Cahier d’exercices

If …

**#!/bin/sh**

**echo -n "Voulez-vous voir la liste des fichiers O/N : "**

**read ouinon**

**if [ "$ouinon" = "o" ] || [ "$ouinon" = "O" ]; then**

**echo "Liste des fichiers :"**

**ls -la**

**elif [ "$ouinon" = "n" ] || [ "$ouinon" = "N" ]; then**

**echo "Ok, Au revoir ! "**

**else**

**echo "Il faut taper O ou N!! Pas $ouinon"**

**fi**

**---**

**#!/bin/bash**

**echo "Entrez votre note :"**

**read -r note**

**if [ "$note" -ge 16 ]; then**

**echo "très bien"**

**elif [ "$note" -ge 14 ]; then**

**echo "bien"**

**elif [ "$note" -ge 12 ]; then**

**echo "assez bien"**

**elif [ "$note" -ge 10 ]; then**

**echo "moyen"**

**else**

**echo "insuffisant"**

**fi**

**---**

**#!/bin/bash**

**read -p "Entrez un répertoire : " repertoire**

**if [ -d $repertoire ]**

**then**

**echo "Bien, vous avez compris ce qui a été demandé !"**

**else**

**echo "Vous n'avez pas compris ce qui a été demandé..."**

**fi**

**---**

**#!/bin/bash**

**read -p "Si vous etes d'accord entrez o ou oui : " reponse**

**if [ ! "$reponse" = "o" -a ! "$reponse" = "oui" ]; then**

**echo "Non, je ne suis pas d'accord !"**

**else**

**echo "Oui, je suis d'accord"**

**fi**

**---**

**if grep "^toto" /etc/passwd > /dev/null**

**then**

**echo "Toto a déjà un compte"**

**fi**

**---**

**#!/bin/bash**

**echo "Entrer votre premier nombre"**

**read imp1**

**echo "Entrer le deuxième nombre"**

**read imp2**

**echo "1. Addition"**

**echo "2. Soustraction"**

**echo "3. Multiplication"**

**echo -n "Choisir un nombre (1,2 ou 3)"**

**read operation**

**if [ $operation -eq 1] ; then**

**echo "Le résultat de l'addition est ... " $(($imp1 + $imp2))**

**else**

**if [ $operation -eq 2] ; then**

**echo "Le résultat de la soustraction est ... " $(($imp1 - $imp2))**

**else**

**if [ $operation -eq 3] ; then**

**echo "Le résultat de la multiplication est ... " $(($imp1 \* $imp2))**

**else**

**echo "Nombre invalide"**

**fi**

**fi**

**---**

**#!/bin/bash**

**echo "Entrer une valeur (chiffre)"**

**read count**

**if [ $count -eq 100 ]**

**then**

**echo "Le compte est 100"**

**elif [ $count -gt 100 ]**

**then**

**echo "Le compte est plus grand que 100"**

**else**

**echo "Le compte est plus petit que 100"**

**fi**

**---**

**#!/bin/bash**

**# Section qui lit un intrant**

**echo "Entrez un code de couleur R ou J ou V"**

**read COULEUR**

**echo $COULEUR**

**#**

**if [ "$COULEUR" == "R" ] ; then**

**echo "Arrêtez... Laisser la voie libre"**

**elif [ "$COULEUR" == "J" ] ; then**

**echo "Soyez prudent... Le feu va passer au rouge"**

**elif [ "$COULEUR" == "V" ] ; then**

**echo "Allez... La voie est libre"**

**else**

**echo "Votre code de couleur est incorrect"**

**fi**

**---**

**#!/bin/bash**

**# Section qui lit un intrant**

**echo "Entrez votre âge"**

**read AGE**

**#**

**if [ "$AGE" -lt 20 ] || [ "$AGE" -gt 60 ] ; then**

**echo "Désolé... Vous n'avez pas l'âge requis"**

**elif [ "$AGE" -gt 20 ] && [ "$AGE" -lt 30 ] ; then**

**echo "Vous êtes dans la vingtaine"**

**elif [ "$AGE" -gt 30 ] && [ "$AGE" -lt 40 ] ; then**

**echo "Vous êtes dans la trentaine"**

**elif [ "$AGE" -gt 40 ] && [ "$AGE" -lt 50 ] ; then**

**echo "Vous êtes dans la quarantaine"**

**elif [ "$AGE" -gt 50 ] && [ "$AGE" -lt 60 ] ; then**

**echo "Vous êtes dans la cinquantaine"**

**fi**

**---**

**#!/bin/bash**

**echo "Entrer le premier nombre"**

**read premier**

**echo "Entrer le deuxième nombre"**

**read second**

**if [ $premier -eq 0 ] && [ $second -eq 0 ]**

**then**

**echo "La première et la seconde valeur sont égales à zéro"**

**elif [ $premier -eq $second ]**

**then**

**echo "Les deux valeurs sont égales"**

**elif [ $premier -gt $second ]**

**then**

**echo "$premier est plus grand que $second"**

**else**

**echo "$premier est plus petit que $second"**

**fi**

**---**

**#!/bin/bash**

**echo "Entrez une adresse IP ..."**

**read IP**

**if [ ! -z $IP ]**

**then**

**ping -c 1 $IP**

**if [ $? -eq 0 ] ; then**

**echo "L'hôte distant répond à la demande de ping"**

**else**

**echo "L'hôte distant ne répond pas à la demande de ping"**

**fi**

**else**

**echo "Aucune adresse IP n'est entrée"**

**fi**

**---**

**#!/bin/bash**

**echo "Entrez votre prénom ..."**

**read prenom**

**echo "Entrez votre nom ..."**

**read nom**

**echo $prenom**

**echo $nom**

**longeur1=${#prenom}**

**longeur2=${#nom}**

**echo "Le nombre de caractères de votre prénom est ... $longeur1"**

**echo "Le nombre de caractères de votre nom est ... $longeur2"**

**---**

**#!/bin/bash**

**# Vérifier si le fichier test1.txt existe**

**if [ -e test1.txt ]; then**

**echo "Le fichier test1.txt existe. Le processus se contimue."**

**else**

**# sinon, affichage de la seconde étape et vérifier la prochaine validation.**

**echo "Le fichier test1.txt n'existe pas. Vérification dans le répertoire temp."**

**# Vérifier si le répertoire temp existe et, si oui, afficher.**

**if [ -d temp ]; then**

**echo "Le répertoire temp existe. Création du fichier test1.txt dans le**

**répertoire temp"**

**#Créer le fichier temporaire test1.txt**

**touch temp/test1.txt; else**

**# Autrement affiher la seconde déclaration et effectuer la prochaine validation**

**echo "Le répertoire n'existe pas. Est-ce un lien symbolique"**

**if [ -s lien-symbolique ]; then**

**echo "Le lien-symbolique existe. Tout es OK."**

**else**

**echo "Le lien-symbolique n'existe pas. Rien ne va plus."**

**exit 2**

**fi**

**fi**

**fi**

**---**

**#!/bin/bash**

**# script4.sh test si $passwdir existe**

**passwdir=/etc/passwdd**

**checkdir() {**

**if [ -e $passwdir ]; then**

**echo "le fichier $passwdir existe"**

**else**

**echo "le fichier $passwdir n'existe pas"**

**fi**

**}**

**checkdir**

**exit**

**---**

**#!/bin/bash**

**# script5.sh structure de base d’un script**

**target=$1**

**usage() {**

**echo "Usage: $0 <fichier/dossier>"**

**exit**

**}**

**main() {**

**ls -l $target**

**echo "nombre de lignes : $(wc -l $target)"**

**stat $target**

**}**

**if [ $# -lt 1 ]; then**

**usage**

**elif [ $# -eq 1 ]; then**

**main**

**else**

**usage**

**fi**

**exit**

**---**

**#!/bin/bash**

**note=0**

**moyenne=0**

**i=0**

**until [ "$note" -lt 0 ]; do**

**echo "Entrez votre note (q pour quitter) :"**

**read -r note**

**if [ "$note" = "q" ]; then**

**note=-1**

**echo "au revoir !"**

**elif [ "$note" -ge 16 ]; then**

**echo "très bien"**

**elif [ "$note" -ge 14 ]; then**

**echo "bien"**

**elif [ "$note" -ge 12 ]; then**

**echo "assez bien"**

**elif [ "$note" -ge 10 ]; then**

**echo "moyen"**

**elif [ "$note" -ge 0 ]; then**

**echo "insuffisant"**

**else**

**echo "au revoir !"**

**fi**

**if [ "$note" -ge 0 ]; then**

**let moyenne=$moyenne+$note**

**let i=$i+1**

**fi**

**done**

**if [ "$i" -le 0 ]; then**

**let i=1**

**fi**

**let moyenne=$moyenne/$i**

**echo "La moyenne est de $moyenne ($i notes)"**

**---**

**#!/bin/bash**

**operation () {**

**result=1**

**for (( i=0 ; i<$value ; i++ ))**

**do**

**let result=$result\*$value**

**done**

**echo "$value^$value = $result"**

**}**

**if [ "$#" -eq 0 ]; then**

**echo "Saisir une valeur"**

**read -r value**

**else**

**value=$1**

**fi**

**operation**

**---**

**#!/bin/bash**

**appreciation () {**

**if [ "$note" -ge 16 ]; then**

**echo "très bien"**

**elif [ "$note" -ge 14 ]; then**

**echo "bien"**

**elif [ "$note" -ge 12 ]; then**

**echo "assez bien"**

**elif [ "$note" -ge 10 ]; then**

**echo "moyen"**

**else**

**echo "insuffisant"**

**fi**

**}**

**# programme principal**

**clear**

**if [ "$#" -ne 0 ]; then**

**note=$1**

**else**

**echo "Saisir une note"**

**read -r note**

**fi**

**appreciation**

**---**

**#!/bin/bash**

**read -rp 'Entrez le nom du répertoire : ' repertoire**

**if [ -d "$repertoire" ]; then**

**[[ $repertoire != \*/ ]] && repertoire="${repertoire:+$repertoire/}"**

**shopt -s nullglob #sinon repertoireVide/\*, retournera repertoireVide/\***

**fichiers=( ${repertoire}\* )**

**((${#fichiers[@]})) && {**

**for (( x=${#fichiers[@]}-1; x>=0; x-- )); do**

**echo "${fichiers[x]}"**

**done**

**} || echo "$repertoire est vide"**

**else**

**echo "$repertoire n'est pas un répertoire"**

**fi**

**---**

**#!/bin/bash**

**TestUser () {**

**if grep "^$util:" /etc/passwd > /dev/null**

**then**

**echo $ligne**

**fi**

**}**

**if [ "$#" -eq 0 ]; then**

**echo "Chemin et nom du fichier :"**

**read -r fichier**

**else**

**fichier="$1"**

**fi**

**if [ -e "$fichier" ]; then**

**while read -r ligne; do**

**set -- "$ligne"**

**util="$1"**

**TestUser**

**done < $fichier**

**else**

**echo "Le fichier $fichier n'existe pas."**

**fi**

**---**

**#!/bin/bash**

**menu()**

**{**

**echo "1 - Vérifier l'existence d'un utilisateur"**

**echo "2 - Connaître le UID d'un utilisateur"**

**echo "3 - Quitter"**

**read -r choix**

**}**

**verification()**

**{**

**read -p "le nom de l'utilisateur= " nom**

**grep $nom /etc/passwd > fichtest**

**if [ -s fichtest ] ;then**

**echo "ce utilisateur existe"**

**else**

**echo "ce utilisateur n'existe pas"**

**fi**

**}**

**uid()**

**{**

**read -p "le nom de l'utilisateur= " nom**

**cat /etc/passwd | grep $nom > fichtest**

**if [ -s fichtest ] ;then**

**grep $nom /etc/passwd | cut -d : -f 3**

**else**

**echo "ce utilisateur n'existe pas"**

**fi**

**}**

**repeat=1**

**while [ $repeat -eq 1 ] ; do**

**menu**

**if [ $choix -eq 3 ] ; then**

**echo "au revoir"**

**repeat=2**

**fi**

**if [ $choix -ne 1 ] && [ $choix -ne 2 ] && [ $choix -ne 3 ] ;then**

**echo "erreur de saisie"**

**fi**

**if [ $choix -eq 1 ] ;then**

**verification**

**fi**

**if [ $choix -eq 2 ] ;then**

**uid**

**fi**

**done**

**---**

**if [ $# -eq 0 ]**

**# pas de paramètre, on lit les chaînes sur stdin**

**then echo "chaine 1 ? \c"**

**read ch1**

**echo "chaine 2 ? \c"**

**read ch2**

**echo "chaine 3 ? \c"**

**read ch3**

**elif [ $# -ne 3 ]**

**# nombre de paramètres différent de 3 : erreur.**

**then echo nombre de paramètres incorrect >&2**

**exit 5**

**else**

**# les chaînes sont présentes sous la forme de paramètres**

**# on les affecte aux variables ch1, ch2, ch3**

**ch1="$1"; ch2="$2"; ch3="$3"**

**fi**

**# ici, nous avons trois chaînes : ch1, ch2, ch3**

**# quelle que soit l'obtention.**

**# Sont-elles identiques ?**

**if test "$ch1" = "$ch2" -a "$ch2" = "$ch3"**

**then**

**echo les trois chaînes sont identiques**

**exit 0**

**elif test "$ch1" = "$ch2"**

**then**

**echo les deux premières sont identiques**

**exit 1**

**elif test "$ch1" = "$ch3"**

**then**

**echo la première et la dernière sont identiques**

**exit 2**

**elif test "$ch2" = "$ch3"**

**then**

**echo les deux dernières sont identiques**

**exit 3**

**# elles sont donc toutes différentes**

**else echo les trois chaînes sont différentes**

**exit 4**

**fi**

case

**#!/bin/sh**

**case $USER in**

**root)**

**echo "Mes respects M. $USER" ;;**

**jean | patrick)**

**echo "Salut à $USER !" ;;**

**toto)**

**echo "Ne faites pas le zigo $USER !" ;;**

**\*)**

**echo "Qui êtes vous ?" ;;**

**esac**

**---**

**#!/bin/sh**

**echo "Quel est votre fruit préféré?"**

**read -${BASH\_VERSION+e}r fruit**

**case $fruit in**

**pomme)**

**echo "Mmmmh, j'adore!"**

**;;**

**banane)**

**echo "Hmmm, j'aime beaucoup, non?"**

**;;**

**orange|tangerine)**

**echo "Beurks ! Je n'aime vraiment pas !"**

**exit 1**

**;;**

**\*)**

**echo "Fruit inconnu- Êtes-vous certain que ce ne soit pas toxic ?"**

**esac**

**---**

**#!/bin/sh**

**echo -n "Etes-vous fatigué ? "**

**read on**

**case "$on" in**

**oui | o | O | Oui | OUI ) echo "Allez faire du café !";;**

**non | n | N | Non | NON ) echo "Programmez !";;**

**\* ) echo "Ah bon ?";;**

**esac**

**---**

**#! /bin/sh**

**# Affichage du jour de la semaine en lettres**

**if [ $# -lt 1 ]; then**

**d=`date +%w`**

**phrase="Aujourd'hui c'est"**

**else**

**d=$1**

**phrase="Le jour donnÃ© correspond Ã "**

**fi**

**case $d in**

**0 ) jour=Dimanche ;;**

**1 ) jour=Lundi ;;**

**2 ) jour=Mardi ;;**

**3 ) jour=Mercredi ;;**

**4 ) jour=Jeudi ;;**

**5 ) jour=Vendredi ;;**

**6 ) jour=Samedi ;;**

**7 ) jour=Dimanche ;;**

**esac**

**echo $phrase $jour**

**---**

**#!/bin/bash**

**case $1 in**

**"Chien" | "Chat" | "Souris")**

**echo "C'est un mammifère"**

**;;**

**"Moineau" | "Pigeon")**

**echo "C'est un oiseau"**

**;;**

**\*)**

**echo "Je ne sais pas ce que c'est"**

**;;**

**esac**

Boucles (for, while, until)

**#! /bin/sh**

**question="Réponse (oui/non) ?: "**

**reponse="rien"**

**while [ $reponse != "oui" -a $reponse != "non" ]**

**do**

**echo $question**

**read reponse**

**done**

**---**

**#!/bin/bash**

**if [ "$#" -eq 0 ]; then**

**echo "Saisir une valeur : "**

**read -r val**

**else**

**val=$1**

**fi**

**# Dans le cas où c'est négatif, on rend la valeur positive**

**if [ "$val" -lt 0 ]; then**

**let val=-1\*$val**

**fi**

**result=1**

**val2="$val"**

**while [ "$val" -ne 0 ]; do**

**printf "$val "**

**let result=$result\*$val**

**let val=$val-1**

**if [ "$val" -ne 0 ]; then**

**printf "\* "**

**fi**

**done**

**echo "= $result"**

**---**

**#!/bin/bash**

**j=0**

**k=0**

**l=0**

**if [ "$#" -eq 0 ]; then**

**echo "Saisir le répertoire"**

**read -r rep**

**else**

**rep=$1**

**fi**

**cd $rep**

**for i in \*; do**

**if [ -d "$i" ]; then**

**echo "$i"**

**let j=$j+1**

**fi**

**if [ -f "$i" ]; then**

**echo $i**

**let k=$k+1**

**fi**

**if [ -x "$i" ]; then**

**echo $i**

**let l=$l+1**

**fi**

**done**

**echo "Il y a $j répertoires, $k fichiers et $l exécutables dans $rep"**

**---**

**#!/bin/bash**

**# Si on a au moins un paramètre et que le premier paramètre est un répertoire**

**if [ "$#" -ge 1 ] && [ -d "$1v ]; then**

**cd $1**

**fi**

**nb=0**

**for fichier in \*; do**

**tab[$nb]="$fichier"**

**let nb=$nb+1**

**done**

**# affichage inversé**

**for (( i=$nb ; i>=0 ; i=$i-1 )); do**

**echo ${tab[$i]}**

**done**

**---**

**#! /bin/sh**

**# Affichage du jour de la semaine en lettres**

**homes=/home**

**for rep in `ls $homes`**

**do**

**echo `du -h $homes/$rep/ | grep "$homes/$rep/$" | cut -d / -f 1` $rep**

**done**

**---**

**#!/bin/bash**

**MaxBits=8**

**pow () {**

**value2=1**

**for (( k=1 ; k<$i ; k ++ )); do**

**let value2=$value2\*2**

**done**

**}**

**if [ "$#" -eq 0 ]; then**

**echo "Saisir une valeur décimale"**

**read -r value**

**else**

**value="$1"**

**fi**

**declare -a bin**

**j=0**

**for (( i="$MaxBits" ; i>=0 ; i-- )); do**

**pow**

**if [ "$value" -ge "$value2" ]; then**

**bin[$j]="1"**

**let value=$value-$value2**

**else**

**bin[$j]="0"**

**fi**

**let j=$j+1**

**done**

**printf "La valeur binaire est de : "**

**for (( i=0 ; i<$MaxBits ; i++ )); do**

**printf "${bin[$i]}"**

**done**

**echo ""**

Fonctions

**#!/bin/bash**

**if [ "$#" -lt 3 ]; then**

**echo "Erreur : Il manque des paramètres !"**

**elif [[ "$1" =~ ^[0-9]+$ ]] && [[ "$3" =~ ^[0-9]+$ ]]; then**

**if [[ "$2" =~ ^(\+|\-|\/|\\*){1}$ ]]; then**

**if [ $3 -ne 0 ] || [ "$2" != "/" ]; then**

**echo "Le résultat est : "$(( $1 $2 $3 ))**

**else**

**echo "Erreur : division par 0 !"**

**fi**

**else**

**echo "Erreur : opérateur invalide !"**

**fi**

**else**

**echo "Erreur : opérandes invalides !"**

**fi**

**---**

**#!/bin/bash**

**function pause {**

**echo "Appuyez sur ENTER pour continuer"**

**read**

**}**

**function saisirUser {**

**echo "Saisir l'utilisateur"**

**read -r util**

**}**

**function verifyUser {**

**if grep "^$util:" /etc/passwd > /dev/null; then**

**echo "L'utilisateur existe"**

**else**

**echo "L'utilisateur n'existe pas"**

**fi**

**pause**

**}**

**rep=1**

**while [ "$rep" -eq 1 ]; do**

**clear**

**printf "menu :\n\n"**

**echo "1. Vérifier l'existence d'un utilisateur"**

**echo "2. Connaître l'UID d'un utilisateur"**

**echo -e "3. Quitter\n"**

**read -r choix**

**case "$choix" in**

**1)**

**saisirUser**

**verifyUser ;;**

**2)**

**saisirUser**

**id $util**

**pause ;;**

**q)**

**echo "Au revoir"**

**pause**

**rep=0 ;;**

**\*)**

**echo "Erreur de saisie"**

**pause ;;**

**esac**

**done**

**---**

**#!/bin/bash**

**saisir () {**

**printf "Saisir le premier nombre, puis le signe de l'opération puis le deuxième nombre**

**:\n\n"**

**read -r nb1**

**read -r s**

**read -r nb2**

**}**

**calcul () {**

**case "$s" in**

**"+") let result=$nb1+$nb2 ;;**

**"-") let result=$nb1-$nb2 ;;**

**"\*") let result=$nb1\*$nb2 ;;**

**"/") let result=$nb1/$nb2 ;;**

**\*)**

**let result=0**

**echo -e "Erreur de saisie !\nLe résultat est faux.";;**

**esac**

**}**

**calcul2 () {**

**let result=$nb1$s$nb2**

**}**

**if [ "$#" -eq 3 ]; then**

**nb1=$1 ; s=$2 ; nb2=$3**

**else**

**saisir**

**fi**

**calcul**

**echo "Le résultat est $result"**

**calcul2**

**echo "Calculé d'une autre façon : $result"**

**---**

**#!/bin/bash**

**menu()**

**{**

**echo "1 - Vérifier l'existence d'un utilisateur"**

**echo "2 - Connaître le UID d'un utilisateur"**

**echo "3 - Quitter"**

**read -r choix**

**}**

**verification()**

**{**

**read -p "le nom de l'utilisateur= " nom**

**grep $nom /etc/passwd > fichtest**

**if [ -s fichtest ] ;then**

**echo "ce utilisateur existe"**

**else**

**echo "ce utilisateur n'existe pas"**

**fi**

**}**

**uid()**

**{**

**read -p "le nom de l'utilisateur= " nom**

**cat /etc/passwd | grep $nom > fichtest**

**if [ -s fichtest ] ;then**

**grep $nom /etc/passwd | cut -d : -f 3**

**else**

**echo "ce utilisateur n'existe pas"**

**fi**

**}**

**repeat=1**

**while [ $repeat -eq 1 ]**

**do**

**menu**

**if [ $choix -eq 3 ];then**

**echo "au revoir"**

**repeat=2**

**fi**

**if [ $choix -ne 1 ] && [ $choix -ne 2 ]   
 && [ $choix -ne 3 ] ;then**

**echo "erreur de saisie"**

**fi**

**if [ $choix -eq 1 ] ;then**

**verification**

**fi**

**if [ $choix -eq 2 ] ;then**

**uid**

**fi**

**done**

---