

Travaux Dirigés Système n°2

Cours d'Informatique de Deuxième Année

— Licence 2ème Année, Deuxième Semestre MIAS —

Shell

Tout ce TD sera **obligatoirement** effectué en mode console. Pour passer d'un mode à l'autre vous utiliserez les combinaisons de touches suivantes :

- Mode console : *CTRL+ALT+F1*.
 - Mode graphique : *CTRL+ALT+F7*.
-

► Exercice 1. (Taille d'un répertoire)

Ecrivez un script shell *size-rep.sh* qui permet de connaître la taille, i.e. la somme de la taille de tous ses fichiers, d'un répertoire passé en paramètre.

Maintenant, faites une autre version incluant la taille de tous les sous-répertoires dans ce calcul.

Remarque : La commande *cut* ne permet pas de faire ce que vous voulez, c'est à dire sélectionner la colonne numéro 5. Vous allez devoir vous servir de la commande *sed* pour éditer le flux de caractère et rendre possible l'utilisation de la commande *cut* à cette fin.

► Exercice 2. (QCM)

Ecrivez un script permettant de répondre à un formulaire par des choix multiples.

Malheureusement, de nombreux utilisateurs vont faire des fautes de frappes... Pour éviter de devoir réaliser de nombreux tests bien souvent illisibles, utilisez la commande *select*.

► Exercice 3. (Un petit jeu)

Ecrivez un script qui permet de dire à l'utilisateur s'il est proche (*brulant*), éloigné (*gelé*), etc. d'un nombre mystère. Vous pouvez mettre en place un système de *cheat code*, i.e. quand le code est saisi l'utilisateur gagne automatiquement !

► **Exercice 4. (La commande find)**

Utilisez la commande `find` pour retrouver tous les fichiers de nom `a.out`, tous les fichiers dont le nom se termine par `~` et tous ceux dont le nom commence par `#`.

Ecrivez un script `clean.sh` qui supprime tous les fichiers trouvés par la commande `find`.

Attention : on n'utilisera pas le symbole `|` pour effectuer cette action, mais uniquement les paramètres de la commande `find`.

► **Exercice 5. (Variables d'environnements)**

1. Affichez toutes les variables d'environnements.
2. Affichez toutes les variables d'environnements dont les valeurs associées contiennent le mot `'fr'`
3. Ecrivez un script qui demande à l'utilisateur sa nationalité (on se limitera à anglais et français dans les résultats) et qui affiche le résultat entré par l'utilisateur.
4. Modifier les variables d'environnements obtenues précédemment en fonction du résultat entré par l'utilisateur.

► **Exercice 6. (Bonus : Le bashrc)**

A la racine de votre compte, ouvrez le fichier `.bashrc`, puis ajoutez y des raccourcis qui permettent :

1. L'exécution de `size-rep.sh`.
2. L'exécution de `clean.sh`.

Cette exécution doit être possible quelque soit votre répertoire courant, en tapant simplement les commandes `size-rep` et `clean`.

Créer un répertoire `bin` dans votre répertoire personnel ou vous rangerez tous les scripts que vous voulez pouvoir utiliser à tout moment. Modifier la variable `PATH` de votre `.bashrc` afin de le faire.

► **Exercice 7. (Session X)**

A partir du mode console, ouvrez une nouvelle session `X`. Pour se faire tapez la commande :

`X : 2&`

Vous venez d'ouvrir une nouvelle session graphique. Faites `Ctrl - Alt - F1` pour retourner en mode console.

Lancez les commandes

xeyes – *display* : 2& et *xlogo* – *display* : 1&

Quelles différences observez vous en faisant *Ctrl – Alt – F7* et *Ctrl – Alt – F8* ?

Lancez un terminal (*xterm*) et un gestionnaire de fenêtre (*icewm*) dans la nouvelle session graphique.

► **Exercice 8. kill_by_name.sh**

A l'aide de la commande *ps* qui affiche la liste des processus, récupérez à l'aide de la commande *grep* et *cut* l'identifiant de tous les processus vous appartenant et dont le nom est passé en argument et terminez les à l'aide de la commande *kill*.

► **Exercice 9. Affichez votre adresse ip**

A l'aide des commande *ifconfig*, *grep* et *cut*, récupérer et afficher l'adresse IP de votre ordinateur.

Où se trouve la commande *ifconfig* ?

Indice : Servez vous de la commande *locate* pour la trouver.

Modifier la variable *PATH* de votre shell afin de pouvoir utiliser cette commande sans en donner le chemin complet. Repercutez cette modification dans votre *.bashrc* pour qu'elle soit permanente. A present faite un script shell qui affiche votre adresse IP.

Bonus : Faites la même chose mais en vous servant de *sed* au lieu de *cut* pour isoler l'adresse IP dans votre ligne.

► **Exercice 10. xkill**

En mode graphique, lancer depuis un terminal le programme *xeyes*, en mettant un "&" à la fin de votre commande afin d'avoir toujours accès au terminal. Lancez à présent le programmer *xkill*. Un curseur apparaît, cliquez sur la fenêtre de *xeyes*. Que se passe il ?

On vous demande maintenant de reimplémenter le programme *xkill* en vous servant uniquement des programmes *xwininfo*, *grep*, *cut* et *xkill* mais en utilisant obligatoirement l'option *-id* de celui ci.

► **Exercice 11. Afficher les fichiers par ordre croissant de taille**

A l'aide uniquement de la commande *ls* affichez les fichiers du repertoire courant triés

par ordre de taille croissant.

Créer un alias dans votre `.bashrc` pour réaliser cette opération.

A présent faite la même chose en vous servant de `ls -l` (sans autre options) et du programme `sort`.

► **Exercice 12. Réalisez un serveur web en une ligne de shell**

En mode graphique ouvrez deux terminaux. Dans le premier à l'aide de la commande `netcat` lancer un serveur qui écoute sur le port 3333. Dans le second terminal connectez vous sur le port 3333 de votre ordinateur (`localhost`) à l'aide de la commande `telnet`. Que se passe t'il si vous écrivez ensuite dans un terminal ? Et dans l'autre ?

Remarque : Taper `^C` (Control+C) pour terminer le serveur.

A présent redirigez la sortie de la commande `who` dans l'entrée de votre serveur `netcat` à l'aide d'un pipe (le caractère `|`), que se passe t'il quand vous vous connectez à votre serveur à l'aide de la commande `telnet` ?

► **Exercice 13. La commande sed et les couleurs ANSI**

Taper dans un terminal la commande `echo "\E[33mFIAT LUX\E[0m"`. Que se passe il ?

Taper à présent la commande `echo -e "\E[33mFIAT LUX\E[0m"`. Comment expliquer la différence ?

Réaliser à présent une commande `sed` qui lit sur son entrée standard un code source C et met en vert (faites varier les nombre 33 pour trouver le code correspondant à cette couleur) toutes les occurrences du mot clefs "int".

Remarque : Le code qui marchait pour `echo` ne marche pas pour `sed`, au lieu de :

`E` il faut rentrer `^V < Esc >`

C'est à dire Control-v suivit de la touche Echap.

► **Exercice 14. Interaction avec internet**

A l'aide des commandes `wget`, `grep` (indice, utilisez les options A, B ou C), et d'autres qui vous semblent appropriées, afficher le numéro de la dernière version stable du noyau linux à partir du site `kernel.org`.

► **Exercice 15. Compression de projets**

Cette commande vous servira pour vos projets. On vous demande d'écrire une commande qui à l'aide de `find` et de `zip` (et éventuellement `grep`) n'archive, dans le fichier `.zip`, que les fichiers strictement nécessaires à la compilation, au test et à la notation de votre projet.

A présent faire la même chose avec la commande `tar` et `gzip`.

► **Exercice 16.** Placez vous dans le répertoire `/usr/include`, à l'aide de la commande `grep` retrouver la taille maximale d'un entier sur votre système.

Comment accéder à cette information depuis un programme C ?

► **Exercice 17.** Placez vous dans le répertoire `/usr/include`, à l'aide de la commande `grep` retrouver la taille maximale d'un entier sur votre système.

Comment accéder à cette information depuis un programme C ?

► **Exercice 18.** Utiliser la commande `find` pour trouver tous les fichiers `.c` sur les comptes des étudiants de votre promotion.

Rediriger la sortie d'erreur vers `/dev/null` afin de ne plus avoir de message d'erreur lorsque vous ne pouvez pas lire dans un répertoire.

Rediriger la sortie standard pour stocker le résultat dans un fichier.

*Ecrivez un script qui affiche la proportion de programme C dans votre compte par rapport à votre promotion. **liste.txt***

► **Exercice 19.** Que fait la commande `yes` ?

Utilisez la commande `yes` en utilisant un fichier en entrée.

Redirigez la sortie vers la carte son.