

## SHELL DOS

Sistema operativo senza interfaccia grafica

1

## SHELL DOS

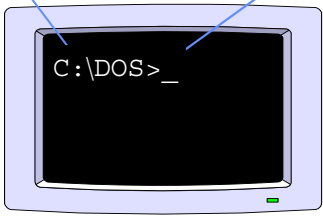
- DOS = *Disk Operating System*
- Il DOS mette a disposizione dell'utente una serie di comandi, introdotti usando la tastiera
- Prima di poter eseguire un nuovo comando è necessario aspettare la terminazione del precedente
- Per segnalare all'utente che è pronto ad accettare un comando, il sistema operativo presenta il *PROMPT* di sistema

2

## PROMPT DOS

- Ogni volta che il sistema è pronto ad accettare un comando vi sarà il cursore lampeggiante

prompt di sistema      cursore lampeggiante



3

## COMANDI DOS

- La **sintassi generale** di un comando DOS è:  
`COMANDO parametri [opzioni]`
- Non tutti i comandi hanno *parametri*;
- Non tutti i comandi hanno delle *opzioni*; queste possono anche essere omesse.  
Le parentesi [ ] (da non digitare nel comando) racchiudono le opzioni

4

## DOS

### o Osservazioni:

Un comando DOS è inviato al PC, solo dopo aver battuto il tasto di **INVIO** o **ENTER** o **RETURN** o ↵

Il DOS non fa distinzione tra lettere maiuscole e minuscole (*non è case sensitive*)

**JAVA invece SI**

Dopo ogni comando DOS e prima di parametri e opzioni si devono lasciare uno o più spazi (*blank*)

5

## C:\>HELP

### SELEZIONE COMANDI

- o AT Pianifica l'esecuzione di comandi o programmi su un computer.
- o CD Visualizza il nome o cambia la directory corrente.
- o CLS Cancella lo schermo.
- o CMD Avvia una nuova istanza dell'interprete dei comandi di Windows.
- o COPY Copia uno o più file in un'altra posizione.
- o DATE Visualizza o imposta la data.
- o DEL Elimina uno o più file.
- o DIR Visualizza un elenco di file e sottodirectory in una directory.
- o EXIT Termina il programma CMD.EXE (interprete dei comandi).
- o FC Confronta due file o gruppi di file e visualizza le differenze
- o FIND Ricerca una stringa di testo in un file o file.
- o FINDSTR Ricerca stringhe in file.
- o MD Crea una directory.
- o PATH Visualizza o imposta un percorso di ricerca per file eseguibili.
- o RD Elimina una directory.
- o START Avvia una finestra separata per eseguire un programma o un comando
- o TITLE Imposta il titolo della finestra per una sessione CMD.EXE.
- o TREE Visualizza graficamente la struttura delle directory di un'unit... o percorso.
- o TYPE Visualizza il contenuto di un file di testo.

6

## I file

- o L'unità elementare di informazione che il calcolatore è in grado di comprendere è il bit
- o Tutti i dati vengono trasformati in sequenze *ordinate* di bit
- o Per memorizzare più sequenze di bit in un disco è necessario che queste non si mescolino
- o Le sequenze vengono raggruppate in appositi "contenitori" detti *file*
- o Non tutti i file sono uguali
- o I file si dividono in due categorie:
  - file di testo
  - file binari

7

## I file

- o I *file di testo* sono quelli composti da soli caratteri ASCII
- o I file di testo possono essere letti e scritti con un qualunque *editor* (ad esempio *EDIT* di DOS)
- o I *file binari* sono invece memorizzati con una codifica che è diversa a seconda del programma che ha generato il file
- o I file binari non possono essere né letti né modificati con un normale editor

8

## I file

- Il comando *TYPE* del DOS serve per visualizzare il contenuto di un file di testo
- Se si cerca di leggere un file binario come se fosse un file di testo, le informazioni in esso contenute sarebbero interpretate come in codice ASCII pur non essendo tali
- Il risultato di tale operazione sarebbe imprevedibile, in quanto apparirebbero caratteri strani senza alcun significato apparente

9

## I file

- Ad ogni file viene associato un nome per distinguerlo dagli altri
- In DOS i nomi dei file possono essere lunghi da 1 a 8 caratteri
  - i caratteri possono essere alfa-numericici più alcuni caratteri "speciali": £, \$, %, &, \_ , etc.
- N.B. Non essendo il DOS *case sensitive* è indifferente specificare il nome del file scrivendolo in maiuscolo o in minuscolo

10

## I file

- Per aumentare le possibilità di diversificare i nomi dei file, e soprattutto per indicare in modo omogeneo gli stessi tipi di file, è possibile specificare una *estensione*
- In DOS l'estensione del file può essere lunga al più tre caratteri ed è separata dal nome dal carattere "."
- Un file deve sempre avere un nome mentre può non avere un'estensione
- Esempi

```
PIPPO.TXT  
PLUTO  
PAPERINO.C
```

11

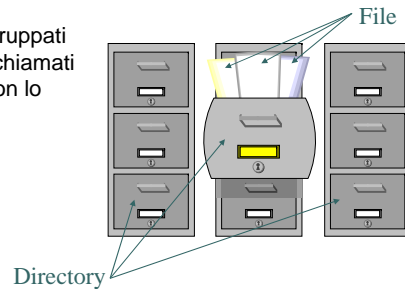
## I file

- Alcune estensioni tipiche:
  - .txt file di testo
  - .doc file del programma *MS-Word*
  - .c file di codice in linguaggio C
  - .xls file contenente una tabella del programma *MS-Excel*
- I file che possono essere mandati in esecuzione (in gergo "*lanciati*") devono avere una di queste tre estensioni:
  - .COM
  - .EXE
  - .BAT

12

## Le directory

- Per catalogare più efficientemente le informazioni, si possono raggruppare i file in qualche modo correlati tra loro
- I file possono essere raggruppati (collocati) in "contenitori" chiamati **directory** (nell'analogia con lo schedario sono i cassetti)



13

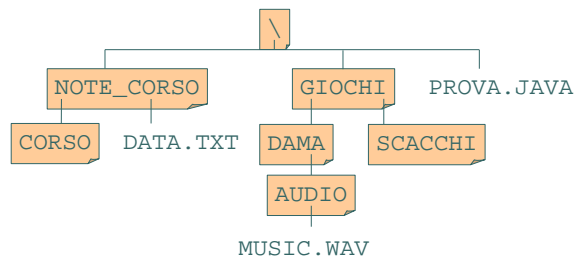
## Le directory

- Una directory ha un nome (stesse regole viste per i file)
- Le directory possono a loro volta contenere altre directory (*subdirectory*):
  - la struttura gerarchica che si viene a creare ha la forma di un albero rovesciato
  - la radice dell'albero, che in DOS si indica con il simbolo \ (*back slash*), è una directory creata alla *formattazione* (preparazione) del disco
- La struttura delle directory e i file costituiscono quello che viene chiamato *file system*

14

## Le directory

- Teoricamente non esistono limiti per i livelli di annidamento, ma in pratica il limite è imposto dai dispositivi fisici
- Esempio di albero di directory e file (file system)



15

## Le directory

- Quando viene creata una nuova directory, automaticamente vengono generate due directory speciali:
  - . è un semplice riferimento alla directory padre (cioè quella al livello immediatamente superiore)
  - .. è un semplice riferimento alla directory stessa

16

## Il pathname

- Per poter identificare un file in un file system non è quindi sufficiente specificare solo il suo nome e la sua estensione:  
*bisogna anche indicare in quale directory si trova*
- Per specificare univocamente la directory dove si trova il file bisogna indicare tutta la sequenza di directory dalla radice del file system fino alla directory contenente il file stesso

17

## Il pathname

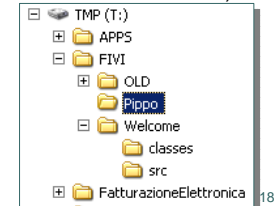
### pathname assoluto

- La sequenza di directory che parte dalla radice del file system viene chiamata *pathname assoluto*
- E' possibile "percorrere" l'albero delle directory
- La directory dove ci si trova in un dato momento è detta *directory corrente* (o *di lavoro*)

"T:\FIVI\pippo\lucidi.ppt"

### pathname relativo

- Un pathname è *relativo* quando invece di partire dalla radice del file system, si riferisce alla directory corrente (il corrispondente pathname assoluto viene calcolato dal sistema)



18

## Il pathname

- In DOS i nomi delle directory che compongono un pathname sono separati gli uni dagli altri dal carattere "\"
- I pathname assoluti iniziano sempre con il carattere "\ (che identifica la radice del file system)
- Esempio  
Il file `MUSIC.WAV` viene specificato *in modo assoluto* così:  
`\GIOCHI\TENNIS\SUONO\MUSIC.WAV`

19

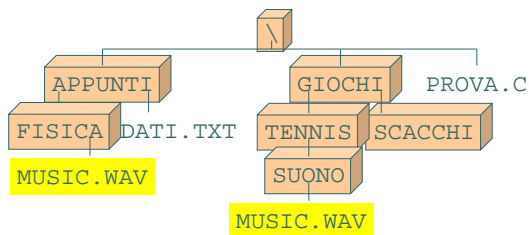
## Il pathname

- Esempi
  - a) Si supponga che la directory corrente sia `SCACCHI`, il file `MUSIC.WAV` viene specificato *in modo relativo* come:  
`..\TENNIS\SUONO\MUSIC.WAV`
    - la parte iniziale `..\` indica che dalla directory `SCACCHI` si risale di un livello (la directory padre, qui è `GIOCHI`)
  - b) Si supponga che la directory corrente sia `TENNIS`, il file `MUSIC.WAV` viene specificato *in modo relativo* come:  
`SUONO\MUSIC.WAV`

20

## Il pathname

- Due file possono avere lo stesso nome purché non siano contenuti nella stessa directory
- Due directory possono avere lo stesso nome purché non abbiano la stessa directory padre



21

## Unità logiche

- Il DOS utilizza più file system: ad ogni unità logica (disco) corrisponde un file system
- Le unità logiche sono specificate con lettere (non importa se maiuscole o minuscole) seguite dal carattere ":"
  - Il lettori di floppy hanno associate A: e B:
  - le altre unità logiche a partire dalla C: in su

22

## Unità logiche

- Quando si hanno più dischi, per identificare univocamente un file è necessario indicare anche in quale disco si trova il file system
- Esempio  
Supponendo che il file system visto in precedenza si trovi nel disco identificato dalla lettera C:, allora il file `MUSICA.WAV` viene specificato (qui in modo assoluto) in questo modo:

`C:\GIOCHI\TENNIS\SUONO\MUSICA.WAV`

23

## Unità logiche

- In realtà un singolo hard disk può contenere più di un file system: il disco viene suddiviso (*partizionato*) in più parti e ogni parte viene trattata come un disco indipendente
- Per cambiare il *disco corrente* (ossia passare da una unità logica ad un'altra) si digita al prompt la lettera dell'unità che si desidera raggiungere seguita da ":"
- Esempio  
Per raggiungere il disco C basta digitare  
`C: (seguito da INVIO)`

24

## Unita' logiche

- Il prompt standard indica l'unità corrente seguita dalla directory corrente; in questo modo l'utente ha sempre l'indicazione della sua posizione nel file system
- Esempio  
`C:\GIOCHI\SCACCHI>`
- Nel caso venga specificata un'unità non esistente il sistema operativo risponde con un messaggio di errore segnalando che l'unità prescelta è inesistente e chiedendo una nuova unità valida

25

## Caratteri jolly

- I *caratteri jolly* possono essere usati per indicare più file contemporaneamente
- Sostituiscono delle parti di nome o di estensione comuni a un gruppo di file
  - il carattere '\*' sostituisce una stringa (sequenza di caratteri) di lunghezza qualunque (anche nulla) a partire dalla posizione in cui è inserito
  - il carattere '?' sostituisce esattamente un carattere in corrispondenza della posizione in cui è inserito

26

## Caratteri jolly

- Si supponga di avere tre file:
  - `conti.bat`
  - `paghe.dat`
  - `prezzi.dat`
- Con il comando `COPY` si vogliono copiare tutti i file che iniziano con la lettera `P` e con qualunque estensione:  
`COPY P*. * destinazione`
- Si vogliono copiare tutti i file che hanno nome qualunque ed estensione che termina con `at`  
`COPY *.*at destinazione`

27

## Comando DIR

- Nome comando:  
`DIR`
- Funzione:  
visualizza il contenuto della directory specificata
- Sintassi:  
`DIR [unità][path][nome_dir]`
- Note:  
con l'opzione `/P` visualizza una schermata alla volta; con `/W` visualizza fino a 5 nomi per riga
- Esempi:  
`DIR C:\DATI\ /P`  
`DIR /W`  
`DIR ..\..\COMPITI`

28

## Comando CD

- Nome comando:  
`CD` `CHDIR`
- Funzione:  
cambia la directory di lavoro;  
senza parametri visualizza il nome della directory corrente
- Sintassi:  
`CD [path]`
- Nota: il cammino specificato (*path*) può essere assoluto o relativo (ma solo in riferimento all'unità corrente)
- Esempi:  

```
CD \MSDOS\UTIL
CD ..\PIPP0
CD .. (passa alla directory padre)
```

## Comando CLS

- Nome comando:  
`CLS`
- Funzione:  
cancella lo schermo
- Sintassi:  
`CLS`
- Nota:  
restituisce il prompt di sistema in alto a sinistra

## Comando COPY

- Nome comando:  
`COPY`
- Funzione:  
copia uno o più file in un'altra posizione o concatena una lista di file
- Sintassi:  
`COPY origine destinazione`  
`COPY file1 + file2 +... + file-n tuttoqui`
- Nota: nel caso venga usato per concatenare due o più file, `COPY` restituisce il nome dei file uniti mentre procede nella concatenazione

## Comando COPY

- Esempi:  

```
copy c:\game\dati.txt c:\copia.txt
copy ..\..\fax.doc fax.doc
copy a:\*. * c:\backup
copy c:\compiti.* a:\dati\scuola
copy c:\autoexec.b?t .
copy dati1.txt+dati2.txt dati12.txt
```



## Comando DEL

- Nome comando:  
**DEL** o **ERASE**
- Funzione:  
cancella i file specificati come parametro
- Sintassi:  
**DEL** *[unità][path]nome\_file*  
**ERASE** *[unità][path]nome\_file*
- Nota: con l'opzione **/P** il comando **DEL** chiede conferma per la cancellazione
- Esempi:  
DEL c:\dati\cifre.dat  
DEL ..\lucidi.ppt  
DEL program.bat

## Comando HELP

- Nome comando:  
**HELP**
- Funzione:  
accede ad una guida on-line dei comandi di DOS
- Sintassi:  
**HELP** *[comando]*
- Nota: non specificando un comando presenta la lista di tutti gli argomenti disponibili
- Esempi:  
HELP  
HELP COPY

## Comando MD

- Nome comando:  
**MD** o **MKDIR**
- Funzione:  
crea una directory dal nome specificato
- Sintassi:  
**MD** *[unità][path]nome\_directory*
- Nota:  
se si cerca di creare una directory già esistente questo viene segnalato con un messaggio di errore
- Esempi:  
MD C:\CORSI\FONDINFO  
MD CORSI

## Comando MOVE

- Nome comando:  
**MOVE**
- Funzione:  
permette di spostare uno o più file nella posizione specificata
- Sintassi:  
**MOVE** *[unità][path]origine*  
*[unità][path]destinazione*
- Note:  
può anche essere usato per cambiare nome alle directory; non era presente nelle vecchie versioni del DOS
- Esempi:  
● MOVE A:\\*.\* C:\BACKUP  
● MOVE LUCIDI SLIDE

## Comando PRINT

- o Nome comando:  
**PRINT**
- o Funzione:  
permette di stampare dei file
- o Sintassi:  
**PRINT** [/D:device] [drive:][path]filename[...]
- o Esempi:
  - PRINT LPR: c:\dati.txt

37

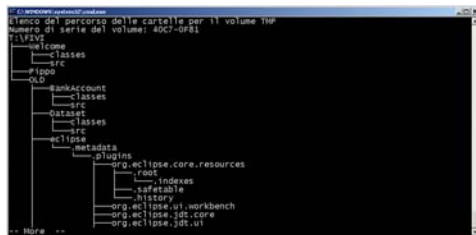
## Comando RD

- o Nome comando:  
**RD** o **RMDIR**
- o Funzione:  
permette di eliminare una directory specificata
- o Sintassi:  
**RD** [unità][path]nome\_directory
- o Nota:  
la directory specificata non può essere eliminata se non vuota
- o Esempi:
  - RD APPUNTI
  - RMDIR C:\UTIL\DATI
  - RD ..\..\SLIDE

38

## Comando TREE

- o Funzione:  
permette di visualizzare graficamente una struttura di file e sotto-directory
- o Sintassi:  
**TREE** [unità][path][dir o unità disco]



```

C:\Programmi\Eclipse>tree
C:\Programmi\Eclipse
├── eclipse
│   ├── classes
│   ├── src
│   ├── org.eclipse.ui
│   ├── org.eclipse.ui.workbench
│   ├── org.eclipse.jdt.core
│   └── org.eclipse.jdt.ui
└── org.eclipse.core.resources
    ├── root
    ├── indexes
    ├── safetable
    └── history

```

39