

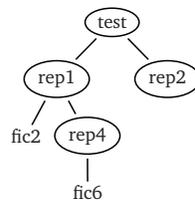
Introduction aux systèmes d'exploitation (IS1)

TP 3 – Liens, i-nœuds

Dans ce TP, nous allons explorer un peu plus en détail les possibilités du système de fichier UNIX à travers le concept de lien et la notion de numéro d'i-nœuds.

Exercice 1 – Préparons le terrain.

1. Pour commencer, créez l'arborescence suivante dans votre dossier IS1.



2. Donnez trois façons de désigner le fichier `fic6` depuis votre répertoire personnel `~`.
3. A l'aide de la commande `echo`, écrivez `Bonjour à vous jeune padawan` dans le fichier `fic6`.
4. A l'aide de la commande `cat` affichez le contenu du fichier `fic6` depuis le répertoire `rep2`.
5. L'option `-l` de la commande `ls` permet entre autre d'observer les droits d'un fichier/répertoire. A quoi correspondent les autres informations que l'on obtient grâce à cette commande ?

Liens

La commande `ln` sert à créer des *liens*. Il existe deux types de liens, les liens dits *physiques* (*hard links* en anglais), et les *liens symboliques* (*soft links* en anglais). Elle s'utilise de façon similaire à la commande `cp` :

```
ln [-s] source destination
```

- `source` est le nom du fichier sur lequel on crée un lien.
- `destination` est le nom du lien.
- L'option `-s` permet la création d'un lien symbolique. Par défaut, le lien créé est physique.

Exercice 2 – Créons des liens sur des fichiers.

1. Comme expliqué ci-dessus, la commande « `ln` » sert à créer des *liens*. Utilisez-la pour créer un lien **physique** du fichier `fic6` dans `test` sous le nom de `lpfic6`.
2. Modifiez le fichier `lpfic6` à l'aide de la commande `echo`. Que constatez-vous pour le fichier `fic6` ? Réciproquement, modifiez `fic6`, lisez `lpfic6`. Concluez.
3. Modifiez les droits d'accès au fichier `fic6` pour les membres du groupe. Que constatez-vous pour `lpfic6` ? Pouvez-vous avancer une explication ?

4. La commande « `ln` » peut aussi créer des liens *symboliques* avec l'option `-s`. Créez un lien symbolique du fichier `fic6` dans `test` que vous appellerez `lsfic6`.
5. Regardez toutes les informations concernant les fichiers `lpfic6` et `lsfic6`. Quelles différences notez-vous ?
6. Essayez de modifier les droits d'accès au fichier `lsfic6`. Que constatez-vous ?
7. Modifiez les droits d'accès au répertoire `rep1` pour ne plus y avoir accès. Essayez d'afficher le contenu de `lpfic6` et `lsfic6`. Que constatez-vous ? Pouvez-vous avancer une explication ?
8. Modifiez de nouveau les droits d'accès au répertoire `rep1` pour y avoir de nouveau accès. Déplacez le fichier `fic6` dans le répertoire `rep1`. Essayez d'afficher le contenu de `lpfic6` et `lsfic6`. Que constatez-vous ?
9. Redéplacez le fichier `fic6` dans le répertoire `rep4`. Essayez à nouveau d'afficher le contenu de `lpfic6` et `lsfic6`. Supprimez le fichier `fic6` puis recommencez. Que constatez-vous ?

Pour la suite du TP, recréez le fichier `fic6` dans le répertoire `rep4`.

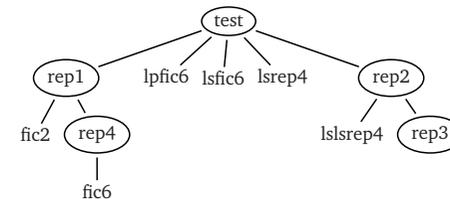
Exercice 3 – Créons des liens sur des répertoires

1. Observons maintenant plus attentivement le répertoire `rep2`. Combien y a-t-il de liens sur ce répertoire ? A quoi correspondent ils ?
2. Dans le répertoire `rep2`, créez un sous-répertoire `rep3`. Combien il y a-t-il maintenant de liens sur le répertoire `rep2` ? Expliquez.
3. Créez un lien physique `lprep4` du répertoire `rep4` dans le répertoire `test`. Que remarquez-vous ?

Avec les implémentations existant actuellement, seul le super-utilisateur peut créer un lien matériel sur un répertoire, et encore, ce n'est pas toujours possible. Par contre, on peut tout à fait créer un lien symbolique sur un répertoire.

4. Créez un lien symbolique `lsrep4` du répertoire `rep4` dans le répertoire `test`.
5. Créez un lien symbolique `lslsrep4` du lien `lsrep4` dans le répertoire `rep2`.
6. Donnez trois manières différentes de se déplacer dans le répertoire `rep4` à partir du répertoire `test`.
7. En utilisant successivement ces trois méthodes, déplacez vous dans le répertoire `rep4` puis remonter dans le répertoire parent à l'aide de la commande `cd ..`. Que remarquez-vous ?
8. Que se passe-t-il si on utilise la commande `ls -Ra1` sur `lsrep4` ? et sur `lslsrep4` ?

A la fin de cet exercice, nous avons maintenant l'arborescence suivante :



I-nœuds

Un système UNIX n'identifie pas un fichier par son nom. En effet, ce n'est pas commode à manipuler et comme on l'a vu avec les liens deux fichiers de noms différents à priori peuvent correspondre au même "bloc mémoire". Dans un système UNIX, un fichier quel que soit son type est en fait identifié par un numéro appelé numéro d'i-nœud ("inode" en anglais). Le lien entre le numéro d'i-nœud attribué par le système et le nom attribué par l'utilisateur se situe en réalité dans le contenu du répertoire dans lequel "se trouve" le fichier.

Exercice 4 – I-nœuds

1. A l'aide de la commande `ls` munie de l'option appropriée, observez le numéro d'i-nœud du fichier `fic2`. Copiez le fichier `fic2` dans le répertoire `rep3`. Quel est son numéro d'i-nœud ?
2. Changez le nom de ce dernier fichier, pour l'appeler `fic6`. Le numéro d'i-nœud change-t-il ?
3. Dans le répertoire `rep4`, créez un fichier `fic10`. Dans le répertoire `test`, créez un lien symbolique et un lien physique sur ce nouveau fichier (on les appellera `lsfic10` et `lpfic10`).
4. Affichez le numéro d'i-nœud du fichier `lsfic10` en listant tout le fichier `test` puis en listant uniquement le fichier `lsfic10`. Que remarquez-vous ?
5. Comparez les numéros d'i-nœud entre le fichier `fic10` du repertoire `rep4`, `lpfic10` et `lsfic10`. Que remarquez-vous ? Expliquez maintenant plus clairement les dernières questions de l'exercice 2.
6. Observez maintenant le numéro d'i-nœud de la racine et de votre répertoire personnel. Que remarquez-vous ?