

TP 2 – Bash

Rediriger plusieurs sorties : `$ commande 1 > fichier1 2 > fichier2`

`#!/bin/bash`

La séquence `#!` est appelée shebang : elle indique à l'OS que le fichier est un script. Juste derrière, on indique le chemin vers un interpréteur capable d'exécuter le script

Pour que la création ou la modification d'une variable soit permanente, il faut ajouter la commande au fichier `~/.bashrc` 1 qui est lu à chaque démarrage de bash

Exercice 1 :

1. `printenv PATH` : `/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin`
2. `HOME`
- 3.
4. `MY_VAR=...`
5. la variable `MY_Var` n'existe plus après la commande `bash`
6. `export MY_VAR=`
7. `export NOM='prénom nom '`
8. `echo "Bonjour à vous, $NOM"` ou `echo 'Bonjour à vous,' $NOM`
9. `unset` supprime la variable, sinon la variable existe toujours mais est vide
10. `echo '$HOME=' $HOME`

`PATH=$PATH:/chemin` par ex `$PWD`

Exercice 2 :

Contrôle mot de passe :

```
#!/bin/bash
testpwd() {
    PASSWORD="azerty"
    read -s -p 'mot de passe :' mdp
    if [ $_mdp == $_PASSWORD ]; then
        echo -e "\nMot de passe correct!"
    else
        echo -e "\nMot de passe incorrect!"
    fi
}
testpwd
```

Calcul factorielle :

```
#!/bin/bash
factorielle() {
    local RESULT=1
    if [ $# = 0 ]; then
        echo 'Pas de nombre'
    else
        for i in $(seq 1 $1)
        do
            ((RESULT=$i*RESULT))
        done
        echo "factorielle $1 est égale à $RESULT"
    fi
}

factorielle $1
```

```
#!/bin/bash
statistiques() {
    PARAM_CHECKED=0
    function is_number()
    {
        re='^[+-]?[0-9]+([.][0-9]+)?$'
        if ! [[ $1 =~ $re ]] ; then
            echo 1
        else
            echo 0
        fi
    }
    for param in $* ; do
        if [ $(is_number $param) -eq 1 ]; then
            echo "Paramètre non valide"
        else
            PARAM_CHECKED=$((PARAM_CHECKED + 1))
        fi
    done
    if [ $PARAM_CHECKED == $# ]; then
        tab=()
        for i in $(seq 1 $#); do
            tab[$i]=$1
            shift
        done
        echo $tab
        MAX=-100
        MIN=100
        SOMME=0
        for element in $(seq 1 ${#tab[*]}); do
            if [ $element -gt $MAX ]; then
                MAX=$element
            elif [ $element -lt $MIN ]; then
                MIN=$element
            fi
            SOMME=$((SOMME + $element))
        done
        MOYENNE=$((SOMME/${#tab[*]}))
        echo "Valeur min : $MIN"
        echo "Valeur max : $MAX"
        echo "Valeur moyenne : $MOYENNE"
    fi
}

statistiques $1 $2 $3
```

```
#!/bin/bash
affiche_heure() {
    now=$(date +%T)
    hour=$(cut -d ':' -f1 <<< $now)
    echo $hour
    if [ $((hour)) -lt 15 ] && [ $((hour)) -gt 6 ];
    then
        echo "$now bonne soirée"
    elif [ $((hour)) -lt 17 ] && [ $((hour)) -gt 15 ];
    then
        echo "$now bonne fin de journée!"
    elif [ $((hour)) -lt 23 ] && [ $((hour)) -gt 17 ];
    then
        echo "$now bonne journée!"
    elif [ $((hour)) -lt 6 ] && [ $((hour)) -gt 23 ];
    then
        echo "$now bonne nuit!"
    fi
}
affiche_heure
```