



La programmation shell

TP 8

Vous avez fait tous les exercices sur Moodle du TP7 et surtout ceci :

- [Exemples de scripts](#)
- [Quelques exercices](#)

Maintenant vous pouvez faire les exercices d'entraînement suivants :

1. que fait le script :

```
for x in un deux trois quatre
do
    echo x= $x
done      ?
```

2. Faites un script (*saluer*) qui vous dit bonjour en affichant votre login («Bonjour, toto») (bien sûr toto est un exemple), quand vous tapez *saluer [votre-login]* mais aussi, quand vous tapez juste *saluer*
3. Tapez le script suivant, puis expliquez les résultats

```
echo 'je suis la commande cataloguée ' $0 ;
echo 'j ai ' $# ' paramètres' ;
echo 'ces paramètres sont: ' $* ;
echo -n le premier paramètre est ;
echo ' ' $1 ;
echo -n et le quatrième est ;
echo ' ' $4 ;
echo 'le processus en cours a le numero ' $$
```

4. Faire un script qui affiche la phrase : Le script [params] a [n] paramètre(s), que voici : [paramètres], en remplaçant les mots entre crochets par leur valeur; par exemple :
\$ *params* bla ble bli blo blu
La commande *params* a 5 paramètre(s), que voici : bla ble bli blo blu
5. Créez une commande qui, lorsqu'elle est appelée, renvoie le nombre d'arguments qui lui ont été fournis, ainsi que le premier de ces arguments.
6. écrire un script *combien* qui vous dit combien de personnes sont loguées sur une machine donnée; par exemple : \$ *combien* nacre
Il y a 5 personne(s) loguées sur nacre en ce moment.
7. On veut remplacer le suffixe .htm d'un ensemble de fichiers en -.html (pour le tester fabriquer plusieurs fichiers)
8. Ecrire un petit script *op* qui calcule la somme de 2 nombres passés en paramètre ainsi que la division et le reste, puis incrémentation du 1^{er} argument

9. Créer un petit script qui affiche les nombres de 1 à 10 à chaque ligne (avec while), puis leur somme (avec for)
10. écrire le programme script -shell **reponse_oui** affichant *OUI* si l'utilisateur a saisi le caractère *o* ou *O*
11. avec la commande date créer un script qui donne le jour est : mar 4, le mois est : nov.
Remarque : suivant la configuration de linux cela peut être en anglais ou en français
12. même script mais le mois étant abrégé (les 3 premières lettres du jour), le script écrit le jour en entier (utiliser case)
13. Ecrire un script qui recherche le fichier passé en argument dans le répertoire courant, il se termine lorsqu'il le trouve, variante : le fichier est lu (n'est pas en argument) si le fichier n'existe pas, on redemande de le saisir.
14. faire un menu qui affiche
 - choix 1-: calcul de ma moyenne
 - choix 2- calcul de la somme
 - choix 3 - le minimum
 - choix 9 – Quitter

Ce menu envoie un message d'erreur si le code est différent de 1,2, 3, 9
15. Réalisez un shell-script qui, en fonction de l'heure courante (retournée par commande **date**, uniquement l'heure) affiche "Bonjour" entre 0h et 12h, "Bon après-midi" de 12h à 17h, et "Bonne soirée" de 17h à 24h
16. Transformer la chaîne suivante en majuscule, idem une chaîne quelconque passée en paramètre avec le script minmax
17. Un ensemble de noms de fichiers sont en majuscules. On veut tout convertir en minuscule
18. Créez un script **rang** qui affiche le rang de son premier argument dans la liste des arguments suivants. Par exemple, **rang** toto tata titi toto tutu, le rang de toto est 3 (utiliser shift)

Compléments

19. On ne s'intéresse ici qu'à des fichiers contenant un mot par ligne. Écrire un script **compte_mots** qui compte le nombre de mots contenant une des lettres «r», «s» ou «t», et parmi eux, ceux qui ont au moins deux telles lettres. On donnera aussi le nombre de mots ne contenant aucune voyelle. Cela donnera par exemple : **compte_mots** titi, bien sûr vous avez fabriqué titi
 Dans le fichier titi, vous avez :
 45 mots contenant «r», «s» ou «t», et parmi eux,
 12 contiennent deux de ces lettres au moins.
 Il y a aussi 10 mots ne contenant aucune voyelle.
20. On veut chercher toutes les occurrences des quatre éléments (terre, air, eau, feu) dans la première partie de *Germinal*.
21. Chercher le mot «mine» dans les chapitres 3, 4 et 5 de la première partie de *Germinal*, et obtenir un fichier dans lequel figure le nombre d'occurrences du mot dans les fichiers, avec le numéro des lignes
22. Créez un script coupe qui prend trois arguments, le premier étant un nom de fichier et les deux autres des entiers *l* et *l'*, et qui affiche les lignes comprises entre *l* et *l'* dans le fichier. Par exemple,