

## Exercice 1

Cf. correction séparée par D. Masson.

## Exercice 2

- 1)
  - a. Oui. Exécute un premier xterm sur 25 colonnes en arrière-plan, puis un deuxième xterm avec le nombre courant de colonnes (COL non affecté) en bloquant le terminal appelant.
  - b. Oui. Mais exécution plutôt inattendue : liste tous les fichiers du répertoire courant (\*) plus ceux appels {" , " } ' ' (espace). Le ';' terminal ne sert à rien
  - c. Oui. Exécute la commande contenue dans var si définie, sinon ls -ls puis passe la sortie à wc
  - d. Oui. Si var n'est pas définie, lance ls -ls | wc, autrement dit compte le nombre de fichiers du répertoire courant. Si var est définie, et vaut la chaîne « toto », lancera toto | wc.
  - e. Non. Function serait une commande externe (fichier binaire) à cause de 'F', et donc () pose problème
- 2)
  - a. teste si /etc est un répertoire, et si oui, écrit "message" (ls et wc ne servent à rien!)
  - b. même chose que a. avec un paramètre au lieu de /etc
  - c. Ecrit /usr/local/bin/\* tel quel (aucune résolution à cause des guillemets)
  - d. Ecrit tous les fichiers se terminant par ~ par ordre alphabétique inversé

## Exercice 3

1.1. Ecrit la valeur que vaut x (si elle est définie), suivie de « +1 ». N'écrit ni 12, ni 13, ni 12+1, et x, si elle était définie avant l'appel, est inchangée. La ligne a pour effet de créer une variable d'environnement appelée « x », valable seulement pour la commande echo, qu'elle ignorera bien entendu.

1.2. Equivaut à 2 instructions :  
x=12  
echo \$((x+1))

Donc : affecte à 12 à x, puis écrit 13.

1.3. Equivaut à 2 instructions :  
x=y  
echo \$((x\*2))

Erreur, car la chaîne de caractères « y » ne peut être multipliée par 2.

1.4. Teste si /etc est un répertoire. Si oui, liste ce qu'il contient, puis pipe vers word count, autrement dit si /etc existe, affiche le nombre de fichiers qu'il contient.

1.5. Exécute la commande 'ls -l /???' sur la machine chose en tant que user. Les /??? ne sont pas, et ne doivent pas être résolus (d'où les '), sinon ce seraient les noms des répertoires locaux qui seraient passés à ssh.

1.6. Affecte « /usr/lib... » à nom, puis affiche la partie répertoire et la partie non de fichier seule par pattern matching.

2. C'est une fonction récursive, qui teste si la chaîne passée en argument est un palindrome. `${mot#${mot%?}}` et `${mot%${mot#?}}` sont les premier et dernier caractères de la chaîne.

Les deux dernières lignes suppriment un caractère à gauche, puis à droite, avant de rappeler la récursion. Elle se termine lorsque la chaîne est vide.