

Sistemi operativi



Corso di Laurea Triennale in Ingegneria Informatica

Lezione 1

- Unix - storia e filosofia
- Debian GNU/Linux
- Primi passi su Linux

Informazioni sul corso

➤ Fabio Del Vigna:

- ricevimento su appuntamento (email)
 - ⇒ fabio.delvigna@for.unipi.it
 - ⇒ specificare come oggetto “SistemiOperativi”
- http://for.unipi.it/fabio_del_vigna/
- Esame
 - esame orale (parte pratica) relativa alle esercitazioni in laboratorio
 - esame orale (parte teorica),
 - iscrizione obbligatoria attraverso il sistema di prenotazione esami.
- Acknowledgement: materiale didattico rielaborato a partire dai lucidi di Daniel Cesarini

Obiettivi dei laboratori di S.O.



- **Basi per l'utilizzo e l'amministrazione di sistemi Unix (con riferimento a Debian GNU/Linux. Interagire e configurare un sistema informatico utilizzando la CLI (Command Line Interface).**
- **Programmazione in ambiente concorrente (processi, libreria pthread), sincronizzazione e comunicazione.**

Contenuti del corso 1/3

- **Storia di UNIX**
- **Comandi base, redirectione e pipeline**
- **Editor, utenti e gruppi**
- **Gestione permessi, comando “su”**
- **Shell (interazione e scripting)**
- **Filesystem, Virtual FileSystem, configurazione**
- **Gestione processi, priorità, segnali**
- **Pianificazione di esecuzione (cron)**

Contenuti del corso 2/3

- **Archiviazione e compressione (tar, gzip)**
- **Ricerca di file (find, locate) e in file (grep)**
- **Scripting (Bash)**
- **Interazione tra processi (segnali, ...)**
- **Processi e thread (basi e sincronizzazione)**
- **Unix I/O**

Contenuti del corso 3/3

Extra:

- **Strumenti di ausilio allo sviluppo di codice (Make, diff, patch, svn)**
- **Configurazione e installazione di Debian**
- **Seminario/ programmazione avanzata con i thread C++ 14**
- **Varie ed eventuali**

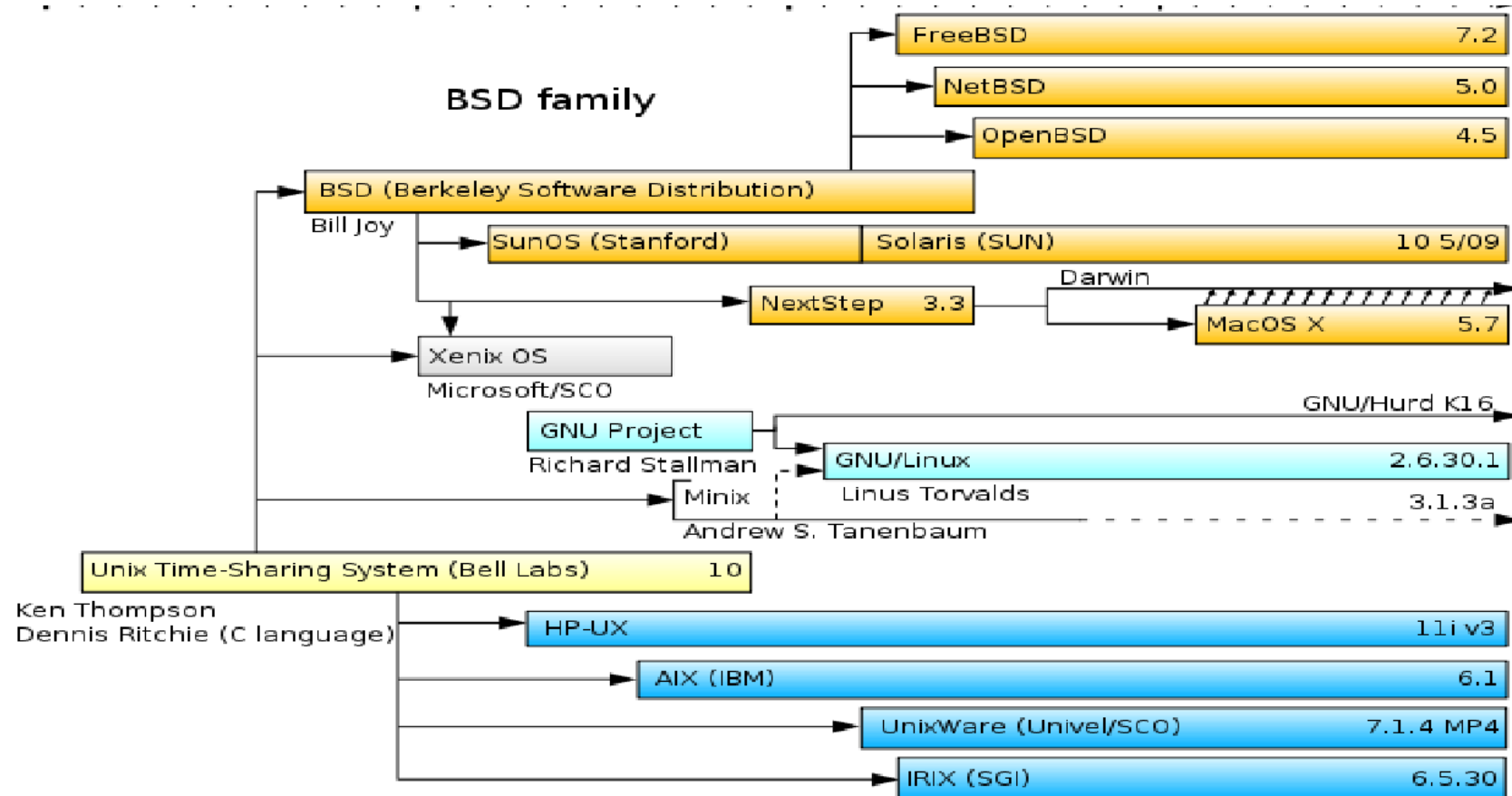
Lezione 1

- Introduzione a Unix.
- Filesystem.
- Elementi fondamentali:
 - Shell
 - Comandi di base
 - ...
- Altri comandi.

Un po' di storia

- 1969 - Prima edizione di UNIX sviluppata da Ken Thompson e Dennis Ritchie presso i Bell Laboratories.
- 1973 - Thompson e Ritchie riscrivono UNIX in C
- Derivati di Unix:
 - BSD (Berkeley Software Distribution)
 - GNU/Linux
 - ...

Genealogia di Unix





Progetto GNU, FSF,
GPL

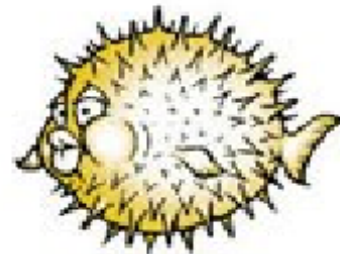
- **1983**, Stallman lancia il progetto GNU con lo scopo di sviluppare un sistema operativo Unix-like libero.
- **1985**, Stallman fonda la Free Software Foundation (FSF), che promuove lo sviluppo di software libero ed, in particolare, del progetto GNU.
- **1989**, Stallman e Moglen scrivono la versione 1.0 della GPL (o GNU GPL, GNU General Public License).
- La GPL è una licenza per software libero.



- GNU/Linux è un sistema operativo Unix-like costituito dall'integrazione del kernel Linux con elementi del sistema GNU.
- Prima versione rilasciata con licenza GPL.
- (es. Debian, Ubuntu, ...)

GNU/Linux

... e poi...



debian



ubuntu

GNU/Linux - LIBERO e GRATUITO

➤ **LIBERO:**

È sviluppato e rilasciato sotto la licenza GNU/GPL (General Public License). Il codice sorgente è liberamente disponibile e modificabile.

➤ **GRATUITO:**

(Quasi) tutte le distribuzioni sono scaricabili gratuitamente da internet ed è possibile trovare CD con varie versioni di Linux in edicola, allegati a giornali del settore. Il tutto senza spendere 1 €!

Quanti “tipi” di Linux esistono?

- Centinaia! Vengono chiamate anche **distro**, solitamente create da **comunità** di sviluppatori o società, che scelgono, preparano e compilano i pacchetti da includere. Tutte le distribuzioni condividono il **kernel Linux** ma si differenziano tra loro per il cosiddetto "**parco software**", cioè i pacchetti preparati e selezionati dagli sviluppatori per la distribuzione stessa, per il sistema di gestione del software e per i servizi di assistenza e manutenzione offerti

Debian GNU/Linux

- Contratto sociale
- Filosofia
- Software libero
- Nome: Debora Ian ...



http://www.debian.org



The screenshot shows the Debian website homepage. At the top, there is a navigation bar with links for "Informazioni su Debian", "Ottener Debian", "Supporto", and "Angolo degli sviluppatori". A search box is located on the right. Below the navigation bar is a large banner with the Debian logo and the text "THE UNIVERSAL OPERATING SYSTEM". A button for "Scarica Debian6.0 (32-bit PC Netinst Installer)" is visible in the top right of the banner. The main content area starts with a paragraph defining Debian as a free operating system. Below this, there are several columns of links for "Informazioni", "Ottener Debian", "Notizie", "Supporto", and "Varie". A section titled "Per cominciare" follows, containing a list of instructions for new users. The footer of the page is partially visible at the bottom.

debian /

Informazioni su Debian Ottener Debian Supporto Angolo degli sviluppatori

Scarica Debian6.0
(32-bit PC Netinst Installer)

debian
THE UNIVERSAL OPERATING SYSTEM

Debian è un sistema operativo (OS) [libero](#) per il tuo computer. Un sistema operativo è l'insieme dei programmi di base ed utilità che fanno funzionare il tuo computer.

Debian fornisce più di un semplice OS: viene distribuito con oltre 29000 [pacchetti](#), programmi già compilati e impacchettati in modo tale da permettere installazioni facili sulla tua macchina. [Per saperne di più...](#)

Informazioni

- » Il nostro contratto sociale
- » Software Libero
- » Partner
- » Donazioni
- » Contatti

Aiutare Debian

Ottener Debian

- » Rivenditori di CD
- » Immagini ISO per CD
- » Installazione via rete
- » Preinstallato

Angolo degli sviluppatori

Pacchetti Debian

Notizie

- » Notizie dal progetto
- » Eventi

Documentazione

- » Informazioni sulle Versioni
- » Manuale d'installazione
- » Libri su Debian

Supporto

- » Debian diventa Internazionale
- » Informazioni sulla sicurezza
- » Segnalazioni di bug
- » Liste di messaggi
- » Archivi delle liste di messaggi
- » Port/Architetture

Varie

Mapa del sito

Cerca

IDENTICA

PLANET

L'[ultima versione stabile di Debian](#) è 6.0. L'ultimo aggiornamento di tale versione è stato rilasciato il 12 Maggio 2012. Si possono leggere maggiori informazioni [sulle versioni di Debian disponibili](#).

Per cominciare

- Se si vuole cominciare a utilizzare Debian, se ne può [ottenere una copia](#) e poi leggere il [manuale d'installazione](#) per installarla.
- Se si sta aggiornando all'ultima versione stabile da un versione precedente, si leggano le [note di rilascio](#) prima di procedere.
- Per ottenere aiuto nell'installazione o nell'uso di Debian si vedano le nostre pagine sulla [documentazione](#) e sul [supporto](#).
- Gli utenti che parlano lingue diverse da quella inglese dovrebbero controllare anche la sezione [internazionale](#).
- Gli utenti che utilizzano piattaforme diverse da quella Intel x86 dovrebbero controllare anche la sezione [port](#).

Caratteristiche di Unix

- **Multitasking:** è possibile la contemporanea esecuzione di più processi a divisione di tempo.
- **Multiutente:** più utenti (con diversi privilegi) possono interagire contemporaneamente.
- **Portabilità:** grazie all'utilizzo del linguaggio C nella realizzazione del sistema.
- **Modularità:** suddivisione in moduli.

Struttura di Unix

- 2 componenti principali:
 - kernel: il cui scopo è interagire con l'hardware,
 - applicazioni: che si rivolgono al nucleo per ottenere i servizi richiesti dalle loro funzioni:
 - ⇒ interpreti dei comandi (shell)
 - ⇒ programmi di sistema (strumenti messi a disposizione dal s.o.)
 - ⇒ programmi utente.

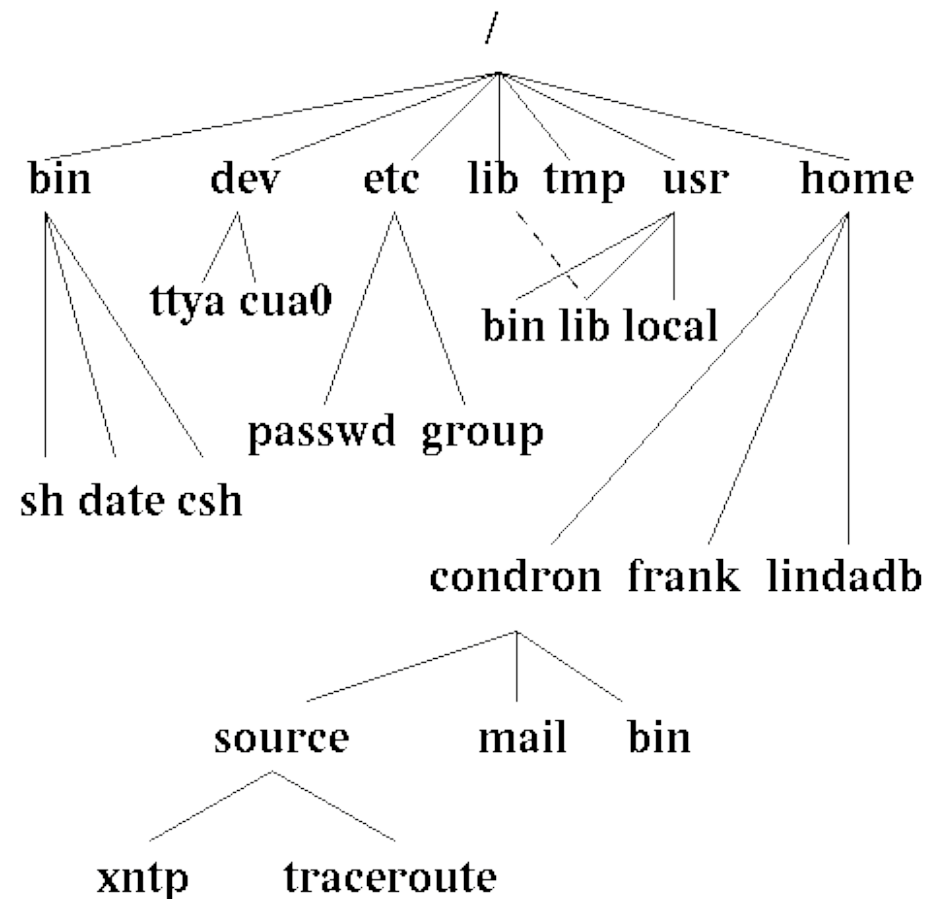
Linux invece è solo un kernel. Tutto il resto viene aggiunto dalla distribuzione.

Filesystem – File e directory

- ***File***: sequenza di 0 o più byte contenenti informazioni il cui significato è stabilito dal proprietario del file.
- ***Directory***: specifica entità del *filesystem* che elenca altre entità, tipicamente *file* e altre *directory*.

File system - Struttura

- **Struttura ad albero.**
- **Ogni nodo è un file o una directory di file.**
- **Le directory, a livello di sistema, sono trattate come dei file:**
 - per il sistema sono tutti i-node



File System - Path

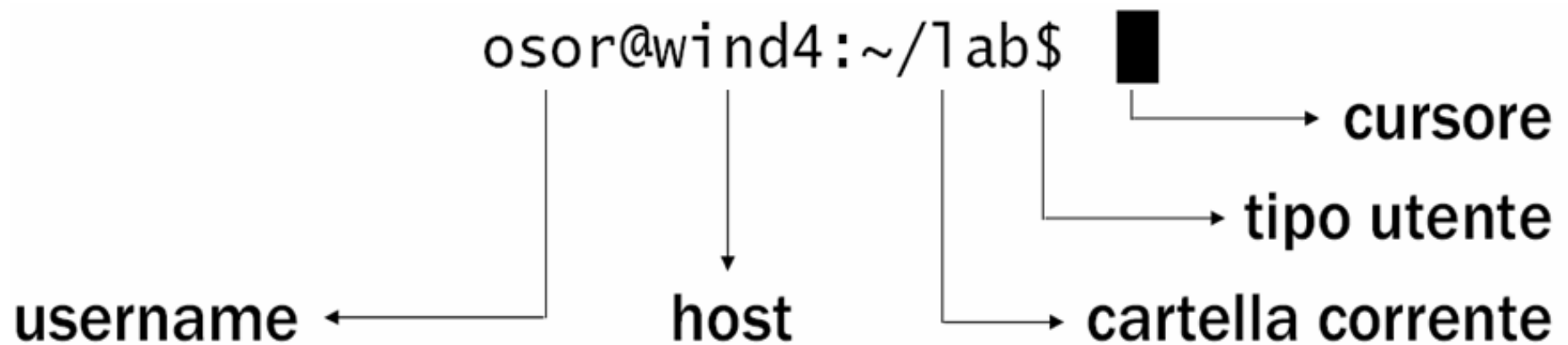
- ***Percorso assoluto:*** si esprime l'intero percorso partendo da root (/)
- ***Percorso relativo:*** si esprime partendo dalla directory di lavoro
 - . directory corrente
 - .. directory padre

File System – Cartelle predefinite

- **Sottocartelle della cartella principale / (*root*):**
- ***/bin* programmi eseguibili dall'utente**
- ***/dev* file dei device**
- ***/etc* file di configurazione del sistema**
- ***/lib* librerie condivise**
- ***/tmp* file temporanei**
- ***/usr* comandi e strumenti dell'utente**
- ***/root* home dell'utente root**
- ***/home* cartelle personali degli utenti**

- **La shell è un interprete di comandi che ripetutamente:**
 - stampa un prompt
 - legge un comando scritto dall'utente e terminato con enter/return/invio
 - esegue il comando o segnala un errore se non è in grado di completarlo
- **Esistono varie shell (sh, bash, tcsh, ...)**

Shell – Interprete dei comandi 1/2



- UNIX e Linux sono case-sensitive
- Più comandi possono essere messi in una stessa riga separati da ;

Shell – Interprete dei comandi 2/2

- **Funzioni**
 - autocompletamento (tasto TAB)
 - history (freccia SU/GIU)
- **Terminali Virtuali:**
 - Alt + F1 (F2,F3,...) : passaggio ad altri terminali.

- **Utente** `root`
 - amministratore del sistema,
 - può compiere qualsiasi tipo di operazione.
- **Utente normale**
 - utilizzatore del sistema,
 - ha dei privilegi limitati.

Accesso al sistema

- **Login:**
 - *login:* nome_utente
 - *password:* password
- **Disconnessione:**
 - `logout` oppure **Ctrl-D**

Shutdown

➤ I sistemi UNIX permettono solo all'utente root di avviare la procedura di arresto del sistema con i comandi seguenti:

➤ # shutdown -h now

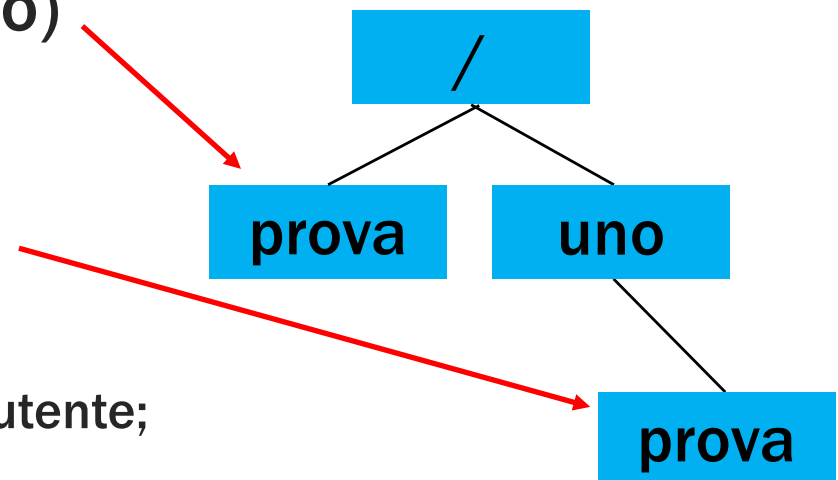
➤ # shutdown -r now (riavvio finale).

Comando cd

- **cd (change directory) consente di passare da una directory ad un'altra.**

Esempio dove sia uno la directory corrente, se si digita:

- **\$ cd /prova (path assoluto)**
⇒ **si passa alla directory prova;**
- **\$ cd prova (path relativo)**
⇒ **si passa alla directory prova;**
- **\$ cd ~ o cd**
⇒ **Si passa alla directory home dell'utente;**
- **\$ cd ..**
⇒ **Si passa alla directory padre che, in tal caso, è /.**



Comando pwd

- `pwd` (print working directory) visualizza il percorso **assoluto** della directory corrente.
- Esempio:

```
$ cd /bin
```

```
$ pwd
```

```
/bin
```

Comando ls 1/3

ls (list segments) visualizza i nomi di file o il contenuto delle directory indicate.

Esempio:

- per vedere il contenuto della directory / (1):

```
$ cd /
```

```
$ ls
```

```
bin dev home mnt tmp boot etc lib  
root var
```

- per vedere il contenuto della directory / (2):

```
$ ls /
```

```
bin dev home mnt tmp boot etc lib  
root var
```

Comando ls 2/3

ls [opzioni] file/dir1 file/dir2...]

- [file/dir1 file/dir2...] : specificano i file e/o le directory da elencare, in loro assenza è elencato il contenuto della directory corrente.
- [opzioni] : definiscono le opzioni con cui eseguire il comando `ls`. Tali opzioni vengono specificate dopo il simbolo `-`.

Comando ls 3/3

Opzioni principali:

- `$ ls -l`
 - Visualizza, oltre ai nomi, altre informazioni quali ad esempio modalità di accesso, proprietario del file,...
- `$ ls -a`
 - vengono elencati anche gli elementi i cui nomi iniziano con punto (i cosiddetti file nascosti).
- Le opzioni sono “cumulabili”, ad esempio è possibile scrivere:
- `$ ls -la`

Wildcard

- Insiemi di file possono essere riferiti usando i metacaratteri (wildcard).
- *: sostituisce zero o più caratteri.
- ?: sostituisce un carattere singolo.
- [a,b,c] o [a-z]: sostituisce un carattere nell'insieme.
- Esempio:
- `ls *.c`
 - Elenca i file che terminano con .c

Comandi di help – man 1/2

- Documentazione consultabile tramite il comando `man` (manual).
- Le pagine di tale documentazione sono raggruppate in sezioni omogenee in base al tipo di argomento trattato, ad esempio:
 - Sezione 1: comandi generali
 - Sezione 2: chiamate di sistema
 - Sezione 3: ??? fate **`man man`**

Comandi di help – man 2/2

La sintassi generale è la seguente:

```
man nome_comando
```

per accedere alle pagine man di una sezione del manuale che non sia quella predefinita, è necessario specificare la sezione alla quale si è interessati.

Esempio:

- `$ man 1 printf`
- `$ man 3 printf`

Comandi di help - Apropos e whatis

Altri due comandi di help disponibili in Unix sono:

- `apropos`: ricerca per parola chiave (opera in modo analogo al comando `man -k`).
- `whatis`: fornisce la riga sommario relativa ad un comando.

Comandi sui file - mkdir, rmdir, cp, mv

- `mkdir [opzioni] directory...: crea una directory (vuota).`
- `rmdir[opzioni] directory...: elimina le directory specificate solo se sono vuote.`
- `cp[opzioni] origine destinazione: copia file o directory in un'unica destinazione.`
- `mv[opzioni] origine destinazione: sposta i file e le directory.`

Comandi sui file - touch, cat, rm, ln

- `touch [opzioni] file...` : cambia data e ora di accesso e di aggiornamento dei file. Se si specificano file che non esistono, questi vengono creati vuoti.
- `cat [opzioni] [file...]` : concatena dei file e ne emette il contenuto attraverso lo standard output.
- `rm [opzioni] file...` : Rimuove i file indicati come argomento. In mancanza dell'indicazione delle opzioni necessarie, non vengono rimosse le directory.
- `ln [opzioni] sorgente destinazione` : crea collegamenti fisici o simbolici a file e directory.

Lettura file - less e more

➤ `less [opzioni] [file]...`

➤ `more [opzioni] [file]...`

Utilizzati per leggere file di puro testo.

"less > more"

Lettura file - tail, head

- `tail [opzioni] [file]...` :
mostra le ultime linee di dati
provenienti da uno o più file di testo.
- `head [opzioni] [file]...` :
mostra le prime linee di dati provenienti da uno o
più file di testo.

Redirezione e pipeline

La redirezione reindirizza i dati in modo da destinarli ad un file o da prelevarli da un file (uso di `<`, `2>`, `>`, `>>`).

- Redirezione dell'input (`<`)

- `$ cat < elenco`

- ⇒ Si ottiene in questo modo la visualizzazione del contenuto del file elenco.

- Redirezione dello standard error (`2>`)

- `$ ll 2> elenco`

- ⇒ Salva lo standard error nel file elenco.

Redirezione e pipeline

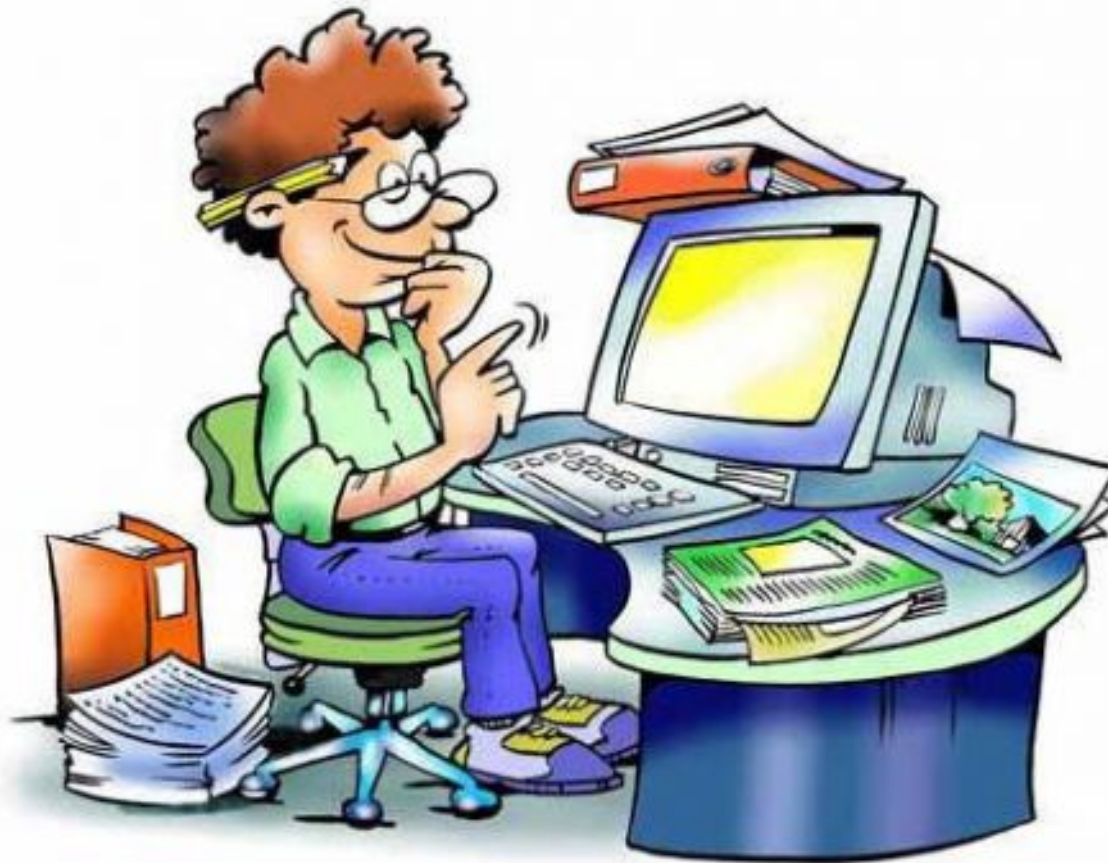
- Redirezione dell'output (>, >>)
 - `$ ls -l > elenco`
 - ⇒ Questo comando genera il file elenco con il risultato dell'esecuzione di `ls`.
 - Genera un nuovo file ogni volta, eventualmente sovrascrivendo ciò che esiste già con lo stesso nome.
 - La redirezione dell'output può essere fatta in aggiunta, aggiungendo i dati ad un file esistente con l'uso di >>:
 - `$ ls -l /tmp >> elenco`
 - ⇒ In tal modo viene aggiunto al file elenco l'elenco dettagliato del contenuto della directory `/tmp`.

Redirezione e pipeline

La pipeline è una forma di redirezione in cui la shell invia l'output di un comando come input del successivo.

- `$ cat elenco | sort`
 - In questo modo, `cat` legge il contenuto del file `elenco` che, invece di essere visualizzato sullo schermo, viene inviato dalla shell come input di `sort` che lo riordina e poi lo emette sullo schermo.

Esercizi





Esercizio 1

- 1) Aprite il terminale virtuale 1 ed eseguite l'accesso.
- 2) Aprite il terminale virtuale 2 ed eseguite l'accesso (Per il resto dell'esercitazione utilizzate il primo terminale per consultare il manuale ed il secondo per eseguire i comandi).
- 3) Create una directory "Esercitazione1".
- 4) Create un file "esercitazione.txt" all'interno di "Esercitazione1" che contenga la scritta "Esercizio" (non dovete usare un editor testuale per creare il file).
- 5) Visualizzate il contenuto del file "esercitazione.txt" usando il comando `more` (Passate a `more` prima il path relativo e poi il path assoluto del file).
- 6) Spostatevi in "Esercitazione1" e subito dopo usate il comando per tornare nella vostra home.



Esercizio 2

- 1) Visualizzate il percorso della directory corrente.
- 2) Spostatevi in “Esercitazione1” e create 3 file “f1.txt”, “f2.txt”, “f3.txt” contenenti rispettivamente la scritta “Uno”, “Due”, “Tre”.
- 3) Con un solo comando create il file “f_tot.txt” partendo da “f1.txt”, “f2.txt”, “f3.txt” fatto come segue:

Uno (a capo) Due (a capo) Tre
e visualizzatene il contenuto.

- 4) Cancellate i file “f_tot.txt”, “f1.txt”, “f2.txt”, “f3.txt”.
- 5) Adesso create il file “fcitta.txt” fatto come segue:

Milano (a capo) Perugia (a capo) Asti

- 6) Visualizzate a video il contenuto di “fcitta.txt” ma in ordine alfabetico.
- 7) Adesso salvate il contenuto di “fcitta.txt” ordinato in un file “fcittaord.txt”.



Esercizio 3

- 1) Visualizzate il contenuto di “fcittaord.txt” con il comando less. Usate la funzione di completamento automatico della shell per passare il file a less. Fino a che punto riesce ad aiutarvi?
- 2) Create un file “fcitta.c” e due cartelle “Testi” e “Sorgenti”.
- 3) Usando i metacaratteri copiate in “Testi” tutti i file .txt ed in “Sorgenti” i file .c .
- 4) Cancellate tutti i file di testo della directory “Esercitazione1”.
- 5) Create 3 file chiamandoli “fa.txt”, “fb.txt”, “fc.txt”.
- 6) Usate un’ espressione che permetta di spostare solo “fa.txt” ed “fc.txt” e non “fb.txt” nella cartella “Testi”.
- 7) Eliminate “fc.txt”.



Esercizio 4

- 1) Cancellate i file della cartella “Sorgenti”.
- 2) Usando rmdir eliminate le cartelle “Testi” e “Sorgenti”. Ci riuscite? Come fareste?
- 3) Create una cartella “sotto” e dentro “sotto” una cartella “sotto1”. Usate il man per trovare l’opzione di rmdir che permette di cancellare anche “sotto” se cancellate “sotto1”.
- 4) Create una cartella “origine” e dentro “origine” create la cartella “sotto_origine” ed il file “pippo.txt”.
- 5) Create una directory “destinazione” e copiate al suo interno il contenuto di “origine”. Se usate il cp senza opzioni cosa vi viene detto? Come dovete fare?
- 6) Adesso copiate non solo il contenuto ma tutta la cartella “origine” in “destinazione”.



Esercizio 5

- 1) Visualizzate il contenuto di “destinazione”.
- 2) Adesso usate l’opzione di ls che visualizza anche i permessi.
- 3) All’interno di “destinazione” create il file “.vedimi”.
- 4) Riuscite a visionarlo con un semplice ls? Come fareste per vederlo?
- 5) Salvate l’output di “ls /etc” in un file “ls_output.txt”.
- 6) Visualizzate solo la parte iniziale del file.
- 7) Adesso solo la parte finale.
- 8) Ora solo la prima riga e poi le ultime 2 righe del file.
- 9) Con un solo comando salvate sul file “terza.txt” solo la terza riga del file.