Travaux Pratiques d'Unix n°8

Cours d'Informatique de Deuxième Année

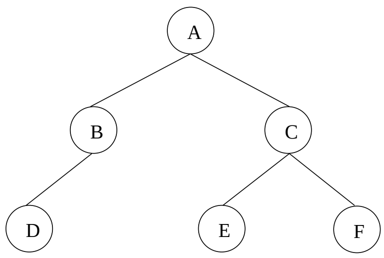
—DEUG Sciences S4-MI—

Commandes de base, Mécanisme de substitution

Lors de cette séance, nous commencerons par décrire quelques commandes de base du système Unix. Nous étudierons également le fonctionnement de l'interpréteur de commande (shell). Nous terminerons par l'examen du système de fichiers utilisé par le système.

▸ Exercice 1. *(Commandes de base)*

1. *À partir de votre répertoire "maison", créer l'arborescence des répertoires ci-dessous, et revenir dans le répertoire initial.*



2. *Créer le fichier vide* `core` *dans le répertoire* `E`*.*

3. *On est toujours dans le répertoire* `E`*. Copier le fichier* `core` *dans tous les autres répertoires de l'arborescence précédemment créée ;*

4. *Déplacer toute l'arborescence* `C`*,* `E`*,* `F` *sous l'arborescence* `D`*.*

▸ Exercice 2. *(Mécanisme de substitution)*

1. *Quel est le résultat de l'exécution de la commande* `ls *.c` *? Détailler l'exécution de cette commande.*

2. *Exécuter la commande* `*`*, et expliquer le résultat.*

3. *Afficher tous les fichiers cachés du répertoire courant.*

4. *Afficher tous les fichiers du répertoire courant*

   dont le suffixe est `.c`*;*

   dont le nom est constitué de six caractères;

   dont le nom est constitué de au moins six caractères;

   dont le nom commence par une majuscule;

   dont le nom ne commence par tp;

   dont le nom contient un chiffre;

   dont le nom contient trois caractères dont un chiffre;

   dont le suffixe est `.h` *ou* `.c` *(donner deux solutions);*

   dont le nom contient `*`*.*

   dont le nom contient deux chiffres entre crochets;

   dont le nom ne contient pas un point d'exclamation;

5. *Créer le répertoire "*`sauvegarde`*" et y copier tous les fichiers dont le suffixe est* `.c`*;*

6. *Changer l'extension de tous les fichiers du répertoire* `sauvegarde` *de* `.c` *en* `.h`*;*

▸ **Exercice 3.** ***(Permissions d'accès)***

1. *Faire en sorte que votre compte ne soit accessible qu'à vous en changeant les droits d'accès de votre répertoire "maison".*

2. *Faire en sorte que le fichier* `core` *ne soit plus accessible en écriture, puis le rendre inaccessible dans tous les cas de figure.*

3. *Créer sous votre répertoire principal un répertoire* `pub` *accessible à tout le monde et y déposer un fichier que vous désirer partager avec tout le monde.*

4. *Vérifier que votre voisin(e) peut le recupérer.*

5. *Faire le ménage sur votre compte en détruisant tous les répertoires et fichiers crées durant ce TP. En profiter pour ranger votre compte en vérifiant les permissions d'accès.*

▸ **Exercice 4.**

1. *Afficher les répertoires, sous-répertoires, et les fichiers classés par ordre de création.*

2. *Afficher tous les fichiers de votre répertoire "maison", exceptés les fichiers représentant les répertoires courant et père.*

3. *Afficher les mêmes fichiers que dans la question précédente, avec l'option* `C`*. Que remarquez-vous ?*

4. *Lister les fichiers contenant un "\".*

5. *Afficher uniquement des informations sur le répertoire dans lequel vous êtes.*

6. *Créer une commande* `ll` *qui affiche tous les répertoires avec "*`/`*" à la fin. Puis, modifier cette commande pour qu'elle affiche en plus tous les fichiers exécutables avec "*`*`*".*

7. *Créer un répertoire* `interdit`*, copier des fichiers dedans, et render-le inaccessible aux autres.*

8. *Comment fait-on pour autoriser l'accès du fichier* `toto` *au groupe, et en même temps en interdire l'accès aux autres ?*

9. *On peut interdire l'accès à un fichier (écriture ou lecture) pendant qu'un programme y accède. Comment fait-on ?*

10. *Est-il possible qu'un fichier ait la caractéristique précédente et en même temps être autorisé en accès à tout le groupe ?*