

Systèmes d'exploitation I

Module: M3 - LFSMI - S3

Prof. R. HANNANE

r.hannane@uca.ma

Faculté des Sciences Semlalia, Marrakech
Université Cadi Ayyad



Les commandes de manipulation du contenu de fichier

9) Afficher le contenu du fichier

- ⑩ **cat** : (conCATenate) est une **commande multi-usage qui permet d'afficher le contenu des fichiers texte et aussi de concaténer des fichiers**

► Exemple:

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ cat fich1.txt
1
2
3
4
5
6
7
8
9
10
11
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```

Les commandes de manipulation du contenu de fichier

9) Afficher le contenu du fichier

Attention:

- La commande **cat** ne pose pas de problème pour l'affichage du contenu des petits fichiers
⇒ Mais elle reste un mauvais choix pour afficher le contenu des fichiers larges
- Pour afficher le contenu des fichiers large on aura besoin de commande qui permet d'afficher le fichier page par page et d'offrir aussi la possibilité de naviguer à travers le fichier en utilisant les touches du clavier

Les commandes de manipulation du contenu de fichier

9) Afficher et naviguer dans le contenu du fichier

- ⑪ **more** : affiche le contenu des fichiers larges page par page et offre aussi la possibilité de naviguer à travers le fichier en utilisant les touches du clavier

- ⑫ **less** : offre plusieurs fonctionnalités pour afficher les fichiers larges
 - ▶ C'est une commande qui est basé sur **more** (\Rightarrow c'est une amélioration de **more**)
 - ▶ Elle possède plus de fonctionnalités que **more**
 - ▶ C'est la commande utilisée pour afficher les pages **man**

Remarque:

- **Les deux commandes more et less utilisent les mêmes raccourcis clavier pour naviguer dans un fichier**

Les commandes de manipulation du contenu de fichier

9) Afficher et naviguer dans le contenu du fichier

- ▶ **Pour naviguer à travers le contenu d'un fichier affiché par la commande `less`, on utilise des touches claviers**
- ▶ **La touche `h` permet de visualiser l'aide**
- ▶ **La plupart de ses touches sont similaires pour `less` et `more`**
- ▶ Le tableau suivant résume les plus importantes:

Mouvement	Touche clavier
Page suivante	espace
Page précédente	b
Ligne suivante	entrée
quitter	q
aide	h

Les commandes de manipulation du contenu de fichier

9) Afficher et naviguer dans le contenu du fichier

- ▶ Pour chercher dans l'affichage de `less`:
 - à partir de notre emplacement vers la fin du fichier on utilise la touche “`/`” suivie du terme à rechercher puis on tape entrée
 - à partir de notre emplacement vers le début du fichier on utilise la touche “`?`” suivie du terme à rechercher puis on tape entrée
- ▶ Pour naviguer entre les termes trouvées on utilise les touches “`n`” (aller au suivant) et “`N`” (aller au précédent)
- ⇒ Toutes les correspondances trouvées seront mise en surbrillance

Les commandes de manipulation du contenu de fichier

10) Chercher dans le contenu du fichier

⑬ **grep** : filtrer le contenu d'un fichier en se basant sur les lignes correspondant à un terme donné

- ▶ Le terme cherché peut être un mot simple ou fournit à travers des expressions régulières
- ▶ L'option **--color** permet d'afficher le terme recherché en **rouge** ⇒ *Dans la plupart des distributions, il existe un alias (alias grep='grep --color') qui permet d'utiliser l'option --color automatiquement*
- ▶ **Exemple:** Afficher que les lignes contenant le terme alaoui à partir du fichier *personne.txt* qui se trouve dans le chemin décrit

```
abdessanad@abdessanad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ grep --color alaoui /home/abdessamad/Desktop/SE1/test/personne.txt
souad:alaoui:67
souad:elalaoui:42
raflk:elalaout:31
abdessanad@abdessanad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$
```



Les commandes de manipulation du contenu de fichier

10) Chercher dans le contenu du fichier

- ▶ L'option `-c` de `grep` affiche combien de ligne correspondant au terme recherché
- ▶ Exemple:

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ grep -c alaoui /home/abdessamad/Desktop/SE1/test/personne.txt
3
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$
```

- ▶ L'option `-n` de `grep` affiche les numéros des lignes du fichier original correspondante au terme recherché
- ▶ Exemple:

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ grep -n alaoui /home/abdessamad/Desktop/SE1/test/personne.txt
1:said:alaoui:67
3:souad:elalaoui:42
4:rafik:elalaoui:31
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$
```

Les commandes de manipulation du contenu de fichier

10) Chercher dans le contenu du fichier

▶ Autres exemple de grep

commande	résultat	Description de l'option
grep -v nologin /etc/passwd	Toutes les lignes qui ne contiennent pas « nologin » dans /etc/passwd	-v:permet de sélectionner les lignes qui ne coresspondent pas
grep -l linux /etc/*	Afficher tous les fichiers contenu dans /etc et qui contiennent le terme linux	-l:affiche uniquement les fichiers où il ya correspondnce
grep -i linux /etc/*	Afficher toutes les lignes de tous les fichiers de /etc contenant le terme linux (en majuscule ou en minuscule)	-i:ignorer la case
grep -w linux /etc/*	Afficher toutes les lignes de /etc contenant le mot entier linux.	-w:sélectionner les lignes contenant le mot entier

Les commandes de manipulation du contenu de fichier

10) Chercher dans le contenu du fichier

```
abdessamad@abdessamad-HP-Pavillon-Laptop-15-eg0xxx:~/Desktop/SE1/test$ grep -v alaoui /home/abdessamad/Desktop/SE1/test/personne.txt
rachid:benkhelifa:35
abdessamad@abdessamad-HP-Pavillon-Laptop-15-eg0xxx:~/Desktop/SE1/test$ ■

abdessamad@abdessamad-HP-Pavillon-Laptop-15-eg0xxx:~/Desktop/SE1/test$ grep -l alaoui /home/abdessamad/Desktop/SE1/test/personne.txt
/home/abdessamad/Desktop/SE1/test/personne.txt
abdessamad@abdessamad-HP-Pavillon-Laptop-15-eg0xxx:~/Desktop/SE1/test$ ■

abdessamad@abdessamad-HP-Pavillon-Laptop-15-eg0xxx:~/Desktop/SE1/test$ grep -i alaoui /home/abdessamad/Desktop/SE1/test/personne.txt
said:alaoui:67
souad:elalaoui:42
rafik:elalaoui:31
abdessamad@abdessamad-HP-Pavillon-Laptop-15-eg0xxx:~/Desktop/SE1/test$ ■

abdessamad@abdessamad-HP-Pavillon-Laptop-15-eg0xxx:~/Desktop/SE1/test$ grep -w alaoui /home/abdessamad/Desktop/SE1/test/personne.txt
said:alaoui:67
abdessamad@abdessamad-HP-Pavillon-Laptop-15-eg0xxx:~/Desktop/SE1/test$ ■
```

Les commandes de manipulation du contenu de fichier

11) Trier le contenu du fichier

14) **sort** : trie les lignes d'un fichier par ordre alphanumérique ou par ordre de valeur en se basant sur plusieurs champs contenus de la ligne

- ▶ Les champs contenus de la ligne doivent être séparés par un séparateur qui peut être par exemple un espace ou bien : ou bien | ou
- ▶ Cette commande acte sur les fichiers contenant un délimiteur qui est par défaut espace
- ▶ **Exemple:** Trier les lignes du fichier *fich3.txt*

```
abdessamad@abdessamad-HP-Pavillon-Laptop-15-eg0xxx:~/Desktop/SE1/test/test1$ sort fich3.txt
avril
fevrier
janvier
mars
abdessamad@abdessamad-HP-Pavillon-Laptop-15-eg0xxx:~/Desktop/SE1/test/test1$
```

Les commandes de manipulation du contenu de fichier

11) Trier le contenu du fichier

- ▶ L'option **-t** permet de spécifier le séparateur si les champs d'une ligne d'un fichier sont séparés par un autre caractère séparateur différent d'espace(ex: ; ou :)
- ▶ L'option **-k** permet de spécifier le champ suivant lequel on va trier. Cette option doit être suivie d'un argument indiquant le numéro du champ (le premier champ ayant le numéro 1)
- ▶ L'option **-n** permet de **faire un tri numérique des champs**
- ▶ L'option **-r** permet de **faire un tri par l'ordre inverse (décroissant)**

Remarque:

- **Le tri peut être effectué suivant les valeurs de plusieurs champs**

Les commandes de manipulation du contenu de fichier

11) Trier le contenu du fichier

► Exemple1:

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ sort -t. -k1n -k2n -k3n -k4n test02.txt
14.15.13.2
124.135.12.02
168.100.1.06
168.100.1.14
168.100.2.30
168.200.13.2
192.168.1.10
200.13.3.1
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```

► Exemple2:

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ sort -t. -n -k1 -k2 -k3 -k4 test02.txt
14.15.13.2
124.135.12.02
168.100.1.06
168.100.1.14
168.100.2.30
168.200.13.2
192.168.1.10
200.13.3.1
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```

Les commandes de manipulation du contenu de fichier

11) Trier le contenu du fichier

- ▶ Dans **Exemple2** on a utilisé :
 - ▶ L'option **-t** avec l'argument **.** pour indiquer que les champs de la ligne sont séparés par **.**
 - ▶ L'option **-n** pour faire un tri par valeur numérique
 - ▶ L'option **-k** avec l'argument **1,2,3 et 4** pour tri les lignes suivant la valeur du 1er champ, ensuite par le 2ème champ ensuite par le 3ème champ ensuite par le 4ème champ

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ sort -t. -n -k1 -k2 -k3 -k4 test02.txt
14.15.13.2
124.135.12.02
168.100.1.06
168.100.1.14
168.100.2.30
168.200.13.2
192.168.1.10
200.13.3.1
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$
```

Les commandes de manipulation du contenu de fichier

11) Trier le contenu du fichier

- **Exemple:** Soit le fichier “*personnes.txt*” contenant les données suivantes:

```
said:alaoui:67  
rachid:benkhelifa:35  
rafik:elouali:31  
souad:elouali:42
```

⇒ Trier les personnes contenues dans ce fichier par nom puis par prénom ensuite par age

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ sort -t: -k2 -k1 -k3n personne.txt  
said:alaoui:67  
rachid:benkhelifa:35  
rafik:elalaoui:31  
souad:elalaoui:42  
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec cut

15) **cut** : extraie des colonnes à partir d'un fichier ou à partir de l'entrée standard

- ▶ Cette commande **acte sur les fichiers contenant un délimiteur qui est par défaut espace** (comme **sort**)
- ▶ L'option **-d** de **cut** permet de **spécifier un délimiteur différent d'espace**
- ▶ L'option **-f num** permet de spécifier les champs à extraire:
 - Le premier champ étant numéro 1
 - **Ces numéro de champs peuvent être donnés sous forme de valeurs séparés par des virgules ou sous forme intervalle**
 - Exemple de la liste de numéro de champ (colonne)
 - 3-9** : de 3 à 9
 - 1,3,5** : 1, 3 et 5
 - 5** : de 1 à 5
 - 5-** : de 5 à la fin de la ligne

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec cut

- ▶ **Exemple1:** Afficher le 1^{er} et le 2^{eme} champ du fichier *test02.txt*

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ cut -d. -f 2 test02.txt
```

```
192.168  
168.100  
14.15  
200.13  
168.200  
124.135  
168.100  
168.100
```

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```

- ▶ **Exemple2:** Afficher le 1^{er}, le 3^{eme}, et le 4^{eme} champ du fichier *test02.txt*

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ cut -d. -f 1,3,4 test02.txt
```

```
192.1.10  
168.2.30  
14.13.2  
200.3.1  
168.13.2  
124.12.02  
168.1.14  
168.1.06
```

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec cut

- ▶ L'option `-c num` de `cut` permet d'extraire le contenu d'une ligne en se basant sur la position d'un caractère et non celle d'un champ
- ▶ **Exemple:** Extraire le 5^{eme} caractère de chaque ligne du fichier `test02.txt`

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ cat test02.txt
192.168.1.10
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
168.100.1.14
168.100.1.06
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ cut -c 5 test02.txt
1
1
5
1
2
1
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec head

⑯ **head : filtre un fichier en affichant un nombre limité (n) des premières lignes**

- ▶ Par défaut cette commande affiche les 10 premières lignes
- ▶ Exemple:

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ head fich1.txt
1
2
3
4
5
6
7
8
9
10
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec head

- **Exemple1:** Afficher les 10 premières lignes du fichier *test02.txt*

```
abdessanad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ cat test02.txt
192.168.1.10
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
168.100.1.14
168.100.1.06
192.168.1.10
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
abdessanad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ head test02.txt
192.168.1.10
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
168.100.1.14
168.100.1.06
192.168.1.10
168.100.2.30
abdessanad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec head

Remarques:

- ▶ Avec **head** on peut afficher un nombre des premières lignes on donnant comme option: -n nombre ou -nombre
- ▶ **Exemple:** Affiche les 3 premières lignes du fichier *test02.txt*
⇒ **head -n 3 test02.txt**
ou bien
⇒ **head -3 test02.txt**

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ head -3 test02.txt
192.168.1.10
168.100.2.30
14.15.13.2
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ head -n 3 test02.txt
192.168.1.10
168.100.2.30
14.15.13.2
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec head

Attention:

- ▶ L'option -n -nombre demande à la commande **head** d'afficher toutes les lignes sauf les dernières lignes indiquées par nombre
- ▶ **Exemple:** Afficher toutes les lignes du fichier *fich1.txt* sauf les 3 dernières

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ head -n -3 fich1.txt
1
2
3
4
5
6
7
8
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec head

► Exemple général:

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ head -n 3 fich1.txt
1
2
3
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ head -3 fich1.txt
1
2
3
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ head -n -3 fich1.txt
1
2
3
4
5
6
7
8
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec head

- ▶ La commande **head** peut aussi faire un **affichage en se basant sur le nombre d'octets**
- ▶ L'**option -c** permet de faire cet affichage
- ▶ Cette option a la **même utilisation que** celle de l'option **-n**
- ▶ **Exemple:**

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ head -c 5 test02.txt
192.1abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ head -c -5 test02.txt
192.168.1.10
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
168.100.1.14
168.100.1.06
192.168.1.10
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.1abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec tail

17) **tail : filtre un fichier en affichant un nombre limité (n) des dernières lignes**

- ▶ Par défaut cette commande affiche les 10 dernières lignes comme **head**
- ▶ **Exemple1:**

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ tail fich1.txt
2
3
4
5
6
7
8
9
10
11
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec tail

► Exemple2:

```
abdessamad@abdessamad-HP-Pavillion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ cat test02.txt
192.168.1.10
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
168.100.1.14
168.100.1.06
192.168.1.10
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
abdessamad@abdessamad-HP-Pavillion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ tail test02.txt
168.200.13.2
124.135.12.02
168.100.1.14
168.100.1.06
192.168.1.10
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
abdessamad@abdessamad-HP-Pavillion-Laptop-15-eg0xxx:~/Desktop/SE1/test$
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec tail

- ▶ L'option -n nombre ou bien -nombre de **tail** permet d'afficher un nombre des dernières lignes
- ▶ **Exemple:** Affiche les 5 dernières lignes du fichier *test02.txt*

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ tail -5 test02.txt
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ tail -n 5 test02.txt
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec tail

Attention:

- L'utilisation d'une valeur positive(**+nbr**) avec l'option **-n** de la commande **tail** permet d'afficher les dernières lignes à partir de la ligne numéro nbr
- Exemple:

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ tail -n 3 fich1.txt
9
10
11
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ tail -n -3 fich1.txt
9
10
11
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ tail -n +3 fich1.txt
3
4
5
6
7
8
9
10
```

Les commandes de manipulation du contenu de fichier

12) Filtrer le contenu du fichier: Avec tail

- ▶ L'option `-c` de la commande `tail` permet de faire un affichage en se basant sur le nombre d'octets tout comme pour `head`
- ▶ Exemple:

```
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ tail -c 5 test02.txt
2.02
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ tail -c -5 test02.txt
2.02
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ tail -c +5 test02.txt
168.1.10
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
168.100.1.14
168.100.1.06
192.168.1.10
168.100.2.30
14.15.13.2
200.13.3.1
168.200.13.2
124.135.12.02
abdessamad@abdessamad-HP-Pavilion-Laptop-15-eg0xxx:~/Desktop/SE1/test$ █
```