

04 - LListar: ls

Curs 2020 - 2021

ASIX M01-ISO UF1-A01-02 ordres de fitxers

Llistar: ls	1
Descripció	1
L'ordre ls	1
Directoris (.) i (..)	3
Informació d'un llistat llarg	3
Exercicis d'exemple:	4

Llistar: ls

Descripció

L'ordre ls

L'ordre ls permet llistar el contingut dels directoris i mostrar el llistat de maneres diferents segons les opcions que indiquen. Pot llistar un, un conjunt de fitxers o un directori sencer.

Per indicar seleccions de fitxers s'utilitza Pathname Expansion que a través de metacaracters com per exemple * o ? permet generar plantilles de noms d'elements.

Atenció: en cap cas confonguem les expressions llistar el contingut d'un directori (veure la llista d'elements que hi ha, que conté un directori), amb l'expressió mostrar el contingut d'un fitxer (veure les dades, la xixa, que conté un fitxer).

- ❑ **Llistar**: llistar o mostrar el contingut d'un directori es fa amb l'ordre *ls*.
- ❑ **Mostrar el contingut d'un fitxer**: es fa amb l'ordre *cat*.

L'ordre conceptualment té el format bàsic:

ls [opcions] elem[...]

- l és l'opció per dir que faci un llistat llarg.
- a --all és l'opció per dir que faci un llistat mostrant tots els fitxers, incloent els fitxers ocults que són els que comencen per un punt.

elem és allò que volem llistar, pot ser un fitxer o un directori. Es poden indicar múltiples elements a llistar separats per espai. Cada element a llistar ha de contenir la seva ruta (absoluta o relativa).

Exemple de llistat simple:

```
$ ls /
bin dades etc lib lost+found mnt proc run srv tmp var
boot dev home lib64 media opt root sbin sys usr virtualbox
```

- Llista el directori arrel del sistema.
- És un llistat simple en que mostra el nom dels elements

Exemple de llistat **llarg**:

```
$ ls -l /
total 72
lrwxrwxrwx. 1 root root 7 3 ago 2017 bin -> usr/bin
dr-xr-xr-x. 6 root root 4096 20 abr 19:59 boot
drwxrwxrwx. 2 root root 4096 25 set 2019 dades
drwxr-xr-x 22 root root 4200 10 oct 09:24 dev
drwxr-xr-x. 150 root root 12288 10 oct 09:25 etc
drwxr-xr-x. 25 root root 4096 19 set 00:59 home
lrwxrwxrwx. 1 root root 7 3 ago 2017 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 3 ago 2017 lib64 -> usr/lib64
drwx----- 2 root root 16384 17 set 2019 lost+found
drwxr-xr-x. 2 root root 4096 3 ago 2017 media
drwxr-xr-x. 2 root root 4096 3 ago 2017 mnt
drwxr-xr-x. 6 root root 4096 20 jun 17:54 opt
dr-xr-xr-x 300 root root 0 10 oct 11:24 proc
dr-xr-x--- 9 root root 4096 9 oct 21:21 root
drwxr-xr-x 44 root root 1240 10 oct 10:24 run
lrwxrwxrwx. 1 root root 8 3 ago 2017 sbin -> usr/sbin
drwxr-xr-x. 2 root root 4096 3 ago 2017 srv
dr-xr-xr-x 13 root root 0 10 oct 09:24 sys
drwxrwxrwt 16 root root 340 10 oct 11:41 tmp
drwxr-xr-x. 12 root root 4096 17 set 2019 usr
drwxr-xr-x. 22 root root 4096 19 set 2019 var
```

- Observeu que en fer un llistat llarg es mostra per a cada element més informació, els seus atributs (tipus, permisos, propietari, etc).

Exemple de llistat all:

```
$ ls -la /mnt
total 8
drwxr-xr-x. 2 root root 4096 3 ago 2017 .
dr-xr-xr-x. 20 root root 4096 14 mar 2020 ..
```

- Observeu que podem combinar short options (les que van amb un guió al davant). En aquest cas fem un llistat llarg i all de tot el contingut de /mnt, que de fet no conté res.
- Podem observar que el directori /mnt és buit, únicament conté els dos elements punt (.) i punt-punt (..) que es generen sempre dins de tot sots directori.

Directoris (.) i (..)

Dins de tot directori hi ha sempre dos elements ocults anomenats punt (.) i punt-punt (..) que tenen un significat especial. Podeu observar que en un llistat normal no es mostren perquè estan ocults (tot allò que comença per un punt no es llista per defecte).

```
$ ls /opt/  
containerd google pt zoom  
  
$ ls -a /opt/  
. .. containerd google pt zoom
```

Significat d'aquests elements:

. (punt) Significa el propi directori.

.. (punt-punt) Significa el directori pare del directori.

Informació d'un llistat llarg

El fer un llistat llarg amb l'ordre `ls -l` obtenim per a cada element informació dels seus atributs, anem a veure quina és aquesta informació:

```
$ ls -l /etc/passwd  
-rw-r--r-- 1 root root 3550 19 set 00:59 /etc/passwd
```

-	rw-r--r--	1	root	root	3550	19 set 00:59	/etc/passwd
tipus	permisos	enllaços	propietari	grup	mida	data	nom

Tipus: el primer caràcter indica el tipus d'element, si és un regular-file (-), si és un directori (d), si és un fitxer de blocs (b), si és un dispositiu de caràcters (c), si és un pipe (p) i si és un socket (s).

Permisos: indiquen els permisos del propietari, grup o altres sobre l'element. Si hi ha permís de lectura, escriptura o execució. *Els permisos s'estudien més endavant.*

Enllaços: el número d'enllaços indica quants hard-links té el fitxer. *El concepte de links (enllaços) s'estudia més endavant.*

Propietari: Indica el nom del propietari de l'element, el nom de l'usuari a qui pertany l'element.

Grup: indica el nom del grup al que pertany l'element. Tot element pertany a un propietari i a un grup i no necessàriament el propietari ha de pertànyer al grup. *La propietat (usuaris i grups) s'estudia més endavant.*

Mida: La mida en bytes del fitxer (per fi una cosa que entenem!).

Data: La data de creació/modificació de l'element.

Nom: el nom de l'element. Els elements que comencen amb un caràcter punt (.) són ocults i no es llisten per defecte.

Exercicis d'exemple:

Primerament heu de fer l'exercici **00-Creació_de_estructura** per generar els directoris i els fitxers necessaris per fer aquest exercici.

1. Fer actiu el directori */tmp/mp1*.

Des d'aquest directori realitzarem tots els exercicis, tots. No és permès de canviar de directori amb l'ordre `cd`. Totes les ordres del sistema es poden realitzar des de qualsevol directori actiu.

Després de fer aquest directori actiu verificar-ho amb l'ordre `pwd`.

```
$ cd /tmp/m01/
```

```
$ pwd  
/tmp/m01
```

Recordeu quina és l'estructura de directoris i fitxers que estem utilitzant:

```
$ tree /tmp/m01  
/tmp/m01  
├── operatiu  
│   ├── apunts  
│   │   ├── carta.txt  
│   │   ├── dades.pdf  
│   │   ├── dossier.odt  
│   │   ├── informe.pdf  
│   │   ├── projecte.odt  
│   │   └── treball.txt  
│   └── exercicis  
└── xarxes  
  
$ tree /var/tmp/prova/  
/var/tmp/prova/
```

2. Llistar el directori *arrel* (/) fent un llistat simple

```
$ ls /  
bin dades etc lib lost+found mnt proc run srv tmp var  
boot dev home lib64 media opt root sbin sys usr
```

3. Llistar el directori *arrel* fent un llistat de tots els fitxers (*all*) usant la *long-option*.

```
$ ls --all /  
. .autorelabel boot dev home lib64 media opt root sbin sys usr  
.. bin dades etc lib lost+found mnt proc run srv tmp var
```

- Observem que ara apareixen elements que no apareixen en el llistat simple. Són els fitxers o directoris ocults, que comencen amb el caràcter punt (.).
- S'ha utilitzat una long-option per indicar que cal llistar tots els fitxers. Aquesta opció es pot indicar com a `-a` o com a `--all`.

4. Llistar el directori **arrel** (/) fent un llistat llarg i de tots els fitxers.

```
$ ls -la /
total 80
dr-xr-xr-x. 20 root root 4096 14 mar 2020 .
dr-xr-xr-x. 20 root root 4096 14 mar 2020 ..
-rw-r--r-- 1 root root 0 14 mar 2020 .autorelabel
lrwxrwxrwx. 1 root root 7 3 ago 2017 bin -> usr/bin
dr-xr-xr-x. 6 root root 4096 20 abr 19:59 boot
drwxrwxrwx. 2 root root 4096 25 set 2019 dades
drwxr-xr-x 22 root root 4200 10 oct 09:24 dev
drwxr-xr-x. 150 root root 12288 10 oct 09:25 etc
drwxr-xr-x. 25 root root 4096 19 set 00:59 home
lrwxrwxrwx. 1 root root 7 3 ago 2017 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 3 ago 2017 lib64 -> usr/lib64
drwx----- 2 root root 16384 17 set 2019 lost+found
drwxr-xr-x. 2 root root 4096 3 ago 2017 media
drwxr-xr-x. 2 root root 4096 3 ago 2017 mnt
drwxr-xr-x. 6 root root 4096 20 jun 17:54 opt
dr-xr-xr-x 301 root root 0 10 oct 11:24 proc
dr-xr-x--- 9 root root 4096 9 oct 21:21 root
drwxr-xr-x 44 root root 1240 10 oct 10:24 run
lrwxrwxrwx. 1 root root 8 3 ago 2017/sbin -> usr/sbin
drwxr-xr-x. 2 root root 4096 3 ago 2017 srv
dr-xr-xr-x 13 root root 0 10 oct 09:24 sys
drwxrwxrwt 16 root root 340 10 oct 12:33 tmp
drwxr-xr-x. 12 root root 4096 17 set 2019 usr
drwxr-xr-x. 22 root root 4096 19 set 2019 var
```

- Observeu que llista tots els elements (també els ocults) en format llarg.

5. Llistar el directori **/mnt** usant totes les opcions de **l·listat llarg** i de **tots** els fitxers

```
$ ls -la /mnt/
total 8
drwxr-xr-x. 2 root root 4096 3 ago 2017 .
dr-xr-xr-x. 20 root root 4096 14 mar 2020 ..
```

- Observeu que ajuntem les dues opcions **-l** i **-a** amb un sol **-la** que també podria ser **-al** ja que l'ordre dels arguments és indiferent.
- Observeu que en tot directori sempre hi ha dins els dos elements que signifiquen:
punt (.) el propi directori
punt-punt (..) el directori pare.

6. Llistar els fitxers **/etc/fstab**, **/etc/passwd** i el directori **/usr** tot fent un llistat llarg.

```
$ ls -l /etc/fstab /etc/passwd /usr/
-rw-r--r-- 1 root root 490 22 oct 2019 /etc/fstab
-rw-r--r-- 1 root root 3550 19 set 00:59 /etc/passwd

/usr:
total 288
dr-xr-xr-x. 2 root root 69632 3 oct 18:40 bin
drwxr-xr-x. 2 root root 4096 3 ago 2017 games
drwxr-xr-x. 57 root root 4096 26 mai 19:04 include
dr-xr-xr-x. 65 root root 20480 19 jun 13:08 lib
dr-xr-xr-x. 175 root root 135168 3 oct 18:40 lib64
drwxr-xr-x. 51 root root 12288 22 abr 09:36 libexec
drwxr-xr-x. 12 root root 4096 17 set 2019 local
dr-xr-xr-x. 2 root root 20480 27 set 11:48/sbin
drwxr-xr-x. 314 root root 12288 3 oct 18:40 share
```

```
drwxr-xr-x. 4 root root 4096 15 jul 23:17 src
lrwxrwxrwx. 1 root root 10 3 ago 2017 tmp -> ../var/tmp
```

- L'ordre ls admet cap, un o múltiples arguments. En aquest cas se li passen tres arguments que són dos fitxers i un directori.
- Cada argument ha de tenir la seva ruta ben construïda.

7. Llistar el *directori actiu* (llistat *llarg* i de *tot*).

```
$ ls -la
total 0
drwxrwxr-x 4 ecanet ecanet 80 10 oct 12:33 .
drwxrwxrwt 16 root root 340 10 oct 12:53 ..
drwxrwxr-x 4 ecanet ecanet 80 10 oct 12:33 operatiu
drwxrwxr-x 2 ecanet ecanet 40 10 oct 12:33 xarxes
```

- Observeu que no cal passar-li cap argument a ls i per tant interpreta que cal llistar el directori actiu.
- Observeu que tant operatiu com xarxes són directoris i per això la línia comença amb la *lletra d* indicativa de que element és un *directori*.

8. Llistat simple del directory *apunts*.

```
$ ls operatiu/apunts/
carta.txt dades.pdf dossier.odt informe.pdf projecte.odt treball.txt
```

- Observeu que per llistar apunts des del directori actiu utilitzem una ruta relativa.
- L'ordre es llegeix:
llistar el directori apunts que està dins del directori operatiu que està dins del directori actual.

9. Llistat llarg dels directoris *prova*, *xarxes* i *exercicis*.

```
$ ls -l /var/tmp/prova/ xarxes/ operatiu/exercicis/
operatiu/exercicis/:
total 0

/var/tmp/prova/:
total 0

xarxes/:
total 0
```

- Aquest és un altre exemple de passar múltiples arguments a l'ordre ls.
- Se li passen tres arguments i llista els tres directoris.
- En aquest cas els tres directoris estan buits.

10. Llistar el directori */boot* mostrant els *atributs* de cada entrada i mostrant la *mida* en *human-readable* (consulteu el man de l'ordre ls).

```
$ ls -lh /boot/
total 109M
drwxr-xr-x. 3 root root 4,0K 17 set 2019 06f1b134f5ac451a823ad828b1047db0
-rw-r--r--. 1 root root 194K 14 nov 2018 config-4.18.19-100.fc27.x86_64
drwxr-xr-x. 3 root root 4,0K 8 gen 2018 efi
drwxr-xr-x. 2 root root 4,0K 20 abr 19:59 extlinux
drwx-----. 6 root root 4,0K 27 abr 18:51 grub2
-rw-----. 1 root root 69M 17 set 2019 initramfs-0-rescue-06f1b134f5ac451a823ad828b1047db0.img
-rw-----. 1 root root 20M 17 set 2019 initramfs-4.18.19-100.fc27.x86_64.img
-rw-----. 1 root root 3,9M 14 nov 2018 System.map-4.18.19-100.fc27.x86_64
-rwxr-xr-x. 1 root root 8,3M 17 set 2019 vmlinuz-0-rescue-06f1b134f5ac451a823ad828b1047db0
-rwxr-xr-x. 1 root root 8,3M 14 nov 2018 vmlinuz-4.18.19-100.fc27.x86_64
```

- Observeu que el llistat de la mida de cada element ara no és en bytes sinó en una mesura fàcil per a la comprensió humana i que varia segons l'element, uns es mostren en K i altres en M.

11. Llistar /dev/sda* (*ordre avançada*).

```
$ ls -l /dev/sda*
brw-rw---- 1 root disk 8, 0 10 oct 09:24 /dev/sda
brw-rw---- 1 root disk 8, 1 10 oct 09:24 /dev/sda1
brw-rw---- 1 root disk 8, 2 10 oct 09:24 /dev/sda2
brw-rw---- 1 root disk 8, 3 10 oct 09:24 /dev/sda3
brw-rw---- 1 root disk 8, 4 10 oct 09:24 /dev/sda4
brw-rw---- 1 root disk 8, 5 10 oct 09:24 /dev/sda5
brw-rw---- 1 root disk 8, 6 10 oct 09:24 /dev/sda6
brw-rw---- 1 root disk 8, 7 10 oct 09:24 /dev/sda7
```

- Observeu que llista els dispositius de disc del sistema, el disc dur i les partitions.
- El primer caràcter dl llistat llarg de cada element és una *lletra b* indicant que es tracten de dispositius de *bloc* (és lògic són les particions i el disc dur).

12. Llistar /dev/tty* (*ordre avançada*).

```
$ ls -l /dev/tty* | head -n 10
crw-rw-rw- 1 root tty 5, 0 10 oct 09:24 /dev/tty
crw--w---- 1 root tty 4, 0 10 oct 09:24 /dev/tty0
crw--w---- 1 gdm tty 4, 1 10 oct 09:24 /dev/tty1
crw--w---- 1 root tty 4, 10 10 oct 09:24 /dev/tty10
crw--w---- 1 root tty 4, 11 10 oct 09:24 /dev/tty11
crw--w---- 1 root tty 4, 12 10 oct 09:24 /dev/tty12
crw--w---- 1 root tty 4, 13 10 oct 09:24 /dev/tty13
crw--w---- 1 root tty 4, 14 10 oct 09:24 /dev/tty14
crw--w---- 1 root tty 4, 15 10 oct 09:24 /dev/tty15
crw--w---- 1 root tty 4, 16 10 oct 09:24 /dev/tty16
```

- Observeu que llista tots els dispositius de terminals del sistema (les consoles).
- Podem observar que cada línia de dispositiu comença amb la *lletra c* que indica que es tracta d'un dispositiu de *caràcter*.

13. Que passa en fer la següent ordre?

```
$ ls -l /usr/bin/ | head
total 2050412
```

```
-rwxr-xr-x. 1 root root 53264 20 abr 2018 [  
-rwxr-xr-x. 1 root root 40 6 feb 2018 7z  
-rwxr-xr-x. 1 root root 41 6 feb 2018 7za  
-rwxr-xr-x. 1 root root 52760 20 jul 2018 ab  
-rwxr-xr-x. 1 root root 11280 26 feb 2018 abrt-action-analyze-backtrace  
...
```

- Mostra els fitxers executables, les ordres del sistema, del directori /usr/bin.
- Com que n'hi ha moltes fa un llistat molt llarg que no podem seguir de tant ràpid que va.

Podem fer la mateixa ordre mostrant només els 10 primers elements (*ordre avançada*).

```
$ ls -l /usr/bin/ | head  
total 2050412  
-rwxr-xr-x. 1 root root 53264 20 abr 2018 [  
-rwxr-xr-x. 1 root root 40 6 feb 2018 7z  
-rwxr-xr-x. 1 root root 41 6 feb 2018 7za  
-rwxr-xr-x. 1 root root 52760 20 jul 2018 ab  
-rwxr-xr-x. 1 root root 11280 26 feb 2018 abrt-action-analyze-backtrace  
-rwxr-xr-x. 1 root root 15408 26 feb 2018 abrt-action-analyze-c  
-rwxr-xr-x. 1 root root 1269 26 feb 2018 abrt-action-analyze-ccpp-local  
-rwxr-xr-x. 1 root root 6894 26 feb 2018 abrt-action-analyze-core  
-rwxr-xr-x. 1 root root 15376 25 set 2017 abrt-action-analyze-java
```

Podem fer la mateixa ordre mostrant els n (per exemple 14) primers elements (*ordre avançada*).

```
$ ls -l /usr/bin/ | head -n 14  
total 2050412  
-rwxr-xr-x. 1 root root 53264 20 abr 2018 [  
-rwxr-xr-x. 1 root root 40 6 feb 2018 7z  
-rwxr-xr-x. 1 root root 41 6 feb 2018 7za  
-rwxr-xr-x. 1 root root 52760 20 jul 2018 ab  
-rwxr-xr-x. 1 root root 11280 26 feb 2018 abrt-action-analyze-backtrace  
-rwxr-xr-x. 1 root root 15408 26 feb 2018 abrt-action-analyze-c  
-rwxr-xr-x. 1 root root 1269 26 feb 2018 abrt-action-analyze-ccpp-local  
-rwxr-xr-x. 1 root root 6894 26 feb 2018 abrt-action-analyze-core  
-rwxr-xr-x. 1 root root 15376 25 set 2017 abrt-action-analyze-java  
-rwxr-xr-x. 1 root root 11256 26 feb 2018 abrt-action-analyze-oops  
-rwxr-xr-x. 1 root root 11264 26 feb 2018 abrt-action-analyze-python  
-rwxr-xr-x. 1 root root 2818 26 feb 2018 abrt-action-analyze-vmcore  
-rwxr-xr-x. 1 root root 1383 26 feb 2018 abrt-action-analyze-vulnerability
```

Podem fer la mateixa ordre mostrant tots els elements fent pauses entre pàgines. Els mostrarà pàgina a pàgina (*ordre avançada*).

- ❑ Per sortir del llistat prémer la *lletra q* de *quit*.
- ❑ Per avançar pàgina a pàgina prémer l'*espai* o les tecles *pgup* i *pgdwn*.
- ❑ Per anar al principi o final prémer la *lletra g* o la *lletra G*. O bé prémer les tecles *inici* o *final*.

```
$ ls -l /usr/bin/ | less
```

14. Comptar quants elements hi ha en llistar /usr/bin (*ordre avançada*).

```
$ ls -l /usr/bin/ | wc -l
```


- Amb el filtre *wc word count* i l'opció de comptar només le slínies podem fer un llistat llarg i comptar quantes línies ha llistat.