

# INFO-F201 – Systèmes d'exploitation

## Projet 1 : "shell scripting & md5"

Année académique 2017–2018

### Introduction

MD5 est l'abréviation de "Message Digest 5", il s'agit d'une fonction de hachage. Une fonction de hachage reçoit en entrée un message (qui peut être un fichier) de taille arbitraire et produit ce que l'on appelle une empreinte, de taille fixe. Ici, il s'agit d'une fonction de hachage cryptographique, c'est-à-dire qu'elle a été conçue de manière à être à sens unique : créer l'empreinte doit être relativement efficace mais, à empreinte donnée, trouver un message qui la produise doit être difficile. C'est ce qui est utilisé dans Linux pour les mots de passe : on stocke l'empreinte du mot de passe plutôt que le mot de passe lui-même. MD5 produit une empreinte de 128 bits. Il y a donc  $2^{128}$  empreintes possibles. Bien que deux fichiers différents puissent avoir la même empreinte, ceci est peu probable de manière non intentionnelle. Des travaux ont compromis MD5 qui ne devrait dès lors plus être utilisé pour des applications où la sécurité est en jeu. Néanmoins, l'algorithme est largement répandu, disponible sous de nombreux systèmes (voir md5sum sous Linux) et ce projet ne traite pas de sécurité.

### Énoncé

Un mélomane réalise que, les années passant, au gré des ajouts, sa bibliothèque MP3 comporte des doublons. Il possède souvent différentes versions d'un même morceau et il arrive parfois que certaines soient identiques, au bit près ! Soucieux de faire le ménage dans sa bibliothèque, il vous charge de réaliser un script lui permettant de détecter ces fichiers identiques. Le script prend le chemin du répertoire de la bibliothèque comme unique paramètre. Voici un exemple d'appel à votre script :

```
#./project1.sh /home/cternon/mp3/
```

En sortie, le script affiche le chemin des fichiers identiques, séparés par des virgules, tout comme le format CSV. Les différents ensembles de fichiers identiques sont séparés par un passage à la ligne. Si les fichiers a.mp3 et b.mp3 sont identiques et que c'est également le cas des fichiers c.mp3 et d.mp3, la sortie correspondra à ceci :

```
/home/cternon/mp3/chantsBulgares/a.mp3,/home/cternon/mp3/compils/voitureIV/b.mp3  
/home/cternon/mp3/divers/c.mp3,/home/cternon/mp3/aTrier/2001/d.mp3
```

### Consignes

Vous pouvez utiliser toutes les commandes vues aux travaux pratiques ainsi que celles citées dans l'énoncé. La clarté, la lisibilité, la qualité d'implémentation et l'efficacité pourront servir de

critères d'évaluation. Commentez donc intelligemment votre code et implémentez ce qui est demandé de manière claire et précise. Le *shell* pour lequel le script doit tourner pour ce projet est bash. Si vous avez une question à propos de l'énoncé, elle est à adresser à Cédric Ternon : [cter-non@ulb.ac.be](mailto:cter-non@ulb.ac.be)

### Consignes pour la remise du projet

*À respecter scrupuleusement !*

1. Votre projet doit indiquer **votre nom** et **votre numéro de matricule** (par exemple en commentaires au début de chaque fichier).
2. Votre projet doit être **dactylographié**. Les projets écrits à la main ne seront **pas corrigés**.
3. Votre code doit être **commenté**.
4. Si votre code ne s'exécute pas, votre projet ne sera **pas corrigé**.
5. Le plagiat sera sanctionné comme il se doit.
6. Aucune défense orale du projet n'est prévue. Toutefois, l'étudiant pourra être convoqué pour fournir des explications dans les semaines qui suivent la remise.
7. Vous devez respecter les modalités de remise suivante :
  - Date de remise : **le mercredi 13 décembre 2017**
  - Lieu : **boîte aux lettres du secrétariat « étudiants » du département d'Informatique, local 2N8.104**
  - Heure : **avant 18h00**
  - Une version électronique est à poster sur l'université virtuelle.**Après 18h30**, les projets seront considérés comme **en retard**, et vous aurez **0 points** sur votre note finale de projet.