

IUT DE COLMAR

R405 AUTOMATISATION DES TÂCHES

ANNÉE 2022-23

TP 2 : Bash

MARTIN BAUMGAERTNER

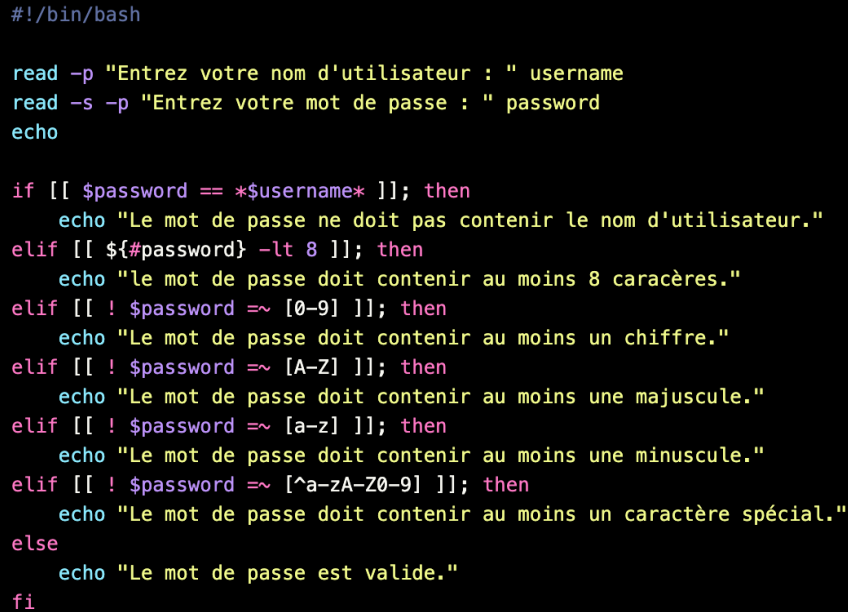
29 mars 2023

Table des matières

1	Exercice 1	2
2	Exercice 2	3
3	Exercice 3	4

1 Exercice 1

Voici donc le script qui vérifie toutes les conditions une à une.



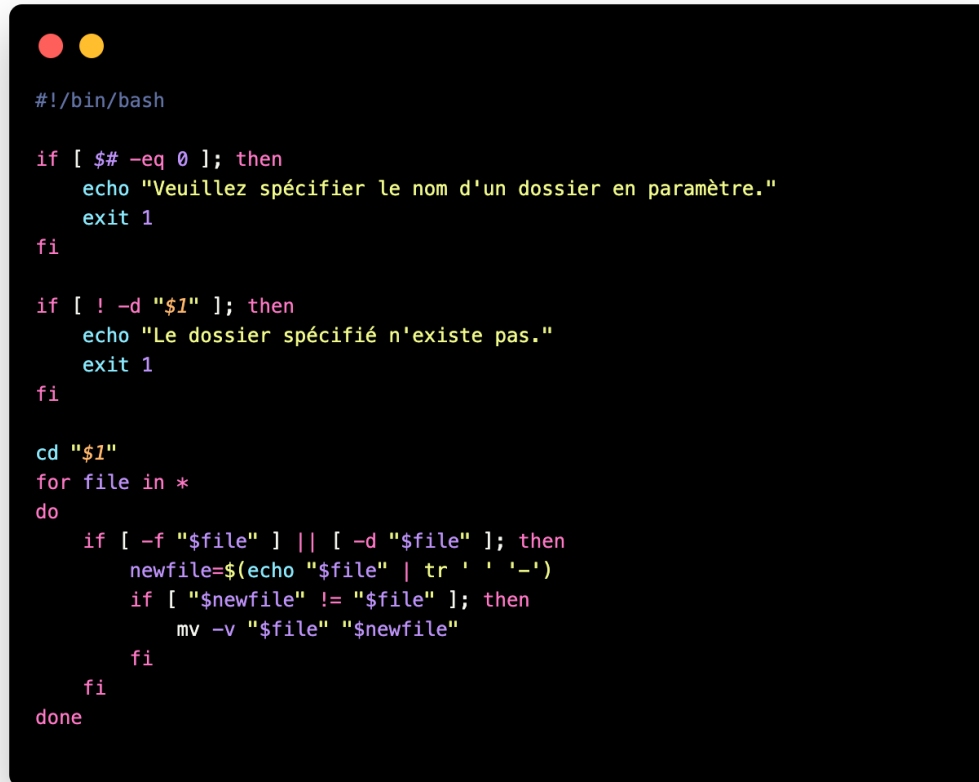
```
#!/bin/bash

read -p "Entrez votre nom d'utilisateur : " username
read -s -p "Entrez votre mot de passe : " password
echo

if [[ $password == *$username* ]]; then
    echo "Le mot de passe ne doit pas contenir le nom d'utilisateur."
elif [[ ${#password} -lt 8 ]]; then
    echo "le mot de passe doit contenir au moins 8 caractères."
elif [[ ! $password =~ [0-9] ]]; then
    echo "Le mot de passe doit contenir au moins un chiffre."
elif [[ ! $password =~ [A-Z] ]]; then
    echo "Le mot de passe doit contenir au moins une majuscule."
elif [[ ! $password =~ [a-z] ]]; then
    echo "Le mot de passe doit contenir au moins une minuscule."
elif [[ ! $password =~ [^a-zA-Z0-9] ]]; then
    echo "Le mot de passe doit contenir au moins un caractère spécial."
else
    echo "Le mot de passe est valide."
fi
```

FIGURE 1 – Exercice 1

2 Exercice 2

A terminal window with a dark background and light-colored text. The script is a bash script that checks the number of arguments, checks if a directory exists, and then iterates over files in the directory to rename them based on a pattern. The script uses `if`, `echo`, `exit`, `cd`, `for`, `do`, `fi`, and `done` keywords. The file names are transformed using `tr` to replace spaces with hyphens.

```
#!/bin/bash

if [ $# -eq 0 ]; then
    echo "Veuillez spécifier le nom d'un dossier en paramètre."
    exit 1
fi

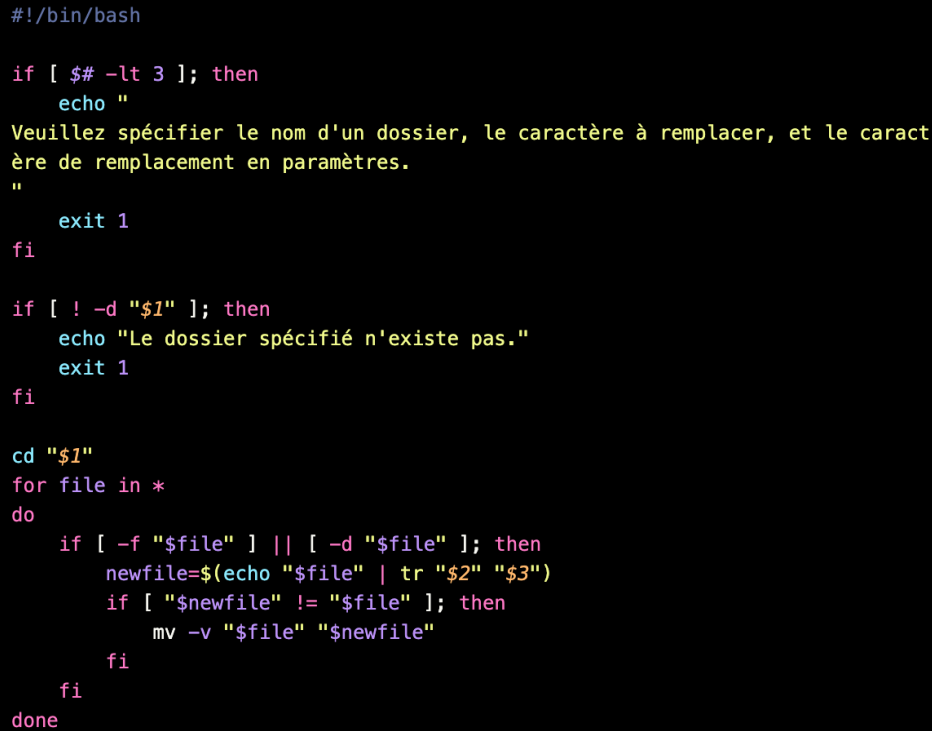
if [ ! -d "$1" ]; then
    echo "Le dossier spécifié n'existe pas."
    exit 1
fi

cd "$1"
for file in *
do
    if [ -f "$file" ] || [ -d "$file" ]; then
        newfile=$(echo "$file" | tr ' ' '-')
        if [ "$newfile" != "$file" ]; then
            mv -v "$file" "$newfile"
        fi
    fi
done
```

FIGURE 2 – Exercice 2

On vient vérifier la condition qui vérifie si le nombre de paramètres passés en entrée est inférieur à 3. Si c'est le cas, le script affiche un message d'erreur demandant à l'utilisateur de spécifier le nom d'un dossier, le caractère à remplacer, et le caractère de remplacement en paramètres, puis le script se termine avec un code d'erreur 1.

3 Exercice 3



```
#!/bin/bash

if [ $# -lt 3 ]; then
    echo "
    Veuillez spécifier le nom d'un dossier, le caractère à remplacer, et le caract
    ère de remplacement en paramètres.
    "
    exit 1
fi

if [ ! -d "$1" ]; then
    echo "Le dossier spécifié n'existe pas."
    exit 1
fi

cd "$1"
for file in *
do
    if [ -f "$file" ] || [ -d "$file" ]; then
        newfile=$(echo "$file" | tr "$2" "$3")
        if [ "$newfile" != "$file" ]; then
            mv -v "$file" "$newfile"
        fi
    fi
done
```

FIGURE 3 – Exercice 3

Le code présent dans la condition else vérifie si l'option -r n'est pas spécifiée. Dans ce cas, le script change le répertoire de travail courant pour le répertoire spécifié dans le premier paramètre. Ensuite, le script itère sur chaque fichier et dossier dans le répertoire courant, en vérifiant s'il s'agit d'un fichier ou d'un dossier. Le script remplace les caractères spécifiés dans le deuxième paramètre par les caractères de remplacement spécifiés dans le troisième paramètre pour chaque fichier ou dossier, et les renomme avec le nouveau nom s'il est différent de l'ancien nom. En fin de compte, si l'option -r n'est pas spécifiée, le script ne parcourt que le répertoire courant et renomme les fichiers et dossiers dans ce répertoire en fonction des caractères spécifiés.