

Instructions pour le rendu des TPs

La vérification des TP se fait par l'intermédiaire d'un script. Il vous faudra donc impérativement respecter les règles ci-dessous pour permettre cette automatisation.

1. Le rendu se fera sous la forme d'une archive (TPi_LOGIN1_LOGIN2.tar.gz ou TPi_LOGIN.tar.gz si vous êtes seul).
2. L'arborescence de votre répertoire (contenu de l'archive) doit être

```

TP_LOGIN1_LOGIN2/
├── CMakeLists.txt
├── README.md
├── doc/
│   ├── CMakeLists.txt
│   ├── analyse.txt
│   └── doxyfile.rc
├── src/
│   ├── CMakeLists.txt
│   └── *.cxx *.hxx
├── test/
│   ├── CMakeLists.txt
│   └── *.cxx *.hxx
├── cmake
└── build/

```

Le Makefile doit contenir les cibles all, check, doc, clean

3. Vérification de votre projet (nécessite Python 2.7 ou supérieur) :
Tout au long de votre développement : Lancez régulièrement le script `verifier.py TPi_LOGIN1_LOGIN2` afin de vérifier que votre dossier est conforme aux instructions données
Avant l'envoi : Vérifiez que votre archive est lisible et que votre code sera testable par le correcteur : lancez `verifier.py TPi_LOGIN1_LOGIN2.tar.gz` et vérifiez l'absence d'erreurs.
4. **Documentation** : A la racine de votre projet, 'make doc' doit générer une documentation valide (`doc/html/*.html`)
5. Noms des fichiers pour la phase d'implémentation :
TP_1 et TP_2 Dvector.h et Dvector.cpp
Aucun fichier complémentaire ne doit être requis (pas de utilitaire.h/cpp communs).
6. Nom des classes et des fonctions : Respecter scrupuleusement les énoncés.
7. **Gestion des erreurs** : Si vous souhaitez gérer des erreurs (cas limites) non demandées dans l'énoncé, faites-le par l'intermédiaire d'exception (avec la méthode `throw`)