

Les Scripts Shell - Correction

Exercice 1 :

Ecrire un script nommé **concat** qui concatène puis trie deux fichiers file1 et file2 dans un nouveau fichier file3. Les noms des trois fichiers doivent être passés en paramètre. Le contenu des fichiers doit être trié par ordre alphabétique.

```
#!/bin/bash
# les arguments sont récupérés dans $1, $2 et $3
if [ $# -ne 3 ] # if [ $# != 3 ] fonctionne aussi
then
    echo "usage : $0 fichier1 fichier2 fichier_concat"
else
    cat $1 $2 | sort > $3
fi
```

Exercice 2 :

Ecrire un script nommé **check** qui a deux paramètres et affiche OK si un mot qu'il reçoit en 2^{ème} paramètre est dans le fichier nommé en 1^{er} paramètre et KO sinon.

```
#!/bin/bash
if [ $# -ne 2 ]
then
    echo "usage : $0 fichier mot_a_verifier"
else
    if grep $2 $1
    then
        echo OK
    else
        echo KO
    fi
fi
```

Exercice 3 :

Ecrire un script nommé **repert** qui crée un répertoire passé en argument si le nom du répertoire n'est pas déjà un fichier ou un répertoire

```
#!/bin/bash
if [ $# -ne 1 ]
then
    echo "usage : $0 repertoire_a_créer"
else
    if ls | grep $1 >> /dev/null
    then
        echo "le repertoire $1 existe déjà"
    else
        mkdir $1
    fi
fi
```

Exercice 4 :

Etendre le script précédent à un nombre non limité d'arguments

```
#!/bin/bash
if [ $# -lt 1 ]
```

```
then
    echo "usage : $0 rep1 [rep2] [rep3] ..."
else
    for nom_rep in $*
    do
        if ls | grep $nom_rep >> /dev/null
        then
            echo "Le répertoire $nom_rep existe déjà"
        else
            mkdir $nom_rep
        fi
    done
fi
```

Exercice 5 :

Ecrire un script nommé **checkmail** qui a un paramètre. Ce paramètre est une adresse mail. Le script vérifie si cette adresse est valide.

On considère qu'une adresse mail valide a la forme prenom.nom@organisation .type où

Prénom et nom sont des chaînes de 4 à 11 caractères en minuscule

Organisation est une chaîne de minimum 3 caractères et peut contenir des minuscules, des majuscules, des chiffres et le caractère « - ».

Type est une chaîne de 2 ou 3 caractères minuscules.

```
#!/bin/bash
if [ $# -ne 1 ]
then echo "mauvais nombre d'arguments"
else
    if echo $1 | grep -E '^[a-z]{4,11}\.[a-z]{4,11}@[a-zA-Z0-9\-\]{3,}\.[a-z]{2,3}$' > /dev/null
    then
        echo « adresse mail valide »
    else
        echo « adresse mail invalide »
    fi
fi
```

Exercice 6 : Réalisez un programme de gestion d'agenda téléphonique. L'agenda sera le fichier agenda.txt. Chaque enregistrement contient le nom d'une personne et son numéro de téléphone. Les deux champs sont séparés par une virgule. Le programme affiche le menu suivant :

1. Ajouter une nouvelle fiche
2. Rechercher une fiche avec une partie du nom
3. Détruire une fiche
4. Modifier une fiche (on modifie un numéro de téléphone d'un nom donné)
5. Lister l'annuaire
6. Fin

Votre choix ? :

```
#!/bin/bash
fagenda="fagenda.txt"
while true
do
    echo "1- Ajouter un contact"
    echo "2- Rechercher un contact"
    echo "3- Supprimer un contact"
    echo "4- Modifier le numéro d'un contact";
    echo "5- Lister le répertoire";
    echo "6- Fin";
    echo "Votre choix ?"
```

```
read choix
case $choix in
1)    echo "Nom:"
        read nom
        echo "Tel:"
        read tel
        echo "$nom,$stel" >> $fagenda
        ;;
2)    echo "Nom ou partie du nom ?"
        read nom
        if ! grep -i $nom $fagenda
        then
            echo "$nom introuvable"
        fi
        ;;
3)    echo "Nom:"
        read nom
        if grep -w $nom $fagenda >> /dev/null
        then
            echo "Supprimer définitivement $nom ?"
            read confirm
            if [ confirm="O"] || [confirm="o"]
            then
                grep -v $nom $fagenda > agenda.tmp
                rm $fagenda
                mv agenda.tmp $fagenda
            fi
        else
            echo "$nom introuvable"
        fi
        ;;
4)    echo "Nom ?"
        read nom
        if grep -w $nom $fagenda >> /dev/null
        then
            echo "nouveau numéro:"
            read tel
            grep -v $nom $fagenda > agenda.tmp
            rm $fagenda
            mv agenda.tmp $fagenda
            echo "$nom,$stel" >> $fagenda
        else
            echo "$nom introuvable"
        fi
        ;;
5)    sort $fagenda | more
        ;;
6)    exit
        ;;
*)    echo "Choix inconnu"
        ;;
esac
```

done