

Introduction aux systèmes d'exploitation (IS1)

TP n° 5 : révisions

Le but de ce TP est de vous faire réutiliser les commandes que vous avez découvertes lors des TP précédents.

Exercice 1 – familiarisation avec le programme de contrôle


On rappelle que le fichier `~/ .bashrc` de votre compte au script doit contenir la ligne suivante (qui sera indispensable les jours des vrais contrôles) :

```
export LC_COLLATE=C
```

1. Ouvrir deux terminaux, de préférence côte à côte.
2. Dans un terminal, dit « *terminal de contrôle* » exécuter « `~is1p/factice` » avec l'option « `--clair` » ou « `--noir` » (option par défaut) selon que votre terminal a un fond clair ou foncé.



Cela crée un répertoire `~/FACTICE`, qui sera la racine de l'arborescence manipulée ensuite, puis une première question s'affiche.

3. Dans l'autre terminal, dit « *terminal de travail* », se placer dans `~/FACTICE`.
4. Répondre ensuite aux questions posées par le programme : exécuter les commandes nécessaires dans le terminal de travail, et donner les réponses éventuelles dans le terminal de contrôle.

Avant de passer aux exercices suivants, télécharger depuis Moodle le script `tp5.sh`, puis faire ce qu'il faut pour pouvoir l'exécuter dans votre répertoire `~/Cours/2023/IS1` et l'exécuter. Cela crée une arborescence de racine `~/Cours/2023/IS1/TP5`, dans laquelle tous les exercices suivants devront être effectués. Créer dans ce répertoire un fichier `reponses_TP5.txt` pour répondre aux questions marquées par le symbole .

En fin de TP, créer une archive `arborescence_tp5.tar` de votre répertoire `~/Cours/2023/IS1/TP5` (contenant en particulier le fichier `reponses_TP5.txt`), et déposer le fichier obtenu sur Moodle.

Exercice 2 – chemins relatifs et absolus

1. Afficher toute l'arborescence de `~/Cours/2023/IS1/TP5/Chemins_relatifs`, et la dessiner sur une feuille de papier.
2.  Quel est le chemin absolu du fichier ordinaire nommé `toto.sh` qui se trouve dans le (seul) répertoire dont le nom de base est `E` ? L'exécuter dans la console en utilisant son chemin absolu. Quel message s'affiche ?
3.  Depuis chacun des répertoires de travail suivants, exécuter **ce même fichier** `toto.sh` (présent dans le répertoire `E`) en utilisant sa référence relative :

- a. `~/Cours/2023/IS1/TP5/Chemins_relatifs/A/B/E/`
- b. `~/Cours/2023/IS1/TP5/Chemins_relatifs/A/B/`
- c. `~/Cours/2023/IS1/TP5/Chemins_relatifs/A/B/F/`
- d. `~/Cours/2023/IS1/TP5/Chemins_relatifs/A/B/G/`
- e. `~/Cours/2023/IS1/TP5/Chemins_relatifs/A/C/`
- f. `~/Cours/2023/IS1/TP5/Chemins_relatifs/A/`
- g. `~/Cours/2023/IS1/TP5/Chemins_relatifs/A/D/H/`

Exercice 3 – *jokers*

Le but de cet exercice est de sélectionner (puis d'afficher) une partie des fichiers de la sous-arborescence `~/Cours/2023/IS1/TP5/Jokers`.

1. Créer une copie de sauvegarde de l'arborescence `Jokers` sous le nom `CopieJokers` (dans le répertoire `~/Cours/2023/IS1/TP5/`).
2. Lister le contenu du répertoire `Jokers`.
3. Déplacer dans son sous-répertoire `Corbeille` (tous) les fichiers (ordinaires) dont le nom commence par une voyelle (et seulement ceux-là).
4. Faire de même pour ceux dont l'avant-dernier caractère est une majuscule.
5. Supprimer de `Jokers` les fichiers dont le nom est formé d'au moins 5 caractères.
6. Supprimer de `Corbeille` les fichiers dont le nom contient deux chiffres.
7. Remplacer dans `Jokers` les fichiers de `Corbeille` dont le nom contient un `r` ou un `v`.
8. Déplacer dans `Ronsard` les fichiers de `Jokers` dont le nom ne termine pas par une voyelle, mais en contient une.
9. Déplacer dans `Boileau` les fichiers de `Jokers` dont le nom commence par une minuscule, mais n'est pas formé uniquement de minuscules.
10. Vérifier que les affichages obtenus par les commandes « `cat Ronsard/*` » et « `cat Boileau/*` » vous évoquent quelque chose.

Si ce n'est pas le cas, reprendre l'exercice au début en recréant `Jokers` à l'aide de `CopieJokers`... (Et vérifier que le fichier `~/ .bashrc` contient la ligne `LC_COLLATE=C`.)

Exercice 4 – « *echo* », « *cat* » et les redirections

1. Visualiser le contenu du fichier `poeme_rimbaud`, puis exécuter la commande :
`echo "Oh! là là! que d'amours splendides j'ai rêvées!" > poeme_rimbaud`
 Visualiser maintenant le contenu de `poeme_rimbaud`.

2. Visualiser le contenu du fichier `chanson_vian`, puis exécuter la commande :
`echo -e "On n'est pas là pour se faire engueuler\nOn est là pour voir le défilé!" >> chanson_vian`
 Comparer.

3. À l'aide de « `cat` » et d'une redirection, créer en une seule ligne de commande un fichier `poeme_boileau` en concaténant tous les fichiers du répertoire `Jokers/Boileau`.

4. Créer un fichier `poeme_ronsard` contenant les 3 premiers vers de *Mignonne, allons voir si la rose*. Ajouter les 3 vers suivants en une seule ligne de commande.

Exercice 5 – manipulation d'arborescence

- Quelle est la profondeur de l'arborescence `Vide`? Supprimer toute l'arborescence.
- Afficher toute l'arborescence de racine `Multiliens` et la dessiner en précisant le numéro d'inoeud de chaque fichier.
Combien y a-t-il de fichiers ordinaires (distincts) dans cette arborescence?
- Déplacer les répertoires `Protegee/Amonbofis/C` et `Protegee/Batdaf/A` dans `Protegee/Cloridric`.
- Copier le répertoire `Protegee/Batdaf` (et son contenu) dans `Protegee/Amonbofis`.
- Supprimer le répertoire `Protegee/Batdaf`.
- Copier `Protegee/Cloridric` (et son contenu) dans `Protegee/Amonbofis`.

Exercice 6 – liens et répertoires

Déplacez vous dans le répertoire `IS1/TP5/Tri`

- Créer trois sous-répertoires `A`, `B` et `C`.
- Pour chaque fichier de l'arborescence de `Bazar/` qui commence par `a`, créer un lien dans le répertoire `Tri/A`. Faire de même pour les fichiers qui commencent par `b` et le répertoire `Tri/B` et ceux qui commencent par `c` et le répertoire `Tri/C` (le premier paragraphe du manuel de `ln` peut être utile).
- Toujours dans `Tri/`, créer trois autres sous-répertoires `Quatre`, `Cinq` et `Six`.
- Dans `Tri/Quatre`, créer un lien pour chaque fichier de l'arborescence de `Bazar/` qui contient exactement 3 lettres. Faire de même pour `Tri/Cinq` et `Tri/Six`.
- Créer un sous-répertoire `Special` dans `Tri/` puis créer dedans des liens vers tous les fichiers de l'arborescence de `Bazar/` qui contiennent au moins un caractère spécial, c'est-à-dire non-alphabétique et non-numérique.
- Quel est l'inoeud du fichier `bd9j8A`? Supprimer le fichier `bd9j8A` ainsi que tous les liens qui y font référence.

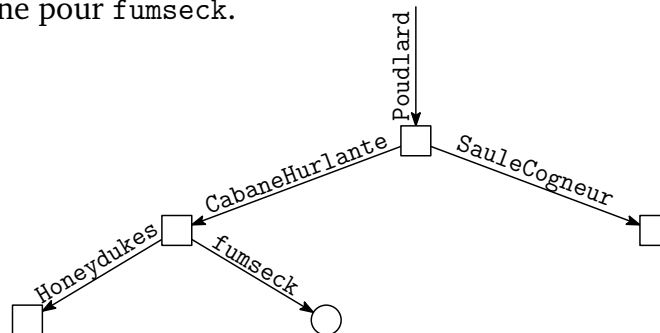
Pour aller plus loin : liens symboliques

Il est parfois utile de créer plusieurs liens sur un même fichier, pour en faciliter l'accès par exemple ; mais il arrive qu'il ne soit pas possible, ou pas adapté, d'utiliser des liens *physiques* comme vus précédemment. En particulier, on ne peut pas créer un lien physique sur un répertoire. Un substitut aux liens physiques est cependant disponible : les liens dits *symboliques* sont des fichiers spéciaux qui pointent vers une référence et non directement vers un i-nœud.

l'option « -s » de « ln » permet de créer des liens symboliques.

Exercice 7 – liens symboliques vs liens physiques

1. ➤ Depuis le répertoire ~/Cours/2023/IS1/TP5, créer l'arborescence suivante, où `fumseck` est un fichier contenant le texte `Voici un animal fantastique`. Pour cela, vous n'avez droit qu'à **deux** lignes de commande, pas une de plus ! – une pour **tous** les répertoires, une pour `fumseck`.



2. Créer dans `Poudlard` un lien physique `lp_fumseck` et un lien symbolique `ls_fumseck` vers le fichier `fumseck`. Comparer leurs contenus, puis leurs numéros d'i-nœud, ainsi que leurs autres caractéristiques.
3. Essayer de modifier les droits d'accès à `ls_fumseck`. ➤ Que constatez-vous ?
4. Modifier les droits d'accès au répertoire `CabaneHurlante` pour ne plus y avoir accès. Essayer d'afficher le contenu de `lp_fumseck` et `ls_fumseck`. ➤ Que constatez-vous ? Pourquoi ? Rétablir les droits.
5. Déplacer `fumseck` dans `SauleCogneur`, puis essayer d'afficher le contenu de `lp_fumseck` et `ls_fumseck`. ➤ Que constatez-vous ? Pourquoi ?
6. Créer un nouveau fichier `fumseck` dans `CabaneHurlante` contenant le texte
... et le phénix renaît de ses cendres!
Afficher le contenu de `lp_fumseck` et `ls_fumseck`. ➤ Expliquer.
7. Déplacer `ls_fumseck` dans `CabaneHurlante` puis tenter d'afficher son contenu. ➤ Expliquer.

8. Supprimer `ls_fumseck` de `CabaneHurlante`, puis créer un nouveau lien symbolique (valide) `ls_fumseck` vers `fumseck` dans `CabaneHurlante`. Déplacer ensuite ce lien dans `SauleCogneur`. 🐞 Expliquer le comportement observé.
9. Créer dans `Poudlard` un lien symbolique `passageSecret` vers `Honeydukes`, puis créer dans `SauleCogneur` un lien symbolique `tunnel` vers `CabaneHurlante`.
10. Donner trois manières différentes de se déplacer dans le répertoire `Honeydukes` à partir du répertoire `Poudlard`. En utilisant successivement ces trois méthodes, déplacez-vous dans le répertoire `Honeydukes` puis comparez les réponses de « `pwd` » et « `pwd -P` » ; remontez ensuite dans le répertoire parent à l'aide de la commande « `cd ..` ». 🐞 Que remarquez-vous ? Comparez avec « `cd -P ..` ».
11. Que se passe-t-il si on utilise la commande « `ls -R` » sur `Poudlard` ?
12. 🐞 Faire ce qu'il faut pour arriver à la situation (gênante !) suivante :

```
$ ls -l Honeydukes
total 0
lrwxrwxrwx 1 titi  staff  6  7 oct 18:41 fred -> george
lrwxrwxrwx 1 titi  staff  4  7 oct 18:41 george -> fred
```