nous permettre de manipuler nos différents plugins.

Il contient notamment les différents malloc des plugins et des free sur la liste de plugins. Et une méthode qui permet de libérer l'espace mémoire alloué lors de l'utilisation de ce module.

Petit modification dans la structure donnée lors de la réalisation du TP on stocke directement le `Lib_Handel` du plugin dans la structure car lors de l'utilisation de la méthode faut que le lien soit toujours actif. C'est pourquoi on décide de le stocker dans la structure et de le fermer à la fin du programme.

Pour la deuxième partie qui est la récupération des plugins. Ça se passe dans le fichier `pluginLoader/pluginLoader.c`

Ce module a pour but de parcourir le dossier plugins et d'être capable de récupérer les différents fichiers présents dans le dossier et de construire les Plugins grâce à la définition précédente.

Pour récupérer les fichiers, on utilise `scandir`. Sans oublier que pour tout fonctionner correctement quelque `Marco` devront être définis comme `#define _GNU_SOURCE` avant les includes.

`Scandir` nous permet aussi la mise en place d'un filtre qui nous permet de limiter les fichiers recherchés. Notre méthode filtre simplement tout les noms de fichiers possédant l'extension « `.so` » De nombre test son effectuer afin de détecter lors d'une erreur, la provenance de celui-ci.

Dernière modification est au niveau de notre calculatrice. On procède à la modification afin que les différentes opérations soient réalisées à partir d'un plugin chargé dans notre liste de plugins.