

Unix : Utilisation et programmation



Pr. Hajar IGUER,
hajar.iguer@uic.ac.ma

Unix[®]
Operating System



Université Internationale
de Casablanca

LAUREATE INTERNATIONAL UNIVERSITIES

BOUCLES



Boucles

- Les structures **while**, **until** et **for** sont des boucles.
- C'est-à-dire qu'elles soumettent à une condition déterminée la répétition (ou non) d'une suite de commandes.



Boucle While

- La boucle **while** exécute les commandes de manière répétée tant que la condition est vraie.
- La première syntaxe:
 - **while [condition]**
 - **do** :
 - **# instructions à répéter**
 - **done**
- La deuxième syntaxe:
 - **while test condition**
 - **do**
 - **# instructions à répéter**
 - **done**



Exemple

```
#!/bin/bash
echo "Tapez le mot de passe :"
read password
while test "$password" !=
"m_d_p_correct"
do
echo "Mot de passe incorrect."
echo "Tapez le mot de passe"
read password
done
echo "Mot de passe correct."
echo "Bienvenue !"
```



Exemple 2

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

```
compteur=0
```

```
while [ $compteur -lt 7 ]
```

```
do
```

```
    echo "Le compteur vaut $compteur"
```

```
    let compteur=$compteur+1
```

```
done
```



Exercice

- Créez une boucle qui répète "vive le Bash !" tant que la valeur d'une variable ne vaut pas 0.

```
#!/bin/bash  
quitter=1  
while [ $quitter -ne 0 ]  
do  
echo "Appuyez sur 0 pour quitter ..."  
read quitter  
echo "vive le Bash !"  
done  
echo "bye !"
```



Boucle Until

- La boucle **Until** exécute les commandes de manière répétée tant que la condition est fausse.
- La syntaxe:
 - **until** [*condition*]
 - **do**
 - *commandes*
 - **done**



Exemple 2

```
#!/bin/bash
compteur=20
until [ $compteur -lt 10 ]
do
    echo $compteur
    let compteur-=1
done
```



Boucle for

- La boucle **for** exécute les commandes autant de fois que le nombre de mots dans la "liste d'arguments".
- Elle affecte successivement à une variable chaque chaîne de caractères trouvée dans une *liste*, et exécute les *commandes* une fois pour chaque chaîne.
- La syntaxe:
 - **for var in liste**
 - **do**
 - **commandes**
 - **done**



Exemple

```
for index in 1 2 3 4 5 6 7 8 9 10
do
    if [ $index -le 3 ]; then
        echo "continue"
        continue
    fi
    echo $index
    if [ $index -ge 8 ]; then
        echo "break"
        break
    fi
done
```



Exercice

- Ecrire un script qui récupère la température pendant une semaine puis affiche la température moyenne.



Correction

```
#!/bin/bash
# Calcul de la température moyenne

for num in 1 2 3 4 5 6 7
do
    read -p "Saisissez la temperature
du jour $num: " Temp
    let TempTotal=TempTotal+Temp
done

let TempMoy=TempTotal/7
echo "Average temperature: " $TempMoy
```



Break & Continue

- Interrompre les boucles **for, while or until:**
- The break statement
 - Exécute les instructions après the done statement
 - Termine l'exécution de la boucle
- The continue statement
 - Ignore les instructions de test pour l'itération en cours
 - Continue l'exécution de la boucle



Exercice

- Ecrire un script qui tant que l'utilisateur ne saisi pas le mdp « ubuntu » et que le nombre d'essai est inférieur à 4, on va lui redemandé de le resaisir

