

# Projet de Java

Ingénieur 2000 – Informatique Réseau – 1<sup>ère</sup> année

Rémi Forax, Philippe Finkel

([forax@univ-mlv.fr](mailto:forax@univ-mlv.fr), [pfinkel@cantor.fr](mailto:pfinkel@cantor.fr))

## Description du projet

Le programme DarkProject (désolé pas trouvé mieux) est un “obfuscator” de bibliothèque et programme Java. Un obfuscator est un programme qui essaye d'éviter la décompilation ou le “reverse engineering” d'un programme compilé.

Il existe déjà un certain nombre d'obfuscator utilisant des techniques plus ou moins compliquées ayant trait à la compilation.

DarkProject utilisera une technique simple effectuant le renommage :

des champs

des méthodes

des classes (et peut-être des paquetages)

## Ligne de commande

```
java -jar Darkproject.jar [-mode] [-mangler] in.jar out.jar
```

java -jar Darkproject.jar permet de lancé le programme.  
in.jar correspond au jar à “obfusquer”, out.jar au résultat.

Le programme DarkProject possède deux modes de fonctionnement. Un mode destiné à l'obfuscation de bibliothèque (**-library**) et un mode destiné à l'obfuscation de programme (**-application**). le mode application est le mode par défaut.

Le mangler correspond à la stratégie de renommage choisie, les stratégies suivantes existes :

**-default**: les méthodes, champs et classes sont renommés séparément. Le premier noms trouvé a pour valeur a0, le second a1 etc. C'est la stratégie choisie par défaut.

**-letters**: même chose que **default** mais les valeurs sont a,b,c etc.

## Calendrier

Ce projet est à faire par binôme (cela veut dire deux personnes pas trois ni une).

Le rendu du projet est découpé en deux parties :

1. Pré-rapport indiquant l'architecture objet et tests d'utilisation.  
Date de rendu : le 1 février, avant minuit.
2. Rendu du projet en lui-même.  
Date de rendu : le 27 février, avant minuit.

Chaque rendu s'effectuera par mail sous forme d'une pièce jointe au format ZIP contenant l'ensemble des programmes et documents. Le mail devra être envoyé avant minuit aux deux adresses données au début de ce document.

## Détail du premier rendu

Le premier rendu est composé de deux documents distincts :

- un rapport d'architecture détaillant l'architecture objet proposé pour le programme
- un rapport de tests contenant des exemples de code commenté des tests à effectuer

Ces deux documents doivent être au format PDF.

L'architecture générale du projet (au moins 10 pages) :

Une introduction sur le format du byte-code (2 pages),  
limitez vous à ce qui est intéressant pour la suite

Répondre aux questions :

- pourquoi certaine méthode ne doivent pas être renommée ?
- pourquoi on "obfusque" pas de la même façon un programme et une bibliothèque ?
- quel est le problème induit par le renommage des classes ?
- quelle solution proposez vous pour palier à ce problème
- quelles sont les instructions du byte-code qu'il va falloir renommer ?

Doit suivre ensuite :

- le découpage en différentes classes (il y a environ 10 classes ici),
- le rôle de chaque classe
- l'ensemble des méthodes publiques de chaque classe

Les tests, voici l'ensemble des tests à effectuer :

- installation de la bibliothèque ASM et utilisation de celle-ci :
  - renommage du noms des champs d'une classe à partir du fichier .class
  - renommage du noms des méthodes d'une classe à partir du fichier .class
- ouverture d'un .jar et extraction des .class contenu
- recopie du contenu d'un .jar vers un autre
- création d'un script ANT de compilation.
- création d'un Jar exécutable.

Chaque test doit être accompagné d'un texte expliquant les concepts manipulés, les instructions pour exécuter le test et le code pertinent du test (le plus clair possible).  
Essayer de faire un document cohérent et lisible, SVP.

## Détail du second rendu

Le second rendu sera détaillé plus tard.

## Références

- Java Virtual Machine Spec – Class File Format (chapitre 4)  
<http://java.sun.com/docs/books/vmspec/2nd-edition/ClassFileFormat-final-draft.pdf>
- Java Virtual Machine Spec – Instruction Set (chapitre 6)

- <http://java.sun.com/docs/books/vmspec/2nd-edition/Instructions2.1.pdf>
- ASM – 1.5.2  
<http://asm.objectweb.org/>
- Jar exécutable  
<http://java.sun.com/j2se/1.5.0/docs/guide/jar/jar.html>
- API  
java.util.zip & java.util.jar
- Ant  
<http://ant.apache.org/>  
<http://ant.apache.org/manual/>