
DS
12 janvier 2006 - Durée : 1h30

Aucun document autorisé

Exercice 1 : Question de Cours

1. Quels sont les différences et les avantages entre les langages interprétés et langages compilés ? Citez un langage interprété et un langage compilé.
2. Quelle(s) commandes permet(tent) d'observer l'ensemble des processus du système ? Comment peut-on tuer un processus (vous donnerez un exemple) ?
3. Proposez des commandes permettant d'obtenir de l'aide sur une commande donnée (que l'on appellera XXX). Exemples ?

Exercice 2 : Évaluation

Un utilisateur tape **successivement** les commandes suivantes dans son terminal. Expliquez pour chaque commande ce qu'elle fait et ce qu'observe l'utilisateur.

1. `a=1`
2. `echo echo $a`
3. `echo "echo $a"`
4. `echo 'echo $a'`
5. `echo `echo $a``

Exercice 3 : Lecture de script

Soit "version1.sh" le script :

```
1  #!/bin/bash
2
3  function myste r ( )
4  {
5      for MOT in $*
6      do
7          nb=`echo $MOT | wc -c `
8          let nc=$nb-1
9          echo -n $nc " "
10     done
11 }
12
13 liste="In girum imus nocte et consumimur igni"
14 myste r $liste
15 echo
```

1. Quel résultat produit ce script ? Justifiez précisément votre réponse : expliquez l'exécution de chaque commande et donnez les valeurs successives de chaque variable.
2. Écrire un script `version2.sh` semblable au précédent, mais traitant des données prises dans un fichier dont le nom sera passé en paramètre, et non pas dans une variable. Par exemple si le fichier `fp` contient la ligne
`In girum imus nocte et consumimur igni`
l'exécution de "`version2.sh fp`" fournira le même résultat que "`version1.sh`".

Exercice 4 : Filtres et redirections

Soit le fichier `comptes` qui contient une liste d'étudiants au format suivant :

```
login prenom nom groupe repAccueil shell
```

Pour les besoins de l'exercice on supposera que les noms et prénoms ne contiennent pas d'espaces.

Donnez les commandes pour

1. afficher les lignes 25 à 35 (incluses) du fichier `comptes` (1 point bonus est donné à ceux n'utilisant que les valeurs numériques 25 et 35).
2. afficher l'ensemble des étudiants n'appartenant pas au groupe `etd3`.
3. sauvegarder dans le fichier `monGroupe` le nom et prénom des étudiants de votre promotion précédés de leur numéro d'apparition.
4. afficher le contenu du fichier `compte` en inversant sur chaque ligne le prénom et le nom.

Exercice 5 : Écriture de script

L'idée de cet exercice est d'écrire le script `poubelle`, admettant comme paramètres un ou plusieurs noms de fichiers, et qui déplace les fichiers indiqués dans un répertoire "poubelle" spécifié par la variable `TRASH`.

1. Écrivez le script `poubelle` respectant les spécifications suivantes :
 - Il commence par vérifier si la variable `TRASH` est bien définie. Si celle-ci n'est pas définie, le script affiche un message "`variable TRASH non définie`".
 - Il vérifie si la valeur attribuée à la variable `TRASH` est bien le chemin d'un répertoire (rappelez-vous, option "`-d`"). Dans le cas contraire, le script affiche "`valeur de TRASH incorrecte`" et s'arrête.
 - Pour chaque nom de fichier donné sur la ligne de commande, le script contrôle que ce fichier existe (rappelez-vous, option "`-f`") ? Selon le cas, il avertira l'utilisateur "`fichier toto inexistant`", ou déplacera le fichier dans `TRASH` avec le message "`fichier toto déplacé`".
2. Donnez un exemple d'utilisation avec la définition de la variable `TRASH`.