

**Système**  
**Initiation à Bash - Correction**  
Pierre Pompidor

**Effaçage du terminal :**

```
clear
```

**Affichage avec un "echo" de chaque paramètre du script, puis du premier :**

```
for p in "$@"
do
    echo $p
done
echo $1
```

Il est aussi possible de tout écrire sur une ligne, le ; occultant le passage à la ligne (non recommandé) :

```
for p in "$@"; do echo $p; done; echo $1
```

Il est également possible d'utiliser \$@ ou \$\* (sans les doubles quotes) pour itérer sur les paramètres, mais ceux comprenant des espaces seront mal pris en compte.

**Affichage avec un "ls -l" de chaque fichier régulier du dossier courant :**

```
for f in *
do
    if [ -f "$f" ]
    then
        ls -l "$f"
    fi
done
```

**Création puis affichage d'une variable qui contient le nombre de fois où le fichier trèspoli contient la chaîne "bonjour"**

```
n=$(grep -c bonjour trèspoli)
echo $n
```

**Affichage du nombre de fois où chaque paramètre du script est retrouvé dans un fichier régulier du dossier courant :**

```
clear
for p in "$@"
do
    for f in *
    do
        if [[ -f $f ]] # si le fichier est un fichier texte ou binaire
        then
            n=$(grep -c "$p" "$f")
            if (( $n > 0 ))
            then
                echo $p a été trouvé $n fois dans $f
            fi
        fi
    done
done
```

Utilisez bien les  **doubles parenthèses**  pour tester si la variable  $n$  à une valeur supérieur à 0, et non pas les doubles crochets (dans ce dernier cas les opérandes sont toujours considérées comme étant des chaînes de caractères) !

**Affichage des noms des fichiers du dossier courant qui contiennent le paramètre du script dans leurs noms :**

```
for p in "$@"
do
    for f in *
    do
        if [[ "$f" == *$p* ]]
        then
            echo $f
        fi
    done
done
```

**Déplacement de fichiers dans un dossier correspondant à leur extension :**

```
if [[ ! -d $1 ]]
then
    echo le dossier $1 "n'existe" pas, je le crée
    mkdir $1
fi
for f in *
do
    if [[ -f "$f" ]] && [[ "$f" == *$1 ]]
    then
        mv "$f" $1
    fi
done
```