

Systeme d'exploitation

TP n° 1

Unix et script shell

Florent Devin

1 Unix

1.1 Liens

- Créer un fichier vide nommé `fic1`.
- Créer un lien symbolique nommé `fic2` sur ce fichier.
- Créer un lien 'hard' nommé `fic3` pointant sur `fic1`.
- Vérifier avec la commande `ls -l` le nombre de liens sur chaque fichier.
- Éditer `fic2` pour ajouter du texte.
- Vérifier le contenu de `fic1` et `fic3`.
- Supprimer `fic1`.
- Que se passe t'il pour `fic2` et `fic3`?

1.2 Variables d'environnement

- Affecter à la variable `a` la valeur 123.
- Afficher la variable.
- Faire la même chose dans un shell fils¹.
- Retourner dans le shell père et modifier la portée de la variable `a`.
- Afficher la variable dans un shell fils.

2 Script shell

2.1 Argument d'une commande

Dans cet exercice, on utilise dans les scripts les arguments passés par l'utilisateur. On en profite pour revoir les commandes `shift`, `tail` et `head`.

1. Écrire un script qui affiche un paramètre sur deux.

¹Pour créer un shell fils il suffit de taper la commande : `"xterm &"`

2. Écrire un script qui trie un fichier par ordre croissant du second champ, sachant que les deux lignes de titre doivent figurer dans la sortie triée.

Exemple :

```
Fichier original :      | Fichier trié :
Nom Note                | Nom Note
-----                | -----
Pierre 14               | Jean 10
Paul 20                 | Pierre 14
Cécile 16               | Cécile 16
Jean 10                 | Paul 20
```

3. Modifier le script pour qu'il fasse précéder les noms des étudiants d'un numéro d'ordre

```
Nom Note
-----
1 Jean 10
2 Pierre 14
3 Cécile 16
4 Paul 20
```

2.2 find

- Écrire un script qui recherche un fichier à partir du répertoire courant et envoie le résultat dans un autre fichier. Le fichier à rechercher ainsi que le fichier de résultat devront être demandés à l'utilisateur. Pensez à bien vérifier les arguments.

Exemple :

```
recherche
Quel fichier voulez-vous chercher ?
file
Quel sera le fichier résultat ?
res
Recherche terminée.
```

Pour demander à l'utilisateur une valeur et la stocker dans une variable, il faut utiliser la commande `read`. Cette commande prend en argument le nom de la variable. Donc pour demander un nom de fichier à l'utilisateur, que l'on stockera dans une variable appelée `toto`, on utilisera la suite d'instructions suivante :

```
echo "Quel_est_le_fichier_que_vous_rechercher_?"
read toto
```