

MANUEL D'EXERCICES PROGRAMMATION SHELL

EXERCICE DU CHAPITRE 2

DÉCOUVERTE DE L'ENVIRONNEMENT

Connectez-vous sur votre machine en suivant les indications de votre formateur.

- Quel shell utilisez-vous ?

- _____
- _____

- Quels sont les shells disponibles sur la machine ?

1. _____
2. _____
3. _____
4. _____
5. _____

- La commande **chsh** permet de changer le shell de connexion de l'utilisateur. Vérifiez que le **ksh** est installé sur votre machine, et demandez à votre formateur si il n'est pas présent. Quand il est installé, changez votre shell de connexion par le **ksh**. Regardez le fichier **/etc/passwd** et vérifiez que votre utilisateur a bien le **ksh** en shell par défaut. Déconnectez-vous et reconnectez-vous. Êtes-vous avec le **ksh** ?
- Remettez en place le shell d'origine.

MANIPULATIONS SUR LE SHELL

- Affichez les alias existants sur votre machine.
- Créez les alias suivant:
 - **ll** pour **ls -l**
 - **ctmp** pour **cd /tmp**
 - **la** pour **ls -la**

EXERCICE DU CHAPITRE 3

SHELLS D'EXÉCUTION

Écrivez le script suivant (`exo3.sh`):

```
cd
echo -n 'répertoire courant: '
pwd
echo -n 'liste des processus: '
ps

cd /tmp
echo -n 'Numéro PID du shell courant: '
echo $$
echo -n 'répertoire courant: '
pwd
```

- Exécutez les commandes à la main, et notez ce que vous obtenez:

- Exécutez le script de 3 façons. Qu'obtenez vous ?

- Notez le PID du shell courant dans les 4 cas:

2.

3.

4.

5.

- Est-il différent de votre shell courant ? Pourquoi ?

PATH

- Quel est le fichier exécuté par la commande `find ~ -print` ?

- _____

- Quels sont les répertoires parcourus lors de la recherche ?

- _____

- _____

- _____

- Quels répertoires restaient à parcourir ?

- _____

- _____

*NB: On affiche le contenu d'une variable avec **echo \$VARIABLE**.*

MANIPULATIONS

- À l'aide de `echo` affichez

- `Aujourd'hui, cours de programmation shell`

- Affichez tous les fichiers commençant par `u` du répertoire `/usr/bin`

- Placez vous dans votre répertoire de connexion. Quel est le résultat de la commande

- `echo zzzz*`

- Pourquoi ?

- _____

- _____

- _____

- Enregistrez la liste des fichiers dont le nom se termine par sh et qui se trouvent dans `/bin` dans le fichier `~/shells`.

- À l'aide de la commande `touch`, créez les fichiers suivants un par un:

- `*`

- _____

- un fichier

- _____

- `[]`

- _____

- `-f`

- _____

- Supprimez les un par un:

- _____

- _____

- _____

- _____

EXERCICE DU CHAPITRE 4

INITIALISATION DU SHELL

- Modifiez votre prompt pour qu'il vous présente

- Le nom du shell,
- Le répertoire courant,
- Le nom de l'utilisateur.

- _____
- _____
- _____
- _____

- Ouvrez un autre shell. Vos modifications sont-elles visibles ? Pourquoi ?

- _____
- _____
- _____
- _____

- Modifiez votre configuration pour que l'ordinateur vous souhaite la bienvenue en cas de shell de connexion. Arrangez vous pour que vos modifications du prompt soient visibles à chaque connexion.

- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____

- Configurez le `ksh` pour qu'il soit toujours en mode emacs, et que son fichier d'initialisation soit le fichier `~/ .kshrc`.

EXERCICE DU CHAPITRE 5

SCRIPTS

- Faites un script ksh qui nous affiche le nombre de paramètres passés, et qui nous les énumère un par un.
- Nous allons créer une calculatrice. Faites un script qui:
 - Appelé sans paramètre, nous demande:
 - un nombre entier,
 - un opérateur (+,-,*,/),
 - un nombre entier,
 - et nous retourne l'opération décrite.
 - Appelé avec 3 paramètres effectue l'opération décrite par les paramètres.
 - Appelé avec un mauvais nombre de paramètres, nous affiche un message d'erreur explicite.
- Faites un script qui nous affiche la liste des 3 répertoires de `/var` qui utilisent le plus d'espace disque. (On pourra utiliser la commande `du -sk rep` pour connaître l'espace disque utilisé par le répertoire `rep`. Ce script est à utiliser en tant que `root`).
- Faites un script qui affiche la liste des utilisateurs définis, avec leurs shells de connexion respectifs.

