**les paramètres positionnels correction**

**Exercice 1 : Affichage des arguments**

**Écrivez un script Bash qui affiche tous les arguments passés en ligne de commande.**

#!/bin/bash

# Affiche tous les arguments passés en ligne de commande

echo "Les arguments passés sont : \$1=$1, \$2=$2, \$3=$3, ..."

**Exercice 2 : Comptage des arguments**

**Écrivez un script Bash qui compte le nombre total d'arguments passés en ligne de commande.**

#!/bin/bash

# Compte le nombre total d'arguments passés

nombre\_arguments=$#

echo "Nombre total d'arguments : $nombre\_arguments"

**Exercice 3 : Concaténation d'arguments**

**Écrivez un script Bash qui concatène tous les arguments passés en une seule chaîne.**

#!/bin/bash

# Concatène tous les arguments passés en une seule chaîne séparée par un espace

arguments\_concatenes=""

for arg in "$@"; do

arguments\_concatenes="$arguments\_concatenes $arg"

done

echo "Arguments concaténés : $arguments\_concatenes"

**Exercice 4 : Calcul de la somme**

**Écrivez un script Bash qui prend trois nombres en tant qu'arguments et affiche leur somme.**

#!/bin/bash

# Vérifie s'il y a trois arguments

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

# Calcule la somme des trois nombres

somme=$(( $1 + $2 + $3 ))

echo "La somme des trois nombres est : $somme"

else

echo "Usage: $0 <nombre1> <nombre2> <nombre3>"

fi

**Exercice 5 : Génération de noms d'utilisateurs**

**Écrivez un script Bash qui génère des noms d'utilisateurs en utilisant les deux premières lettres de chaque argument passé.**

#!/bin/bash

# Vérifie s'il y a au moins un argument

if [ "$#" -ge 1 ]; then

# Génère des noms d'utilisateurs en utilisant les deux premières lettres de chaque argument

for arg in "$@"; do

username="${arg:0:2}"

echo "Nom d'utilisateur pour $arg : $username"

done

else

echo "Usage: $0 <nom1> [nom2] [nom3] ..."

fi

**Exercice 6 : Validation d'un mot de passe**

**Écrivez un script Bash qui vérifie si un mot de passe passé en argument satisfait certaines conditions (longueur minimale, présence de chiffres, etc.).**

#!/bin/bash

# Vérifie s'il y a un argument

if [ -n "$1" ]; then

password="$1"

# Vérifie les conditions du mot de passe

if [[ ${#password} -ge 8 && "$password" =~ [0-9] && "$password" =~ [A-Za-z] ]]; then

echo "Le mot de passe est valide."

else

echo "Le mot de passe ne satisfait pas les conditions requises."

fi

else

echo "Usage: $0 <mot\_de\_passe>"

fi

Modérée

**Exercice 7 : Calcul de la moyenne**

**Écrivez un script Bash qui prend un nombre variable d'arguments (notes) et calcule leur moyenne.**

#!/bin/bash

# Vérifie s'il y a au moins une note

if [ "$#" -ge 1 ]; then

# Calcule la moyenne des notes

somme=0

for note in "$@"; do

somme=$(( somme + note ))

done

moyenne=$(bc -l <<< "scale=2; $somme / $#")

echo "La moyenne des notes est : $moyenne"

else

echo "Usage: $0 <note1> [note2] [note3] ..."

fi

**Exercice 8 : Gestion de fichiers**

**Écrivez un script Bash qui prend en argument un répertoire, puis affiche la taille de chaque fichier dans ce répertoire.**

#!/bin/bash

# Vérifie s'il y a un argument

if [ -n "$1" ]; then

repertoire="$1"

# Vérifie que le chemin est un répertoire

if [ -d "$repertoire" ]; then

# Affiche la taille de chaque fichier dans le répertoire

for fichier in "$repertoire"/\*; do

taille=$(du -h "$fichier" | cut -f1)