



DIRO  
IFT 1215

## DÉMONSTRATION N° 11

– Correction –

*Max Mignotte*

DIRO, Département d'Informatique et de Recherche Opérationnelle, local 2377.

[http: //www.iro.umontreal.ca/~mignotte/ift1215/](http://www.iro.umontreal.ca/~mignotte/ift1215/)

*E-mail: [mignotte@iro.umontreal.ca](mailto:mignotte@iro.umontreal.ca)*

## --- A. TESTS SUR LES FICHIERS ET UTILISATION DE FOR ---

1. Ecrivez un script qui dit si le paramètre passé est :

- Un fichier
- Un répertoire
- Autre chose

```
-----
#!/bin/bash

if [ -z $1 ]; # Si le paramètre est nul on envoie un message
then
  echo "Vous n'avez pas entré de paramètre"
  echo "Le mode d'utilisation du script est $0 NomDuFichier"

elif [ -f $1 ]; # Si c'est un fichier
then
  echo "$1 est un fichier"

elif [ -d $1 ]; # Si c'est un répertoire
then
  echo "$1 est un répertoire"

else # Et sinon
  echo "Autre chose ou n'existe pas"
fi
-----
```

2. Ecrivez un script qui n'affiche que les répertoires et donne leur nombre

```
-----
#!/bin/bash

j=0
for i in * ;
do
  if [ -d $i ];
  then
    echo $i
    j=$((j + 1))
  fi
done
echo "Il y a $j répertoires dans le répertoire"
-----
```

3. Ecrivez un script qui n'affiche que les fichiers et donne leur nombre

```
-----
#!/bin/bash

j=0
for i in * ;
do
  if [ -f $i ];
  then
    echo $i
    j=$((j + 1))
  fi
done
echo "Il y a $j fichiers dans le répertoire"
-----
```

## --- B. UTILISATION DE CASE ---

En utilisant la structure case, écrire un script qui :

- Affiche un menu
- Demande à l'utilisateur de saisir une option du menu
- Affiche à l'utilisateur l'option qu'il a choisi

Exemple de ce qui doit s'afficher à l'écran :

```
-----
***** Menu général *****
```

```
<1> Comptabilité
<2> Gestion commerciale
<3> Paie
<9> Quitter
```

Votre choix :

```
-----
#!/bin/bash
```

```
clear
echo "***** Menu général *****"
echo
echo "<1> Comptabilité"
echo "<2> Gestion commerciale"
echo "<3> Paie"
echo "<9> Quitter"
echo
echo -n "Votre choix : "

read choix
case $choix in
  1) echo "Vous avez choisi l'option 1 Comptabilité"
    ;;
  2) echo "Vous avez choisi l'option 2 Gestion Commerciale"
    ;;
  3) echo "Vous avez choisi l'option 3 Paie"
    ;;
  9) echo "Vous avez choisi de quitter l'application"
    ;;
  *) echo "Vous avez saisi un peu n'importe quoi"
esac
-----
```

## --- C. UTILISATION DE WHILE ---

En utilisant la structure while, écrire un programme qui donne les valeurs de y d'une fonction pour les valeurs de x allant de -10 à 10 avec un incrément de 1.

1. Réalisez le traitement pour les fonctions  $y=x$  et  $y=x^2$

```
-----
#!/bin/bash

x=-10
while [ $x -lt 11 ]; do
  echo "Pour y=x : y=$x"
  echo "Pour y=x^2 : y=$((x * x))"
  x=$((x + 1))
done
-----
```

2. Adaptez le script afin que les bornes -x, +x soient passées en paramètres au script.

```
-----
#!/bin/bash

x=$1
while [ $x -lt $((2 + 1)) ]; do
  echo "Pour y=x : y=$x"
  echo "Pour y=x^2 : y=$((x * x))"
  x=$((x + 1))
done
-----
```

3. Modifiez le script de façon à ce que l'on puisse passer l'incrément en paramètre.

```
-----
#!/bin/bash

x=$1
while [ $x -lt $((2 + 1)) ]; do
  echo "Pour y=x : y=$x"
  echo "Pour y=x^2 : y=$((x * x))"
  x=$((x + $3))
done
-----
```

## --- D. UTILISATION DE WHILE ET CASE ---

En reprenant le script de la question B, modifiez le de façon à ce que celui-ci boucle tant que l'utilisateur ne choisit pas l'option 9. Faites aussi en sorte que l'utilisateur ne puisse pas terminer le programme en utilisant Ctrl-C.

```
-----
#!/bin/bash

trap '' INT # Ctrl-C sera ignoré

choix=0
while [ $choix -ne 9 ];
do
  clear
  echo "***** Menu général *****"
  echo
  echo "<1> Comptabilité"
  echo "<2> Gestion commerciale"
  echo "<3> Paie"
  echo "<9> Quitter"
  echo
  echo -n "Votre choix : "

  read choix
  case $choix in
    1) echo "Vous avez choisi l'option 1 Comptabilité"
       ;;
    2) echo "Vous avez choisi l'option 2 Gestion Commerciale"
       ;;
    3) echo "Vous avez choisi l'option 3 Paie"
       ;;
    9) echo "Vous avez choisi de quitter l'application"
       ;;
    *) echo "Vous avez saisi un peu n'importe quoi"
       ;;
  esac
  if [ $choix -ne 9 ];
  then
    echo -n "Appuyez [Enter] pour continuer"
    read
  fi
done
-----
```

## --- E. FONCTIONS ET RECCURENCE ---

Ecrivez un script qui, pour un répertoire passé en paramètre, fait la chose suivante : donne la liste des fichiers et explore de façon récurrente les répertoires pour donner la liste des fichiers qu'ils contiennent.

```
-----
#!/bin/bash

function liste_fichiers() {
    local fichiers

    for i in `ls $1`;
    do
        if [ -d $1/$i ];
        then
            liste_fichiers $1/$i
        else
            fichiers="$fichiers $i"
        fi
    done
    echo "Repertoire $1 :"
    for i in $fichiers;
    do
        echo "  $i"
    done
}

liste_fichiers $1
-----
```

## --- F. BASH ET HTML ---

Ecrivez un script qui génère une page Web dont le nom est passé en premier paramètre. Cette page doit avoir comme entête le nom du répertoire passé en deuxième paramètre et doit contenir la liste des fichiers et répertoires se trouvant dans le répertoire cité en paramètre. Chaque ligne de la liste doit contenir le nom du fichier ou répertoire, les permissions, et doit être un lien pointant sur celui-ci.

```
-----
#!/bin/bash

echo "<HTML><BODY>" > $1
echo "<H1> $2 </H1>" >> $1

for i in `ls $2`;
do
    echo -n `ls -ld $2/$i | cut -d' ' -f1` >> $1
    echo " <A HREF='ls -d $2/$i'\`ls -d $2/$i`</A>" >> $1
    echo "<BR>" >> $1
done

echo "</BODY></HTML>" >> $1
-----
```