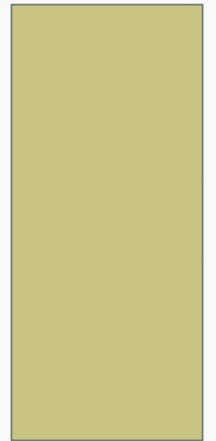


TP 3

SYSTEM INFORMATIQUE



INFORMATIONS GÉNÉRALES

- Références au cours :
 - 4.Intro_c
- Date de reddition : 16 octobre 2018 à 23:59
- **Délivrables**
 - Code source commenté
 - Dossier à rendre : <Prenom1>.<Nom1>.< Prenom2>TP3.zip(ou tar.gz)

OBJECTIF

L'objectif général du TP est de créer un programme qui permet de générer des hashes, également appelés digests (MD5, SHA1, etc.) à partir de chaînes de caractères ou de fichiers. Le but des digests est de pouvoir contrôler l'intégrité de données transmises entre deux entités.

Les objectifs pédagogiques de ce TP sont de:

- manipuler des chaînes de caractères en C;
- manipuler les paramètres `argc` / `argv` de la fonction `main`;
- se familiariser avec la fonction `getopt` qui permet de manipuler les paramètres passés au programme;
- effectuer un lien vers des bibliothèques externes (ici la bibliothèque EVP de openssl);
- se familiariser avec les fonctions de hachage qui est un mécanisme de base de la cryptographie.

EXERCICES 2.1

Exercice: Votre première tâche consiste à vous familiariser avec deux outils de calcul de hachage *sha1sum* et *md5sum*. Ces outils vous permettront par la suite de vérifier que votre programme fonctionne correctement. Afin de vous familiariser avec ces deux programmes:

- lisez le manuel des commandes *sha1sum* et *md5sum*;
- créez un fichier contenant le texte “Le manuel disait: Nécessite Windows 7 ou mieux. J’ai donc installé Linux” sans retour à la ligne. Utiliser les commandes ci-dessus pour calculer les hashes du contenu du fichier.
- combiner la commande *echo* “*Le manuel disait: Nécessite Windows 7 ou mieux. J’ai donc installé Linux*”. avec les commandes ci-dessus pour calculer les hashes sans utiliser de fichier;
- le résultat est différent, pourquoi? Comment résoudre le problème?

EXERCICES 2.2

Exercice:

- lisez le manuel des fonction `EVP_Digest...` (`man EVP_DigestInit`);
- implémentez l'exemple proposé dans les pages du manuel;
- compilez le programme en prenant soin de lier les librairies `libssl` et `libcrypto`;
- testez le programme avec la phrase “Le manuel disait: Nécessite Windows 7 ou mieux. J’ai donc installé Linux.”

EXERCICES 3

Exercice:

- lisez le manuel de la fonction *getopt*;
- implémentez l'exemple proposé dans les pages du manuel et testez le.

EXERCICES 4

4 Intégration: le programme à réaliser

Le but de ce tp est de créer un programme permettant de générer des hash de fichiers et de chaînes de caractères. Le programme pourra donc être appelé de deux manières. Avec l'option `-f` les paramètres d'entrée seront considérés comme des fichiers et un hash code par fichier sera calculé. Sans l'option `-f` les paramètres du programmes seront considérés comme une unique chaîne de caractère. Ainsi:

```
1 hash -f fichier1 fichier2 ...
```

retournera par exemple (cela depend bien sure du contenu du fichier):

```
1 8f85e20c95a5a73b9670883fd2cfc5ecd23e9b8b      fichier1
2 57ba123423aad8a65a3a83066f0c7523787c211      fichier2
```

alors que:

```
1 hash -f ceci est une chaine
```