

---

## TP Outil de développement logiciel : build #1

---

### Rendre une archive par TP sur Celene ; un rendu par étudiant

## Consignes générales

- Tout ce qui est nécessaire au TP est installé dans la VM. Il y a un compte "sti" mot de passe "sti"
- Vous rendez vos rapports pour le contrôle continu sur Celene, avant minuit le jour du TP, sous forme d'archive contenant un rapport en pdf ainsi que tout code utile

## Prologue au TP

Supposons que vous soyez un(e) passionné(e) de chiffres, et que vous vouliez programmer un petit quelque chose qui vous permet de calculer le plus petit commun multiplicateur (Least Common Multiple, ou LCM). Comme vous détestez coder, vous avez récupéré l'implémentation du Projet Rosetta. Celle-ci se base sur un algorithme de calcul du ppcm, ou gcd (Greatest Common Divisor) en anglais. Vous avez mis tout ça au propre dans un répertoire de la VM, sous votre compte sti : `~/TP_Arithmetics/C`

## 1 Un peu de doxygen pour s'échauffer

Vous connaissez déjà doxygen, d'ailleurs vous avez inséré des commentaires doxygen dans le code du projet Arithmetics.

**Question** Générez la doc du projet arithmetics en C. Il y a une erreur dans le doxygen. Corrigez et générez une doc correcte, indiquez l'erreur dans votre rapport et les commandes pour générer la documentation.

## 2 Un peu de make pour se préparer à la suite

Vous connaissez déjà Make. Make est un outil qui permet d'automatiser la compilation et la construction de binaire. L'idée est simple, dans un fichier Makefile (le nom est toujours le même) on définit des cibles, toujours suivant la même forme :

```
cible : dépendances
commande
```

Chaque commande est précédée d'une tabulation. Lors de l'utilisation d'un tel fichier via la commande make la première règle rencontrée (ou la règle dont le nom est spécifié), est évaluée. L'évaluation d'une règle se fait en plusieurs étapes :

Les dépendances sont analysées, si une dépendance est la cible d'une autre règle du Makefile, cette règle est à son tour évaluée. Lorsque l'ensemble des dépendances est analysé et si la cible ne correspond pas à un fichier existant ou si un fichier dépendance est plus récent que la règle, les différentes commandes sont exécutées. Pour un tutoriel complet, voir par exemple chez [developpez.com](http://developpez.com) dont est tiré ce court extrait.

**Question** Ecrivez un Makefile pour compiler le projet Arithmetics, construire le binaire ainsi que la documentation doxygen. Donnez le Makefile dans votre rapport.

**Question** D'après vous, ce Makefile est-il portable, c'est-à-dire, utilisable sur un autre système que le vôtre ?

## 3 Cmake, un outil de build portable

Comme vous avez pu le constater, écrire des Makefile est ennuyeux, potentiellement complexe, et pas très portable. Vous voudriez être un peu plus général quand vous écrivez des fichiers : CMake est un outil qui permet de produire des Makefile automatiquement.

---

Quand on utilise CMake, on écrit CMakeLists.txt qui contient un ensemble de commande que CMake va exécuter. Vous trouverez un bon tuto pour les bases de CMake par exemple : [ici](#).

**Question** Mettez en place dans le répertoire CMake votre *CMakeLists.txt* pour pouvoir produire le binaire de vos sources. Donnez le CMakeLists.txt

**Question** Ajoutez à votre *CMakeLists.txt* le support pour produire votre documentation. Vous pouvez adapter ce qui est dans le répertoire CMake. Donnez le CMakeLists.txt

**Question** Transférez depuis la VM le dossier CMake vers la machine hôte et tentez de construire votre binaire. Que se passe-t-il ?

## 4 Ant

Vous trouvez vraiment que make, doxygen et CMake sont des outils un petit peu datés. Vous vous dites que Java + javadoc + Ant serait une super bonne alternative ! Vous vous décidez donc à reprendre tout de zéro et recoder le tout en Java. Vous avez mis tout ça au propre dans un répertoire de la VM, sous votre compte :

~/TP\_Arithmetics/Java

- Importez le projet dans Eclipse
- Générez un script Ant pour le projet
- Copiez votre projet dans un répertoire, et exécutez Ant à la main
- Générez la javadoc.

**Question** Quelles sont les cibles créées par défaut dans le build.xml ?

**Question** Donnez la ligne de commande pour exécuter Ant et son résultat dans votre rapport.