

TD 5 - Compilation et débogage

Chargez le fichier `td.tgz` dont l'emplacement vous sera communiqué par votre enseignant. Décompressez-le et désarchivez le dans votre répertoire en tapant la commande :

```
tar xzvf td3.tgz
```

Un répertoire `src` sera créé et contiendra les fichiers suivants : `constantes.h`, `liste.c`, `liste.h`, `outils_liste.c`, `outils_liste.h`, `prog0.c`, `prog0.h`, `prog1.c`, `prog1.h`, `prog2.c` et `prog2.h`

I Compilation

Déterminer et dessiner le graphe des dépendances.

Ecrire le Makefile permettant de compiler et de générer les programmes exécutables en respectant ce graphe de dépendance.

Ajouter les cibles permettant le nettoyage des fichiers générés : objets, exécutables, core, etc.

II Débogage

Rajoutez lors des compilations avec votre Makefile une option permettant le débogage. Lancez les exécutables ainsi créés par votre Makefile. Vous constaterez que :

- le programme `prog0` marche correctement.
- le programme `prog1` plante sur un problème mémoire.

Tapez :

```
ulimit -c 1000
```

dans votre shell afin d'activer la création d'une image de votre mémoire sous forme de fichier `core`. Relancer votre programme.

Analysez le fichier `core` à l'aide de `gdb` ou de `ddd` afin de déterminer les fonctions ayant conduits à l'erreur. Corrigez.

- le programme `prog2` ne plante pas mais se comporte curieusement. En suivant pas à pas son exécution à l'aide de `gdb` ou de `ddd`, déterminer quelle erreur est faite.

III Bibliothèque

1. Créez une bibliothèque statique `libliste.a` et modifiez votre Makefile en conséquence.
2. Créez une bibliothèque dynamique `libliste.so` et modifiez votre Makefile en conséquence.

Pour l'exécution, vous aurez à préciser le chemin de cette bibliothèque dynamique en positionnant la variable d'environnement `LD_LIBRARY_PATH`