

Programmazione in Python

Intro sistema operativo e interfaccia utente

Dario Pescini - Mirko Cesarini

Cos'è un computer

- elaboratore elettronico digitale
- macchina che immagazzina e trasforma dati
- è composto da componenti elettronici
- sfrutta segnali digitali $\{0, 1\}$

Hardware e Software

Hardware componenti **fisici** del calcolatore e dei sistemi informatici

- case, tastiera, monitor, processore, etc.

Software componenti **astratte** del calcolatore necessarie a trasformare l'informazione

- programmi: sistema operativo, videogioco, etc.

É possibile distinguerli in

Componenti fisici: elettronici ed elettromeccanici

Insieme dei programmi che gestiscono il comportamento della componente hardware

- software di base (sistema operativo, ...)
- software applicativo ("Programmi")

L'architettura di un calcolatore: Von Neumann

Astrazione di un calcolatore che è la base dei calcolatori moderni.

Macchina di Von Neumann

Ha quattro tipologie di componenti

- Central Processing Unit (CPU)
- Memoria centrale
- interfacce di ingresso e uscita
- bus

Macchina di Von Neumann

CPU

- esegue le istruzioni per l'elaborazione dei dati
- svolge funzione di controllo

Memoria centrale

memorizza e fornisce l'accesso ai dati e ai programmi

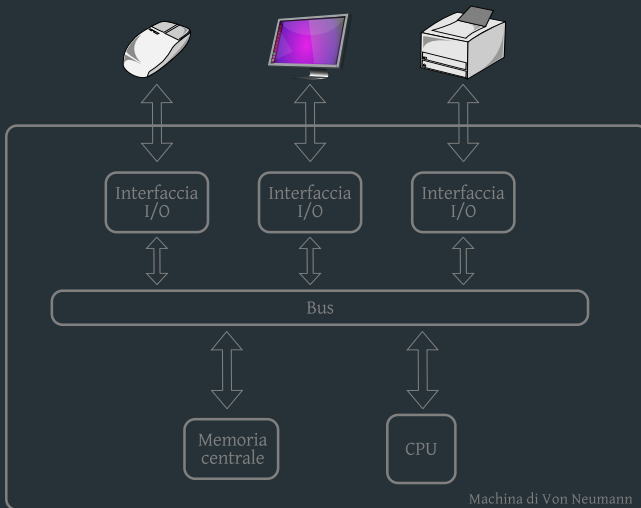
Interfacce di I/O

componenti di collegamento alle periferiche del calcolatore

Bus

si occupa del trasferimento dei dati e dei segnali di controllo tra le componenti del calcolatore

[illegible]

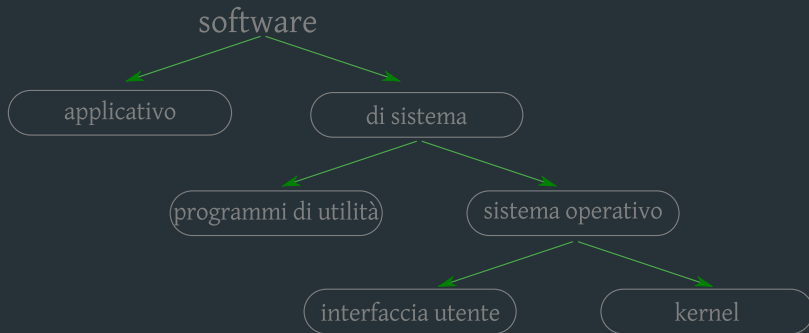


Macchina di Von Neumann

Innovazioni:

- Il codice del programma viene immagazzinato nella stessa memoria dei dati
- è la CPU che interpreta l'informazione e stabilisce se sia un dato o un'istruzione

Software: architettura



Software: architettura

applicativo

Insieme di programmi dedicati a compiti specifici.

- fogli di calcolo
- database
- CAD
- videogiochi
- ...

di sistema

Insieme di programmi dedicati a svolgere compiti comuni a tutti i sistemi di elaborazione.

- interazione con le periferiche
- accesso ai dati
- comunicazione
- ...

Software: architettura

software di utilità

Insieme dei programmi di sistema dedicati all'esecuzione delle attività fondamentali di gestione dell'elaboratore.

- formattazione supporti
- copia dei file
- stampa di file
- sincronizzazione orologio
- ...

sistema operativo

Insieme di programmi necessari alle attività base dell'elaboratore.

Software: architettura

software di utilità

Insieme dei programmi di sistema dedicati all'esecuzione delle attività fondamentali di gestione dell'elaboratore.

- formattazione supporti
- copia dei file
- stampa di file
- sincronizzazione orologio
- ...

software di sistema non incluso nel sistema operativo

sistema operativo

Insieme di programmi necessari alle attività base dell'elaboratore.

Sistema operativo

Sistema operativo

Programma che **controlla** l'esecuzione di programmi applicativi e agisce come **interfaccia** tra le applicazioni e l'hardware del calcolatore.

Obiettivi:

- indipendenza dall'hardware
- uso efficiente delle risorse del calcolatore
- semplificazione dell'uso dell'hardware di un calcolatore

Sistema operativo



Sistema operativo: componenti

interfaccia utente

Parte del sistema operativo dedicata all'**interazione** con l'utente.

Elemento necessario perché l'utente possa richiedere lo svolgimento di specifiche attività.

kernel

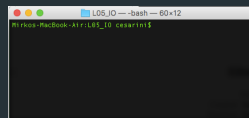
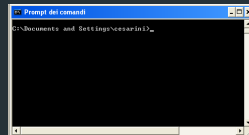
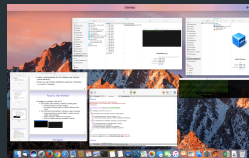
Nocciolo del sistema operativo.

Insieme di programmi preposti a svolgere le **operazioni base** per il funzionamento dell'elaboratore.

Sistema operativo: interfaccia utente

Paradigmi a confronto: GUI vs CLI

- **GUI** (Graphic User Interface).
 - Interazione basata su Finestre, Pulsanti, ...
 - Più recente (inventata negli anni 1980 nei laboratori **PARC della Xerox**)
- **CLI** (Command Line Interface).
 - Interazione basata su comandi digitati a tastiera (comandi singoli, script complessi)
 - Output attraverso: schermo, file, stampante
 - Utile per svolgere operazioni complesse e ripetitive



Sistema operativo: interfaccia utente

- **shell** la comunicazione utente-kernel avviene per mezzo di messaggi testuali sia in input che output, sfruttando tastiera, mouse e monitor.

Unix: Bourne shell, C shell, Korn shell

Windows: shell DOS cmd.exe, power shell

- **GUI** (graphical user interface) Evoluzione della shell, permette l'interazione con il kernel tramite rappresentazioni grafiche (icone) degli oggetti da manipolare.

Unix: KDE, gnome, Aqua

Windows: Windows

Una componente fondamentale è il window manager.

Unix: X11, XQuartz

Windows: Desktop Window Manager

Sistema operativo: Kernel

memory manager Ha il compito di gestire e coordinare l'impiego della memoria principale del computer.

process manager Ha il compito di coordinare l'esecuzione dei processi.

file manager Ha il compito di coordinare l'uso e le funzionalità delle memorie di massa

driver periferiche Programmi preposti all'interazione con i dispositivi collegati all'elaboratore

Sistema operativo: Kernel

memory manager

- tenere traccia delle parti di memoria assegnate ai processi
- decidere quali processi caricare quando la memoria è disponibile
- allocare e deallocare lo spazio di memoria quando necessario

In ambiente multi-processo deve garantire che ogni processo abbia accesso solo alle risorse a lui competenti.

Deve gestire i casi in cui la richiesta di risorse ecceda la RAM (paginazione).

Sistema operativo: Kernel

process manager

- creazione e terminazione dei processi
- sospensione e riattivazione dei processi
- gestione dei deadlock
- sincronizzazione tra processi

Deve garantire la distribuzione del tempo di calcolo a tutti i processi in esecuzione.

Deve garantire che ogni processo abbia tutte le risorse necessarie, e che ad ogni ripresa dell'elaborazione si trovi nel contesto corretto (context switch).

File system

Per l'utente il file system è composto da:

- **file** unità logiche di memorizzazione
 - entità atomica gestione memoria secondaria
 - collezione di informazioni correlate
 - rappresentazione logica uniforme informazioni correlate
- **directory** insieme di informazioni per organizzare e fornire informazioni sui file

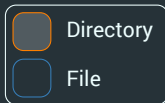
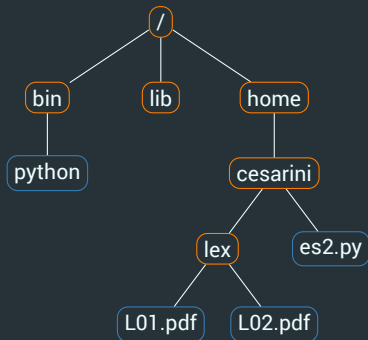
File

Attributi di un file in un filesystem:

- Nome
- Tipo necessario (SO) per decodificare le informazioni contenute nel file
- Posizione e dimensione
- Data e ora informazioni relative a creazione e ultima modifica
- Proprietà utente, gruppo, etc.
- Protezione informazioni per politiche di accesso ed uso del file

Cenni sul File System

L'insieme dei file e directory è organizzato ad albero:



Cenni sul File System

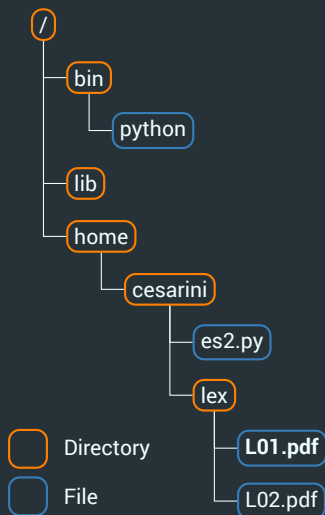
Anche dal punto di vista dell'utente un file sytem è strutturato come un albero:

Favorites	Name	Date Modified	Size	Kind
dario	3StepLinActivation	25 May 2015, 17:17	13 KB	Folder
Desktop	coagulation	10 Mar 2014, 12:52	2,4 GB	Folder
Works	EColi	6 Sep 2016, 11:43	6,1 MB	Folder
Downloads	JCP	28 Aug 2016, 15:25	204,03 GB	Folder
Lezioni	mappaTenda	9 Mar 2015, 11:56	23,2 MB	Folder
ownCloud	MatlabTestTumolo	18 Jun 2015, 13:04	32,5 MB	Folder
Dropbox	NaCoDynCIBB	2 Jun 2016, 11:30	622,8 MB	Folder
Google Drive	Filter	11 Jun 2016, 15:30	338 MB	Folder
Simulation	allFluxesDendro.pdf	11 Jun 2016, 17:22	14 KB	PDF
workspace	cbFluxesDendro.pdf	11 Jun 2016, 17:16	14 KB	PDF
labview	img	7 Jun 2016, 14:09	618 KB	Folder
matlab	NacoGlcExpsF01.h5	2 Jun 2016, 11:25	26,2 MB	HDF5 Data File
DLP	NacoGlcExpsF001.h5	2 Jun 2016, 12:04	28,1 MB	HDF5 Data File
Documents	NacoGlcExpsF001#1e4.h5	2 Jun 2016, 17:45	280,9 MB	HDF5 Data File
Applications	NACOirr.xml	2 Jun 2016, 11:08	46 KB	XML Document
Testi	scatterCBRespFerm.pdf	11 Jun 2016, 17:22	125 KB	PDF
Pictures	NACOirr.xml	2 Jun 2016, 11:08	46 KB	XML Document
Movies	noFilter	3 Jun 2016, 11:27	284,7 MB	Folder
Music	img	2 Jun 2016, 22:59	2,6 MB	Folder
AirDrop	NacoGlcExps#1e4.h5	2 Jun 2016, 13:23	282 MB	HDF5 Data File
All My Files	NACOirr.xml	2 Jun 2016, 11:03	46 KB	XML Document
iCloud Drive	tmp	2 Jun 2016, 14:10	212 bytes	TextEd...ument
Deleted Users	ODH	25 May 2015, 17:18	20 KB	Folder
Devices	input	2 Apr 2015, 09:08	2 KB	Folder
vulcano	ODH.dot	2 Apr 2015, 09:10	4 KB	Micros...e (.dot)
Macintosh HD	ODH.prj	2 Apr 2015, 09:08	137 bytes	Document
Remote Disc	ODH.xml	2 Apr 2015, 09:09	9 KB	XML Document
HD-PATU3	psoGibbs	25 Nov 2015, 08:42	9,2 MB	Folder
	PulseGen	26 Jul 2014, 11:38	783,7 MB	Folder
	RAS	7 Nov 2014, 12:15	1,03 GB	Folder
	RNG_MAP	20 May 2015, 12:08	167 KB	Folder
	RNG_MAPold	10 Mar 2015, 12:22	61,1 MB	Folder
	SAS2R	23 Mar 2015, 18:03	14 KB	Folder
	schlogl	4 Nov 2016, 11:41	164 bytes	Folder
	SecondRound	8 Oct 2014, 09:20	551,8 MB	Folder
	testOliver30	7 Feb 2015, 08:54	369 bytes	Folder
	triSome	26 Jul 2014, 11:38	48,41 GB	Folder

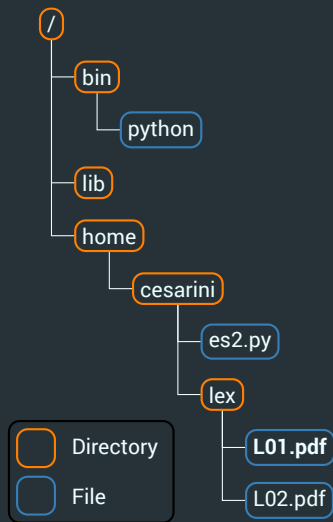
File system, terminologia

Dato il file L01.pdf nel file system a dx,

- L01.pdf è il **nome** del file (base name, o file/dir name)
- pdf è l'**estensione** del file
- /home/cesarini/lex/ è la **directory** che contiene il file
- /home/cesarini/lex/L01.pdf è il **percorso completo** (*fully qualified name* o *full path*)
- **percorso relativo**
 - Supponiamo che /home/cesarini sia la **directory di lavoro** corrente
 - lex/L01.pdf è il **percorso relativo** di L01.pdf rispetto alla



- Nella slide precedente abbiamo introdotto i concetti di nome e percorso (relativo e assoluto) per un file
- In maniera analoga è possibile parlare di nome e percorso (...) per una directory
- Dato il percorso `/home/cesarini/lex/L01.pdf`
 - `/` è la **root**, la directory che contiene tutte le altre directory e gli altri file di un system. E' l'inizio di ogni percorso
 - i rimanenti `/` sono dei separatori, si tratta di un espediente per separare i nomi di file e directory tra loro
 - il simbolo `/` è chiamato *slash*



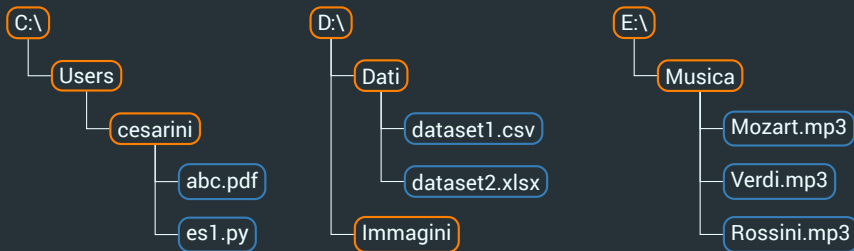
File Systems in Unix e Windows

Vediamo come la terminologia sui file system appena introdotta è declinata in due famiglie di sistemi operativi

OS	Root Directory	Directory Separator	Examples
Unix	/	/	/home/user/docs/Letter.txt
Win	[drive letter:]\	\	C:\User\MrX\Letter.txt \User\MrX\Letter.txt MrX\Letter.txt

- / è chiamato *slash*
- \ è chiamato *back-slash*
- Le [] attorno a *drive letter*: significano che l'indicazione del drive (disco) in Windows è opzionale
- Unix comprende Linux, Mac OS, ...
- Sia in Win sia in Unix sono definiti i percorsi sia assoluti sia relativi

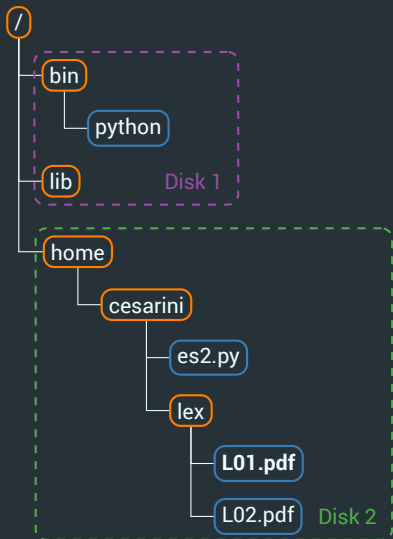
Windows, Dischi multipli e Foreste di Alberi



- I file system di Windows in realtà è una foresta di alberi
- Ogni drive (disco) è la radice di un albero separato dagli altri

Unix e dischi multipli

- Nel file system Unix apparentemente non c'è traccia dei diversi dischi
- In realtà i dischi sono *montati* sull'unico albero di directory
- Ciò rende possibile spostare gruppi di directory da un disco all'altro senza che l'utente se ne accorga



File, Directory e Meta dati

Ad un file, oltre al contenuto, sono associate delle informazioni (chiamate meta dati)

- Data e orario di *creazione* o *ultima modifica*
- Il proprietario e/o il creatore del file
- Permessi di accesso al file

Le stesse informazioni possono essere associate anche alle directory

Ecco un esempio di meta dati di un file system Unix

dir	User Perm.	Group Perm.	Others Perm.	Owner Name	Group Name	Size	Last Modified	Name
-	rw-	r-	r-	cesarini	gmc	1.6M	Dec 11 14:50	L06.pdf
-	rw-	r-	r-	cesarini	gmc	28K	Dec 11 14:50	L06.tex
d	rwx	r-x	r-x	cesarini	staff	170B	Dec 10 17:59	img

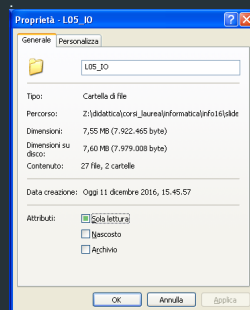
Alcuni meta dati di un sistema windows

Informazioni ottenibili a riga di comando

Last Modified	dir	size	name
10/12/2016 17.59	<DIR>		img
11/12/2016 15.20		1.657.585	L06.pdf
11/12/2016 15.22		29.410	L06.tex

- Nei sistemi Windows l'estensione del file (es. .pdf, .tex) da informazioni su qual applicazione può aprire il file
- Cambiando l'estensione, si modifica l'applicazione associata al file
- Se sul vostro PC windows non vedete le estensioni ...il vostro sistema operativo le sta mascherando (è l'impostazione di default)

Informazioni ottenibili tramite GUI

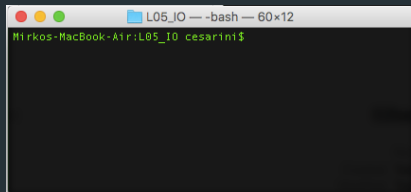
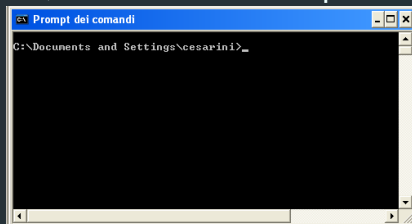


Shell CLI e File System

- Perché studiare le shell CLI in un'epoca di interfacce visuali?
- Nei sistemi operativi moderni, la shell è tuttora presente
- E' utile per svolgere attività ripetitive o complesse
- Es. per cancellare tutti i file che terminano con .txt (100 file, sparsi tra altri)
 - GUI: devo trascinare nel cestino 100 file
 - CLI: basta un comando (ora *non* ve lo spiego)
- La gestione di grossi dataset può essere effettuata su macchine (server) molto grandi, alle quali ci si connette via shell in remoto
- I linguaggi di programmazione
 - Il processo di compilazione può essere svolto tramite comandi dati a Shell
 - In questo corso vedremo l'interpretazione ed esecuzione del codice python, sia tramite CLI, sia tramite GUI

Shell CLI e File System

Questi sono due esempi di shell (Win e Unix)



- Directory corrente (concetto molto importante): la directory nella quale si sta lavorando, in un certo istante
 - Qualsiasi riferimento a file (senza aggiunta di percorso) fa riferimento a file della directory corrente
 - Se voglio utilizzare un file in una posizione diversa dalla directory corrente, dobbiamo fornire il percorso
 - I percorsi relativi, sono sempre calcolati a partire dalla directory corrente

Alcuni comandi di shell

Descrizione	Comando Win	Comando Unix	Note
Visualizza la directory corrente	cd	pwd	In win, cd senza parametri visualizza la directory corrente
Cambia directory corrente	cd <i>percorso</i>	cd <i>percorso</i>	Il nome del comando è lo stesso in entrambi i SO
Scelta del disco su cui lavorare	c:		al posto di c , potete far riferimento ad un qualsiasi disco del computer. Non si applica ai sistemi Unix
Per eseguire uno script python	python sc1.py	python sc1.py	il file sc1.py si deve trovare nella directory corrente
Per copiare un file	copy origine destinazione	cp origine destinazione	origine e destinazione sono i percorsi dei file corrispondenti
Per cancellare un file	del nomefile	rm nomefile	State attenti!!!

Eeguire un comando da shell

- Il prompt della shell attende un comando.
- Al prompt è possibile digitare il comando seguito da eventuali parametri. Premendo invio (il tasto return) il comando viene eseguito.
- Il prompt di Win e Unix sarà rappresentato rispettivamente da `>` e `$` qua di seguito

Win	Unix	Nota
<code>> python es1.py</code>	<code>\$ python es1.py</code>	Esecuzione di un file nella directory locale
<code>> python scripts\leggi.py</code>	<code>\$ python scripts/leggi.py</code>	parametro con percorso relativo
<code>> python C:\users\es3.py</code>	<code>\$ python /home/es3.py</code>	...percorso assoluto

- Un comando ha questo pattern:
`comando [parametro_1] [parametro_2] ... [parametro_N]`
- tutto ciò che è compreso tra `[e]` è opzionale

Eeguire uno script python da shell

- Con un editor di testi, salvate il file *primo.py* in una directory di vostra scelta. Nel file scrivete il testo `print("Hello Word")`
- eseguite i comandi seguenti (> e \$ non vanno digitati)

Win	Unix	Note
Menù avvio/e-segui/cmd	Applications/Utility/Terminal	Aprire la shell
> c: ↵ > cd \users\cesarini ↵	\$ cd /home/cesarini/ ↵	↵: punto in cui premere return
> python primo.py ↵	\$ python primo.py ↵	Osservate il risultato
> copy primo.py secondo.py ↵	\$ cp primo.py secondo.py ↵	creo nella directory corrente il file secondo.py, copia di ...
> del secondo.py ↵	\$ rm secondo.py ↵	Cancello il file appena creato

- Al posto di `\users\cesarini` o `/home/cesarini` digitate il percorso della directory in cui avete salvato il file

Alcuni trucchi

- Nell'usare la shell, occorre spesso ripetere l'esecuzione dei comandi
 - Utilizzate i tasti freccia, ↑ e ↓, per recuperare i comandi dati in precedenza
 - Questa funzionalità si chiama `history`
- Scrivere i percorsi dei file può essere noioso, soprattutto se i percorsi sono molto lunghi, come nell'esempio seguente:

```
/Users/cesarini/didattica/corsi_laurea/informatica/info16/
```

- Non appena avete scritto le prime lettere del nome di una directory provate a premere (il tasto) **tab**, anche diverse volte e state a vedere cosa succede.
- Questa funzionalità si chiama `auto completamento`