Informatique *TD*
Système d'exploitation Ipesup

Exercice 1 - Arborescence et navigation

On travaille en tant qu'utilisatrice `alice` (groupe `mp2i`). Créer l'arborescence suivante.

```
/home/alice/work
|- projets
|  |- os
|  |  |- notes
|  |  |- tp
|  |- reseau
|     |- tp
|- donnees
   |- images
   |  |- raw
   |  |- a_imprimer
   |- textes
```

a) Depuis `/home/alice`, donner une commande (sans `cd`) qui affiche le contenu détaillé de `work/projets/os`.
b) Créer trois fichiers texte : `projets/os/notes/plan.txt`, `donnees/textes/lisez-moi.txt`, `donnees/images/raw/img_001.jpg`.
c) Donner un chemin absolu et un chemin relatif pour accéder à `img_001.jpg` depuis `projets/os/tp`.
d) Expliquer la différence entre chemins absolus et relatifs et le rôle de `.` et `..`
e) Lister avec une seule commande tous les fichiers `.jpg` situés immédiatement sous `donnees/images`.

Exercice 2 - Arborescence et navigation

On suppose que l'arborescence de l'exercice précédent est déjà créée.

1. Depuis `/home/alice/work/projets/os/notes`, afficher le contenu détaillé du dossier `a_imprimer` sans changer de répertoire.
2. Créer un nouveau dossier `archives` à l'intérieur de `donnees/textes`, en utilisant un chemin **absolu**.
3. Depuis le répertoire `reseau/tp`, créer un dossier `resultats` à l'intérieur de `donnees/images/raw`, en utilisant un chemin **relatif**.
4. Depuis `projets/os/tp`, afficher le contenu du dossier `donnees/textes` avec une seule commande, en utilisant un chemin absolu.
5. Depuis `donnees/images`, afficher le contenu du dossier `projets/os` avec une seule commande, en utilisant un chemin relatif.
6. Expliquer la différence entre exécuter `ls ..` et `ls ../..` quand on se trouve dans `donnees/images/raw`.

Exercice 3 - Permissions

On considère :

```
drwxr-x--- 2 alice mp2i 4096 sept. 1 10:00 donnees
drwxr-x--- 2 alice mp2i 4096 sept. 1 10:00 donnees/images
-rw-r----- 1 alice mp2i 123 sept. 1 10:00 donnees/textes/lisez-moi.txt
```

a) Interpréter pour un répertoire les droits r, w, x et leurs effets concrets.
b) Modifier les permissions pour que le groupe puisse créer/supprimer dans `donnees/images`, les autres n'y aient aucun accès, et `lisez-moi.txt` soit lisible par tous mais modifiable seulement par `alice`.
c) Expliquer pourquoi `chmod` sur `donnees` peut être nécessaire pour (b).

Exercice 4 - Inodes et liens physiques

```
-rw-r--r-- 1 alice mp2i 1300458 sept. 1 10:10 donnees/images/raw/img_001.jpg
```

a) Créer un lien physique nommé `photo.jpg` vers ce fichier dans `donnees/images/a_imprimer`, et vérifier les inodes.
b) Retirer les droits de lecture pour groupe/autres sur `photo.jpg` et observer l'effet sur `img_001.jpg`.
c) Supprimer `img_001.jpg` et montrer que les données existent toujours via `photo.jpg`.
d) Expliquer pourquoi un lien physique ne peut pas viser un répertoire ni traverser deux systèmes de fichiers.

Exercice 5 - Liens symboliques

a) Créer `projets/os/courant` comme lien symbolique vers `projets/os/notes`, et expliquer la sortie de `ls -l`.
b) Créer une chaîne de liens (`actuel`, `dernier`), puis exécuter `cd dernier` et interpréter le répertoire atteint.
c) Renommer `projets/os/notes` en `notes_old` et montrer ce qui arrive aux liens.
d) Discuter pourquoi les liens symboliques peuvent traverser les systèmes de fichiers et viser des répertoires.

Exercice 6 - Montage et systèmes de fichiers

a) Expliquer la différence accès par octet / par bloc et donner un exemple de périphérique.
b) En tant que root, monter `/dev/sdb1` dans `/mnt/usb` puis démonter proprement.
c) Comparer FAT32 et Ext4 (droits, méta-données, reprise sur panne) et discuter leur usage.
d) Expliquer les conséquences d'un échange entre Ext4 et FAT32 sur permissions et liens.

Exercice 7 - Redirections et pipes

a) Construire une pipeline qui liste tous les fichiers de `donnees`, trie par taille décroissante et affiche les 3 plus volumineux.
b) Rediriger la sortie standard de (a) vers `rapport_taille.txt` et les erreurs dans `erreurs.log`.
c) Produire via pipeline le nombre de fichiers listés et l'écrire dans `compte.txt` en ajout.
d) Expliquer la différence entre `>` et `>>`, et entre `>` et `2>`.

Exercice 8 - Motifs glob

Dans `donnees/images` on a :

```
img_001.jpg img_012.jpg img_120.png img_A23.jpg img_223.jpg
```

a) Donner l'expansion de `img_*.jpg`, `img_???.jpg`, `img_[0-2][0-9][0-9].jpg`, `img_[^A]*.jpg`.
b) Proposer un motif qui matche seulement les fichiers JPG dont le numéro est strictement < 200.
c) Expliquer où a lieu l'expansion des motifs et prouver avec `echo`.

Exercice 9 - Projet de synthèse

Objectif : organiser un dépôt de cours sans duplication inutile, avec rapports reproductibles.

a) Construire une arborescence `cours/os` avec `chapitres/01`, `chapitres/02`, `exos`, `figures`, et créer quelques fichiers.
b) Créer un sous-répertoire `exos/a_imprimer` contenant des liens physiques vers tous les `.txt`.
c) Créer un lien symbolique `cours/os/courant -> chapitres/02` et montrer son fonctionnement.
d) Générer `rapport.txt` contenant : plus grande entrée, nombre de fichiers `.txt`, liste des liens avec inodes/cibles. Utiliser uniquement commandes de base (`ls`, `ln`, `echo`, `sort`, `wc`, `head`, pipes, redirections).
e) Ajouter un court paragraphe sur : limites des liens physiques, avantages/inconvénients des liens symboliques, impact d'un montage FAT32.