

Université de Nice Sophia-Antipolis
Licence de *Mathématiques Appliquées et Sciences Sociales*, deuxième année

Enseignement de *Linux, vecteur de culture informatique*

Pierre Crescenzo (Pierre.Crescenzo@unice.fr),
Jean-Pierre Lips (Jean-Pierre.Lips@unice.fr) et
Isabelle Mirbel (Isabelle.Mirbel@unice.fr)

Travaux Dirigés 8 de la semaine du 23 novembre 2009 : Recherches dans des répertoires et premiers scripts Shell

1. Introduction

- a. Connectez-vous. Créez votre répertoire de séance `~/Linux-TD/TD08`. Rendez-vous dans ce répertoire.
- b. Récupérez sur le Web les fournitures de ce TD et placez le fichier dans votre répertoire de séance. Désarchivez-le puis effacez l'archive.
- c. Durant tout ce TD, appliquez-vous à utiliser la complétion, l'historique des commandes et les couper/copier/coller au clavier et à la souris.

2. Recherches dans des répertoires

- a. Avec la commande `find`, répondez aux questions suivantes qui concernent, récursivement, toutes les fournitures du répertoire `rep-find` présent dans votre répertoire de séance.
- b. Listez tous les fichiers dont le nom a un suffixe.
- c. Listez tous les répertoires, et eux seuls.
- d. Listez tous les liens symboliques, et eux seuls.
- e. Affichez le nombre total de tous les fichiers.
- f. Listez tous les fichiers ordinaires dont le nom est `eval` ou `truc` ou `reset`.
- g. Listez tous les fichiers ordinaires dont la date de dernière modification est plus récente que celle du fichier `rep-find/LAST`.
- h. Listez tous les fichiers ordinaires dont la taille est strictement inférieure à 58 caractères et dont le nom commence par une voyelle. Affichez aussi leur taille en nombre de caractères.
- i. Listez tous les fichiers dont le nom contient au moins un chiffre, mais pas de suffixe.
- j. Copiez dans le répertoire `tmp` tous les fichiers ordinaires qui ont un suffixe.
- k. Donnez le nombre de lignes de chaque fichier ordinaire.

3. Premiers scripts Shell

- a. À l'aide de l'éditeur `emacs` puis de la commande `chmod`, créez un script Shell appelé `affiche-params1`. Celui-ci affiche sur une unique ligne le nom du script puis le premier paramètre positionnel.
- b. Testez ce script `affiche-params1` comme suit et expliquez les résultats obtenus par les commandes suivantes :
 1. `affiche-params1`
 2. `affiche-params1 +23`

3. `affiche-params1 toto`
 4. `affiche-params1 ~crescenz`
 5. `affiche-params1 $DISPLAY`
 6. `affiche-params1 '$DISPLAY'`
 7. `affiche-params1 --help +23`
 8. `affiche-params1 --help+23`
 9. `affiche-params1 bonjour tout le monde`
 10. `affiche-params1 'bonjour tout le monde'`
- c. Écrivez un script `affiche-params2` qui affiche sur une ligne son nom, le nombre de paramètres positionnels et les deux premiers de ses paramètres puis, sur une deuxième ligne, la totalité des paramètres positionnels. Testez ce script avec divers appels, en vous inspirant des tests de `affiche-params1`.
- d. Écrivez un script `affiche-params3` qui affiche pour chaque paramètre positionnel, sur une ligne : le caractère '`<`' puis la valeur du paramètre puis '`>`'.
- e. Dans un terminal, et non dans un script Shell, en utilisant une boucle `for`, affichez chaque nom de fichier du répertoire `/bin` sur une ligne entre les caractères '`<`' et '`>`'.
- f. En utilisant une boucle `for` et une expression arithmétique, écrivez un script `affiche-params4` qui affiche pour chaque paramètre positionnel, son numéro d'indice (1, 2, 3, etc.) et sa valeur.