

# INF600A: Laboratoire #1

## Pipelines avec sed, find, xargs, etc.

### Solution

---

1.

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

```
#####  
# Lister les fichiers du repertoire dont le nom et l'extension sont identi  
# Cherche dans le repertoire courant si aucun repertoire n'est specifie.  
#####
```

```
if [[ $# == 0 ]]; then  
    REPERTOIRE='.'
```

```
else
```

```
    REPERTOIRE="$1"
```

```
fi
```

```
ls "$REPERTOIRE" |  
grep  "^\(.*\)\\.\\1$"
```

---

## 2.

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

```
#####  
# Supprimer les blancs en fin de ligne dans un fichier  
#####
```

```
if [[ $# == 0 ]]; then  
    echo "usage:"  
    echo "$0 nomDeFichier"  
    exit -1  
fi
```

```
FICHIER="$1"
```

```
sed 's/[ \t]*$//' "$FICHIER" > "$FICHIER".sans-blanc
```

```
echo "** Appel a diff ordinaire: on devrait voir des differences"  
diff "$FICHIER" "$FICHIER".sans-blanc
```

```
echo "** Appel a diff en ignorant les blancs: on ne devrait pas voir de di  
diff -iwB "$FICHIER" "$FICHIER".sans-blanc
```

---

### 3.

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

```
#####  
# Trouver les noms d'usagers dans /etc/passwd  
#####
```

```
# Rappel :
```

```
# - Les champs sont separes par ":"
```

```
# - Le nom d'usager est le premier champ
```

```
# - Il peut y avoir des lignes de commentaires -- debutant par "#"
```

```
# Note: Une solution avec awk est aussi presentee.
```

```
# Solution avec grep -E -o.
```

```
# -o: emet la partie matchee, et non la ligne complete.
```

```
cat /etc/passwd |  
grep -v "^#" |  
grep -E -o '^[^:]*' |  
sort >res-grep
```

```
# Solution avec sed.
```

```
# ^\[^\:]*\) => matche et capture tout ce qui va jusqu'au premier ":", i.e
```

```
# On remplace ensuite toute la ligne par le username
```

```
cat /etc/passwd |  
sed '/^#/d' |  
sed 's/^\[^\:]*\).*\/\1/' |  
sort >res-sed
```

```
# Solution avec awk: la plus directe!
```

```
cat /etc/passwd |  
awk -F: '/^[^#]/ { print $1 }' |  
sort >res-awk
```

```
# On verifie que les solutions produisent toutes le meme resultat.
```

```
if cmp -s res-grep res-sed && cmp -s res-grep res-awk; then
```

```
    cat res-sed
```

```
    echo
```

```
    echo "+++ Ok!"
```

```
else
```

```
    echo "--- Certains resultats sont differents?! A verifier!"
```

```
fi
```

---

#### 4.

```
#!/bin/bash -

#####
# Trouver les noms des enseignants qui ont utilise OTO
#####

# date: ^[^\]*| = tout jusqu'au premier |
# heure: [^\]*| = tout jusqu'au deuxieme |
# nom usager: \( [^\]*\) = entre 2e et 3e |
# commande: .* = tout ce qui suit le 3e |
#
# Ensuite, on remplace tout par \1, i.e., on ne conserve que le nom usager

cat Fichiers/journal-operations.txt |
sed 's/^[^\]*|[^\]*| \( [^\]*\) .*/\1/' | # Voir plus haut
grep -E -v [a-z]{2}[0-9]{6} | # On filtre les codes d'etudiants
sort -u

# Solution avec awk
cat Fichiers/journal-operations.txt |
awk -F\| '{ print $3 }' |
tr -d ' ' |
grep -E -v [a-z]{2}[0-9]{6} | # On filtre les codes d'etudiants
sort -u
```

---

Solutions alternatives pour la version avec `grep`, en utilisant une expression régulière simple et non étendue :

```
grep -v "[a-z]\{2\}[0-9]\{6\}"
grep -v [a-z]\{2\}[0-9]\{6\}
```

Mais la solution suivante ne fonctionne pas!

```
grep -v [a-z]\{2\}[0-9]\{6\}
```

---

```
$ cat foo.sh
echo ab123456 | grep [a-z]\{2\}[0-9]\{6\}
$ bash -x foo.sh
+ echo ab123456
+ grep '[a-z]{2}[0-9]{6}'

$ cat foo2.sh
echo ab123456 | grep [a-z]\{2\}[0-9]\{6\}
$ bash -x foo2.sh
+ echo ab123456
+ grep '[a-z]\{2\}[0-9]\{6\}'
```

---

5.

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

```
#####  
# Trouver les fichiers qui contiennent des url sans http/https  
#####
```

```
find . -name "*.tex" |  
xargs grep "[\]url{" |  
sed 's/^\[[:]*\) :.*url{\([^\)]*\).*$/\1 \2/' |  
grep -v "http"
```

---

## 6.

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

```
#####  
# Script pour deplacer le depot distant de malt vers github, en  
# changeant aussi le nom d'utilisateur.  
#####
```

```
AVANT=malt.labunix.uqam.ca/bidon_joe/GIT  
APRES=github.com/bidon-joe
```

```
# Cette solution est plus complexe car elle modifie uniquement la  
# ligne avec l'attribut 'url'.
```

```
find . -name "config" |  
grep ".*[.]git/config$" |  
xargs sed -i.bak "s|^\[[:blank:]]*url\[[:blank:]]*?=.**\)$AVANT|\1$APRES|"
```