

TP 1&2

Contact : chen.wang@unige.ch
Rose.Defossez@etu.unige.ch
Kathleen.Favre@etu.unige.ch

octobre 2019

1 Buts

Durant ce TP, vous :

- vous familiariserez avec la commande 'man' et 'echo' ;
- apprendrez des opérations de lecture et de manipulation de fichiers en bash ;
- écrirez un script permettant d'extraire les images présentes dans un répertoire arbitraire et de les copier dans un nouveau répertoire en nettoyant leur nom et en normalisant leur format en 'PNG' ;

Votre travail devra être soumis sur Moodle d'ici le 7 octobre minuit au plus tard avec de nom : <Prenom>.<Nom>.TP1.zip(ou tar.gz). Le fichier compressé doit inclure votre code et votre rapport.

2 Exercices

2.1 Commande 'man'

Si vous connaissez un nom de commande mais que vous ne savez pas ce qu'il fait :

a) Utilisez la commande shell 'man' (man [section] <sujet>) pour afficher le manuel de la commande.

Essayez les commandes suivantes dans le shell :

```
$ man man
```

```
$ man echo
```

```
$ man head
```

b) Essayez de rechercher dans un moteur de recherche une commande : par exemple, essayez la recherche suivante : 'bash how to merge two files'.

2.2 Commande 'echo'

Quelle est la fonction de la commande 'echo' ? Quels sont les paramètres de cette commande ?

2.3 Créer un fichier

Comment créer un fichier vide à partir de la ligne de commande ? Quelle est la commande ?

2.4 Nautilus

Essayez d'utiliser la commande 'nautilus' dans le shell. Essayez d'utiliser cette commande avec des paramètres différents (par exemple, votre dossier Documents). Quelle est la fonction de cette commande ?

2.5 Fichiers

a) Quelles sont les différences entre les trois commandes suivantes :

```
$ wc -w < foo.txt  
$ cat foo.txt | wc -w  
$ wc -w foo.txt
```

b) Quelle est la différence entre les commandes suivantes :
\$ head foo.txt -n 6
\$ tail foo.txt -n 6

c) Exécutez la commande suivante :

```
$ sort foo.txt >out1.txt 2>out2.txt
```

Y'a-t-il une différence entre out1.txt et out2.txt ? Décrivez le contenu des deux fichiers out1.txt et out2.txt dans le cas où foo.txt existe et quand il n'existe pas.

d) Trouvez un fichier pdf sur votre ordinateur (par exemple foo.pdf). Trouvez un programme qui peut ouvrir des fichiers pdf (par exemple evince).
Quelle est la différence entre ces deux commandes :
\$ evince foo.pdf
\$ evince foo.pdf &

2.6 Écrivez un script

Votre programme devra respecter les points suivants. Votre note dépendra du nombre de fonctionnalités implémentées correctement et de la qualité de celles-ci.

- Le script prend trois arguments, deux obligatoires et un optionnel
 1. le répertoire dans lequel se trouvent les images
 2. le répertoire dans lequel il faut écrire les images
 3. la résolution de l'image désirée (argument optionnel)
- Les images ne seront pas effacées du répertoire de départ.
- Si le répertoire d'arrivée n'existe pas, il faut le créer
- Il faudra supprimer les espaces, apostrophes et guillemets des noms de fichiers dans l'image d'arrivée
- Seules les images seront copiées quel que soit leur format de départ (et leur extension).

3 Quelques conseils

Pour Exercice 2.6, vous êtes libre de choisir les commandes shell que vous préférez tant qu'elles font parties des commandes de base sur les postes étudiants. Voici néanmoins quelques conseils :

- vous pouvez tester le type réel d'un fichier avec la commande `file`. En utilisant l'option `-i` vous obtiendrez le type MIME qui débute toujours par `image/` lorsqu'il s'agit d'une image.
- la commande `convert` permet de convertir automatiquement une image d'un format à un autre, par exemple : `convert machin.gif machin.jpg` convertit une image du format GIF vers le format JPEG. L'extension est utilisée pour déterminer le format d'arrivée.
- la commande `convert` permet également de changer la taille en passant les dimensions souhaitées. Le ratio de l'image ne change pas. Par exemple : `convert -resize 1280x1024 machin.gif machin.jpg`.
- Découpez votre script en fonctions dès le début. Implémentez les une à une. Cela nous aidera aussi à corriger et évaluer votre programme...