

Travaux Dirigés Programmation C avancée

Informatique 1ère année.

—Julien Allali - allali@enseirb-matmeca.fr —

Code Legacy

L'objectif des deux dernières séances de TP est de mettre en oeuvre l'ensemble des concepts vus pour la mise au propre et l'amélioration d'un code existant.

Pour cela, vous allez devoir repartir des sources de votre projet de premier semestre. Il ne faut pas re-écrire le code depuis zéro mais faire évoluer celui-ci. Si vous le souhaitez, vous pouvez partir du code d'un autre groupe (c'est conseillé pour les élèves ayant déjà un bon niveau de programmation).

Les étapes que nous allons suivre sont les suivantes :

- Réorganisation des sources :
 - Créer les répertoires trunk, branches et tags dans le dépôt.
 - Déplacer vos sources dans le trunk et faire une copie du "trunk" dans "tags/0.0.0".
 - Dans trunk, créer les répertoires `src`, `tests` et `doc`.
 - Déplacer vos sources dans `src`.
- passage sous cmake :
 - Vous devez ré-écrire votre système de compilation automatique avec cmake.
- ajout d'un fichier README.md (format Markdown) digne de nom!

- ajout du support doxygen :
 - Dans le répertoire `doc`, ajouter et configurer votre fichier `Doxyfile`.
 - Vous pouvez vous inspirer de la bibliothèque "statelib" pour :
 - les options à placer dans `Doxyfile` (utilisation de la commande `diff`?)
 - l'ajout de la génération de la doc dans `cmake`
- charger le projet dans un IDE : à partir de ce point, vous devez travailler dans un IDE (qtcreator, codeblocks par exemple).
- On répète plusieurs fois :
 - documentation du code
 - écriture de premier tests
 - lancement des tests dans `valgrind` et `debug` dans `gdb` si nécessaire
 - mise au propre du code
 - lancement des tests dans `valgrind` et `debug` dans `gdb` si nécessaire

Au cours de ce travail, vous pouvez bien entendu revoir et/ou refaire vos API. Cependant, vous devez toujours avancer par petites étapes et bien veiller à ce que votre projet reste stable entre deux modifications. Profitez en pour créer des branches.