

「 TD3 - Traitement de fichiers 」

↳ **Exercice 1 : Lecture de fichiers** ↵

La commande `cat xxx` affiche le contenu du fichier `xxx`. Sur votre bureau, créez un fichier `exemple.txt` rempli d'un texte de plusieurs dizaines de lignes. Affichez-le via la commande `cat`.

↳ **Exercice 2** ↵

La commande `more xxx` affiche le contenu du fichier `xxx` page par page. Testez sur le fichier `exemple.txt`.

↳ **Exercice 3** ↵

La commande `head xxx` affiche le début du fichier `xxx`. Affichez le début du fichier `exemple.txt`.

↳ **Exercice 4** ↵

La commande `tail xxx` affiche la fin du fichier `xxx`. Affichez la fin du fichier `exemple.txt`.

↳ **Exercice 5** ↵

Peut-on afficher seulement les 5 dernières lignes d'un fichier ? les 5 premières lignes ?

↳ **Exercice 6 : Chercher un motif** ↵

Afin de rechercher un motif (par exemple une chaîne de caractères ou un caractère), il est possible d'utiliser la commande `grep`. `grep mot xxx` permet donc de chercher les occurrences de `mot` dans le fichier `xxx`.

- Cherchez un mot présent (plusieurs fois) dans le fichier `exemple.txt` puis un mot non présent.
- Exécutez, dans le répertoire où se trouve le fichier `exemple.txt`, la commande `grep yyy *` où `yyy` est un mot présent dans le fichier `exemple.txt`.
- En utilisant le man, à quoi servent les options `-l` et `-r` ? Quel résultat donne la commande `grep -rl yyy *` exécutée à la racine de votre compte ? `yyy` étant un mot présent dans le fichier `exemple.txt`.

- Quel est l'usage de l'option `-i` ? de l'option `-v` ?
- Le répertoire `/usr/share/doc` contient, pour chaque logiciel installé, une liste d'informations. Via la commande `grep`, recherchez quel logiciel est lié à une personne nommée "luc pionchon".

↳ Exercice 7 : Chercher un fichier [⌈]

La commande `find` permet de chercher un ou plusieurs fichiers ou répertoires, en particulier en fonction de leur nom. `find rep -name "xxx"` cherche tous les fichiers/répertoires nommés `xxx` dans le répertoire `rep` et ses sous-répertoires (dans toute l'arborescence contenue sous le répertoire `rep`). De même, `find . -name "*.txt"` cherche l'ensemble des fichiers d'extensions `.txt` dans le répertoire courant et ses sous-répertoires.

- Depuis votre racine de compte, cherchez le fichier `exemple.txt` via la commande `find`. Utilisez la wildcard `*`.
- Existe-t-il des fichiers d'extension `doc` dans l'un des sous-répertoires du répertoire `/usr/share/doc` ?

↳ Exercice 8 [⌈]

La commande `find` possède de nombreuses options et possibilités. Par exemple, il est possible de chercher les fichiers d'une certaine taille. L'option `-size +XXX` permet de chercher les fichiers d'une taille d'au moins `XXX`. Après avoir lu le manuel, cherchez les fichiers d'une taille supérieure à 5 mégaoctets dans le répertoire `/usr/bin/`. Peut-on chercher des fichiers plus petit que 50 octets ?

↳ Exercice 9 : Compter les lignes et les mots [⌈]

La commande `wc xxx` permet d'afficher des statistiques sur le fichier `xxx`. Combien de mots, de caractères et de lignes sont contenus dans le fichier `exemple.txt` ?

↳ Exercice 10 : Redirections [⌈]

Il est possible de rediriger la sortie d'une commande vers un fichier en utilisant le caractère `>`. Par exemple `ls > maliste` va écrire le résultat de la commande `ls` dans le fichier `maliste`. Si le fichier `maliste` n'existe pas il est créé. Si le fichier `maliste` existe avant la commande, son contenu antérieur est alors écrasé. Si l'on veut que le contenu antérieur ne soit pas écrasé, la redirection `>>` écrit le résultat de la commande à la suite du contenu antérieur. `>>` ajoute donc le résultat de la commande en fin du fichier.

- La commande `date` permet d'avoir l'heure. Créer un fichier contenant l'heure de sa création. Y ajouter le résultat de la commande `ls`.
- Comment faire une copie d'un fichier sans utiliser `cp` ni ouvrir d'éditeur de texte ?
- Comment noter, dans un fichier `monFichier`, l'ensemble des fichiers d'extension `doc` contenus dans un des sous-répertoires du répertoire `/usr/share/doc` ?
- Sans les compter manuellement, combien de fichiers de taille supérieure à 5MB sont présents dans `/usr/bin` ?

↳ Exercice 11 [⌈]

Il est possible de connecter la sortie d'une commande sur l'entrée d'une autre. Par exemple il est ennuyeux d'écrire :

```
ls /usr/bin/a* > /tmp/maListe
wc -l /tmp/maListe
rm /tmp/maListe
```

pour simplement compter le nombre de fichiers commençant par a dans /usr/bin. Via le caractère |, on peut remplacer la séquence suivante par : ls /usr/bin/a* | wc -l. Le résultat de la commande ls /usr/bin/a* devenant l'entrée de la commande wc -l.

- ls -l liste le contenu d'un répertoire. ls -l /usr/bin/ est peu lisible faute à un très grand nombre de fichiers. La commande more permet d'afficher page par page. Ainsi ls -l /usr/bin/ | more permet de lister le contenu d'un répertoire et de l'afficher page par page. Testez.
- Comment compter en une seule commande le nombre de fichiers de taille supérieure à 5MB présents dans /usr/bin ?
- Que va donner la commande ls /usr/share/doc/ | grep ab ?
- Combien de fois apparaît le mot "le" dans les 15 premières lignes du fichier exemple.txt ?