
Exercices UNIX : Correction

Total :

5/20

Votre fichier `exo1.sh` est vide.

1 Exercices 5

1.1 Exercice 1 2

Fichier vide. Heureusement, Emacs vous a sauvegardé un fichier `#exo1.sh#`. Il manque le mot-clé `fi` pour conclure ce dernier.

Exemple de solution :

```
#!/bin/zsh
#
# Appelle sed.

if [ $# -ne 2 ]; then
    echo "Usage : $0 chaine1 chaine2" 1>&2
    echo "Remplace toutes les occurences de chaine1 sur l'entree standard" 1>&2
    echo "par des occurences de chaine2 sur la sortie standard." 1>&2
    exit 1
fi

sed s/$1/$2/g
```

1.2 Exercice 2 0

Faux.

Exemple de solution :

```
#!/bin/zsh

if [[ $# -ne 2 && $# -ne 3 ]]; then
    echo "Usage : $0 chaine1 chaine2 [fichier]" 1>&2
    echo "Remplace toutes les occurences de chaine1 dans fichier" 1>&2
    echo "par des occurences de chaine2 sur la sortie standard." 1>&2
    exit 1
fi
```

```
sed s/$1/$2/g $3
```

1.3 Exercice 3

2

Moins d'erreurs que dans le script précédent. Vos structures de test `if then else fi` sont correctes, mais attention aux espaces dans vos tests : `if [$# -eq 3]` et non pas `if [$# - eq 3]`. Vous persistez aussi à mettre un simple *quote* non fermé au milieu de votre message d'erreur : `print "'Usage : ./exo1 chaine1 chaine2"`.

Exemple de solution :

```
#!/bin/zsh

if [[ $# -ne 2 && $# -ne 3 ]]; then
    echo "Usage : $0 chaine1 chaine2 [fichier]"           1>&2
    echo "Remplace toutes les occurrences de chaine1 dans fichier" 1>&2
    echo "par des occurrences de chaine2."               1>&2
    exit 1
fi

sed -i s/$1/$2/g $3
```

1.4 Exercice 4

1

Vous confondez simples *quotes* « ' » et *backquotes* « ` » ; ces derniers indiquent que le texte encadré doit être exécuté. Il faut donc remplacer `aux= "'echo$3|sed s/$1/$2/g'"` par `aux= "`echo $3|sed s/$1/$2/g`"` (notez aussi l'espace entre `echo` et `$3` qui indique qu'on appelle la commande `echo` avec le paramètre `$3`, et sans lequel l'interpréteur comprend qu'il faut exécuter une commande nommée `echo$3`).

Exemple de solution :

```
#!/bin/zsh

if [[ $# -ne 2 && $# -ne 3 ]]; then
    echo "Usage : $0 chaine1 chaine2 [fichier]"           1>&2
    echo "Remplace toutes les occurrences de chaine1 dans fichier" 1>&2
    echo "par des occurrences de chaine2. Change aussi le nom"    1>&2
    echo "du fichier en conséquence."                       1>&2
    exit 1
fi

sed -i s/$1/$2/g $3
```

```
if [ -f $3 ]; then
  mv $3 'echo $3 | sed s/$1/$2/g'
fi
```

1.5 Exercice 5

0

Exemple de solution :

```
#!/bin/zsh

if [ $# -lt 2 ]; then
  echo "Usage : $0 chaine1 chaine2 [fichiers...]" 1>&2
  echo "Remplace toutes les occurrences de chaine1 dans les fichiers" 1>&2
  echo "par des occurrences de chaine2. Change aussi les noms" 1>&2
  echo "des fichiers en conséquence." 1>&2
  exit 1
fi

chaine1=$1
chaine2=$2
shift; shift

sed -i s/$chaine1/$chaine2/g $*

for file in $*; do
  if [ -f $file ]; then
    mv $file 'echo $file | sed s/$chaine1/$chaine2/g'
  fi
done
```