Travaux Dirigés d'Unix n°10

Cours d'Informatique de Deuxième Année

—DEUG Sciences S4-MI—

Expressions régulières et scripts shell

Le but de ce TD est double :

- utiliser des commandes permettant de sélectionner une partie du contenu d'un fichier à l'aide des expressions régulières;
- présenter les scripts shell et les variables.

▸ **Exercice 1.** ***(Fonctions* `grep` *et* `cut`*)*** *Considérons le fichier* `/etc/passwd` *qui contient la liste de tous les utilisateurs sous le format suivant :*

```
albert:js5('ge:1111:20:Al Albert:/home/albert:/bin/sh
robert:6f4Dfye:1112:20:Bob Robert:/home/bob:/bin/bash
bob:!Hgt65'f:1100:21:Robert Bob:/homebis/morane:/usr/local/bin/sh
willy:jn5c"Ddy:1101:21:Bill William:/homebis/billy:/bin/sh
ribobine:7jytg65g:1029:20:Bert Ribobinette:/home/ribob:/usr/local/bin/tcsh
...
```

Donner les commandes en bourne shell permettant d'afficher :

- *la liste des noms des utilisateurs pouvant se connecter à la machine;*
- *la liste des utilisateurs dont le nom commence par la lettre* `r`*;*
- *la liste des prénoms des utilisateurs;*
- *le shell par défaut de l'utilisateur* `bob`*.*

▸ **Exercice 2.** ***(Variables en shell)*** *Certains résultats des commandes suivantes ont été effacés. À vous de les retrouver.*

```
echo $peintre
...
test -z "$peintre"
echo $?
0
peintre=Picasso
[ -z $peintre ]
echo $?
```

```
...
test "$peintre" = Raffael
echo $?
...
test $peintre != Rembrandt
echo $?
...
test -n $peintre -a -z $peintre
echo $?
...
longueur=5
echo $longueur
...
test $longueur -lt 10
echo $?
...
read largueur
...
echo $largueur
...
echo $?
...
```

▸ **Exercice 3.** ***(Scripts shell)*** *Les structures de contrôle sont tout simplement les boucles, les conditionnelles. Nous donnons ci-dessous la syntaxe de ces structures :*

```
for <var> in <liste>;
do
   <commande>;
   ...
done
```

```
if <test>;
then
     <commande>;
[else
     <commande>;]
fi
```

```
while <test>;
do
   <commande>;
   ...
done
```

- *Écrire un script en Bourne shell qui affiche le nombre de paramètres passés en argument.*
- *Écrire un script en Bourne shell qui affiche un message d'erreur si le nombre de paramètres est inférieur strictement à 1.*
- *Écrire un script en Bourne shell qui affiche ses paramètres suivis de leur numéro. Exemple :*

  ```
  > sc1 il fait beau
  il 1
  fait 2
  beau 3
  ```

- *Écrire un script en Bourne shell qui affiche les paramètres passés à ce script dans l'ordre inverse de leur saisie.*
- *Écrire un script en Bourne shell qui n'accepte qu'un seul paramètre (un entier N) et qui affiche les N premiers entiers ainsi que leur somme.*