

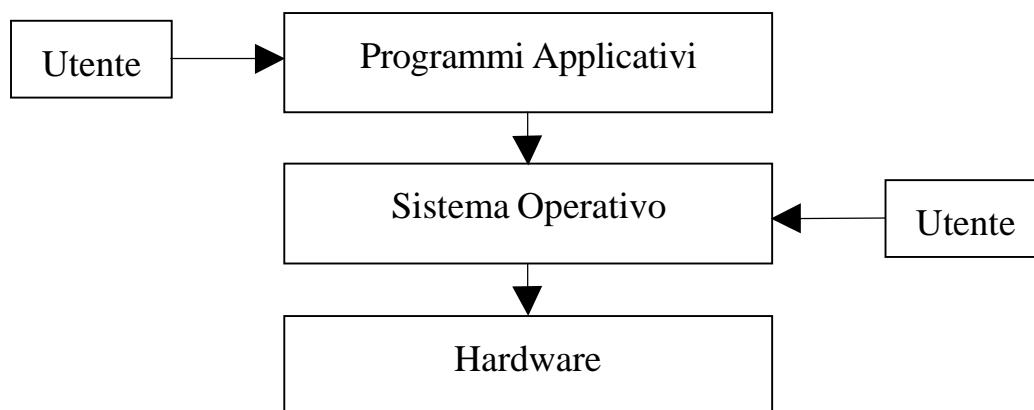
Breve guida al sistema operativo MS-DOS

1. Cos'è un sistema operativo

Un *sistema operativo* (SO) è un insieme di programmi che agisce come intermediario tra l'utente e l'hardware di un computer. Lo scopo di un SO è di fornire un ambiente nel quale un utente sia in grado di eseguire i programmi rendendo conveniente l'utilizzo di un computer. Uno scopo secondario, ma non meno importante, è quello di poter utilizzare l'hardware del computer in maniera efficiente.

Un sistema operativo è un elemento importante di quasi tutti i sistemi di calcolo. Un sistema di calcolo può essere suddiviso in quattro componenti: l'*hardware*, il *sistema operativo*, i *programmi applicativi* e gli *utenti*.

Da un punto di vista funzionale lo schema che riassume l'interazione fra utente e PC è il seguente:



L'hardware, composto dall'unità *centrale di elaborazione* (CPU), dalla memoria e dai dispositivi di input/output (I/O), fornisce le risorse fondamentali di calcolo. I programmi applicativi (compilatori, sistemi di database, programmi commerciali e giochi) definiscono il modo in cui queste risorse vengono utilizzate per risolvere i problemi computazionali degli utenti. Esistono svariate categorie di utenti (persone, macchine, altri computer) che necessitano di risolvere problemi di diversa natura; di conseguenza, possono esserci molti programmi applicativi diversi. Il SO *controlla e coordina* l'utilizzo dell'hardware da parte dei programmi applicativi per utenti.

Oltre all'hardware risorse non meno importanti sono il *software* e i *dati*, organizzati sotto forma di *file*. Il sistema operativo offre gli strumenti per utilizzare e organizzare in modo corretto queste risorse; esso non compie alcuna operazione utile di per sé, ma offre semplicemente un *ambiente* nel quale altri programmi possono lavorare in modo utile.

È quindi possibile considerare un SO come un *allocatore di risorse*: tempo di CPU, spazio di memoria, spazio di memorizzazione file, utilizzo dei dispositivi di I/O e così via. Il SO gestisce tali risorse allocandole, in base alle necessità definite dai loro compiti, a programmi e utenti specifici. Poiché possono esserci molte richieste, anche conflittuali, delle varie risorse, il sistema operativo deve decidere quali richieste soddisfare affinché il computer operi in modo equo ed efficiente.

Una visione leggermente diversa di un sistema operativo è quella incentrata sulla necessità di controllare i dispositivi di I/O e i programmi utente. Un SO è in effetti anche un *programma di controllo* che sorveglia, appunto, l'esecuzione di programmi utente in modo da impedire che vengano commessi errori o che il computer venga utilizzato in modo non corretto, soprattutto per quel che riguarda il funzionamento e la gestione dei dispositivi di I/O.

I SO esistono perché rappresentano una soluzione ragionevole al problema di realizzare un sistema di calcolo utilizzabile. Non esiste, comunque, una definizione universalmente accettata di cosa faccia parte o meno di un SO. Un punto di vista semplicistico è considerare "sistema operativo"

Breve guida al sistema operativo MS-DOS

tutto quanto viene fornito dal venditore. Comunque, i requisiti di memoria e le funzioni richieste variano molto da sistema a sistema. Una definizione più comune è quella secondo la quale il SO è il solo programma che funziona sempre sul computer (generalmente chiamato *kernel*), mentre gli altri sono programmi applicativi. Quest'ultima definizione è la più comune.

I sistemi operativi e l'architettura dei computer si sono influenzati reciprocamente in modo notevole. I sistemi operativi sono stati sviluppati per facilitare l'utilizzo dell'hardware. Una volta progettati e utilizzati i sistemi operativi, è risultato ovvio che modifiche al design dell'hardware potevano comportare semplificazioni ai sistemi operativi stessi.

In conclusione il SO serve a:

1. *Definire l'interazione con l'utente* - dobbiamo avere una interfaccia di comandi che ci permetta di specificare le azioni che, su nostra richiesta, vogliamo che il computer esegua.
2. *Interpretare ed eseguire i comandi dell'utente* - possono essere impartiti solo i comandi esistenti e questi devono avere una ben specifica sintassi, il sistema si deve occupare dei messaggi di errore e di informazione.
3. *Controllare direttamente l'hardware* - A noi interessano più che altro le funzioni del calcolatore e non come vengono eseguite a livello di linguaggio macchina. Ciò dà il vantaggio che il calcolatore può essere usato da chiunque, basta avere un manuale dei comandi disponibili, e che la programmazione è molto più veloce.
4. *Gestire e coordinare le risorse del sistema* - Il sistema operativo mentre esegue i comandi da noi impartiti si occupa anche di gestire al meglio le risorse, soprattutto quelle preziose come la memoria, i dischi, il video.
5. *Semplificare la gestione del sistema hardware agli altri programmi* - Quasi tutti i programmi possiedono *chiamate* al sistema operativo che svincolano i primi da funzioni di base rendendoli più efficienti e *robusti* ossia adattabili.

2. L'MS-DOS

L'MS-DOS è un sistema operativo nato nel 1981 per il PC IBM e adottato in seguito dai suoi successori più evoluti nonché dai numerosissimi compatibili delle più diverse marche. Mentre "MS" indica MicroSoft, "DOS" sta per Disk Operative System ossia sistema operativo a dischetti (il nome deriva dal fatto che può essere contenuto in un numero esiguo di dischetti). L'MS-DOS è stato lo standard di fatto dei sistemi operativi per microcomputer per più di quasi un ventennio. Oggi ci troviamo in una situazione di transizione dato l'avvento di Windows '95 e successive versioni, sempre della MicroSoft. In tali nuovi sistemi operativi viene mantenuto un *ambiente* MS-DOS per consentire l'utilizzo di applicativi e comandi usuali del DOS.

2.1 Le componenti dell'MS-DOS

Per capire a fondo il DOS è necessario conoscere le sue componenti. Si può pensare al DOS come se fosse diviso in due parti, ognuna delle quali è costituita da diversi file:

1. La parte principale del DOS, che è composta da tre programmi;
2. Un certo numero di programmi e file ausiliari.

I tre programmi principali sono:

- IO.SYS
- MSDOS.SYS
- COMMAND.COM

Breve guida al sistema operativo MS-DOS

L'IO.SYS contiene le funzionalità di base per la gestione delle periferiche di input/output. Queste funzionalità sono completamente trasparenti all'utente, in quanto vengono utilizzate dai programmi applicativi senza che l'utente se ne renda conto. L'MSDOS.SYS è il programma principale del DOS e presiede a tutte le esecuzioni. Il COMMAND.COM è il *processore dei comandi*. Si tratta del programma che legge i comandi inseriti da tastiera e li interpreta provvedendo a soddisfare le richieste dell'utente. Inoltre carica una serie di comandi elementari di sistema in memoria in modo che questi comandi siano disponibili immediatamente all'utente.

Qualunque programma o comando DOS non residente in memoria per essere eseguito deve essere processato, ossia deve essere acquisito dalla macchina mediante un comando manuale effettuato dall'utente (*digitazione*). Una volta eseguito il comando che lancia il programma il sistema operativo lo *legge* e lo esegue istruzione per istruzione mediante il processamento di dati, l'esecuzione di operazioni matematiche e logiche e l'utilizzo delle periferiche.

2.2 Il prompt e i dischi

Quando il sistema è pronto in modo di dialogo interattivo si ha un *prompt*. Il prompt è contraddistinto dall'essere la riga su cui lampeggia il cursore. Tale riga è evidenziata da alcuni simboli, solitamente C:\>. Il sistema ci ricorda su che unità disco ci troviamo (C:) oltre che sulla posizione all'interno di questa unità (\). Il > ha soltanto lo scopo di separare la parte visibile del prompt dal cursore lampeggiante.

Per cambiare l'unità disco basta digitare il nome dell'unità disco seguito da ":". Le unità dischi vengono nominati con una lettera: di solito i floppy disk sono A: e B:, l'hard disk principale (quello da cui si avvia il sistema) C:, gli altri hard disk, unità CD-ROM e così via D:, E:, etc..

2.3 File e nomi di file

Tutti i comandi DOS e dei programmi applicativi devono essere digitati per essere eseguiti dal sistema operativo. La stragrande maggioranza di questi comandi e applicazioni risiede non nella memoria del calcolatore, costosa e limitata, ma nelle cosiddette unità di massa rappresentate dai dischi (hard disk, floppy disk, CD-ROM). Le unità di massa hanno il vantaggio di poter conservare grandi quantità di programmi e di dati con costi notevolmente inferiori rispetto alla memoria interna del calcolatore. Ma come si accede ai programmi e ai dati?

I dati e i programmi contenuti in un disco sono organizzati in *file*, ognuno dei quali viene identificato con un nome. Il nome di un file è composto da due parti chiamate rispettivamente *nome di file* ed *estensione*.

Il nome di file deve avere una lunghezza compresa tra 1 e 8 caratteri, e può includere lettere dell'alfabeto, numeri e alcuni segni di punteggiatura e simboli speciali. Non è consentito inserire in un nome di file degli spazi vuoti né caratteri non alfabetici (quali : ; , . /). Si possono usare lettere maiuscole o minuscole (in ogni caso il DOS le considererà tutte maiuscole). Di seguito vengono riportati alcuni esempi di nomi di file:

```
NOMEFILE
12
A
ABC_123
```

L'estensione è una breve appendice che viene aggiunta al termine del nome di file. I due elementi sono separati da un punto; un nome di file con un'estensione si presenta dunque nel modo seguente: NOMEFILE.EXT. L'estensione ha una lunghezza massima di tre caratteri e la sua specifica è facoltativa. Un file deve avere un nome, ma non è indispensabile che abbia un'estensione. Nel caso manchi l'estensione, il punto che la separerebbe dal nome del file non deve essere inserito.

Breve guida al sistema operativo MS-DOS

Lo scopo di un'estensione è quello di indicare la categoria a cui appartiene il nome di file. Si tratta di un metodo facoltativo per specificare un tipo di file in modo da poterlo individuare più facilmente. Ad esempio LETTERA.TXT indica un file di testo che contiene una lettera.

Seguono alcuni esempi di nomi di file, con e senza estensione:

NOMEFILE.EXT	lunghezza massima consentita
A	lunghezza minima consentita
12345678	i numeri sono ammessi
DATI-GEN	i trattini sono ammessi
DATI_GEN	i caratteri di sottolineatura sono ammessi
TELEFONO.LST	i nomi di file possono indicarne il contenuto
LUGLIO.91	un altro esempio di nome di file esplicativo

Si noti bene che alcune estensioni sono obbligatorie in MS-DOS: ci si riferisce a COM, EXE, BAT e SYS. Tali estensioni, infatti, sono destinate dal sistema operativo a definire programmi e comandi particolari:

- COM ed EXE per programmi e comandi codificati in codice binario (eseguibili);
- BAT per le cosiddette procedure batch;
- SYS per i file di sistema come IO.SYS e l'MSDOS.SYS.

2.4 Alberi di directory

Per memorizzare i file abbiamo a disposizione floppy e hard disk (HD). Il computer quando si accende deve sempre essere collegato ad un dispositivo a dischi. Mentre nei vecchi sistemi l'avvio veniva effettuato da dischetto (il prompt era A:\>) ormai si attiva da un HD. Le unità a dischi flessibili sono abbastanza capienti e le unità a disco rigido ormai sono diventate gigantesche, fino a 6 Gb. Si intuisce subito che muoversi in strutture così grandi diventa una cosa veramente difficoltosa, soprattutto quando i file contenuti sono da qualche centinaio a migliaia o addirittura milioni!

È necessario trovare un criterio logico per raggruppare i file. Il SO DOS ha realizzato ciò mediante il concetto di *direttorio* (dall'inglese *directory*). Possiamo pensare i dischi come ad una collezione di scatole ognuna delle quali ha un nome ed una certa dimensione: i file. Questi possono essere raggruppati sulla base del loro contenuto o dell'appartenenza ad un programma specifico, ad esempio di video scrittura o della gestione del magazzino. Comunque ciò non basterebbe. È necessario utilizzare degli ulteriori contenitori, non più fisici ma *logici* ossia *virtuali*, appunto le directory:

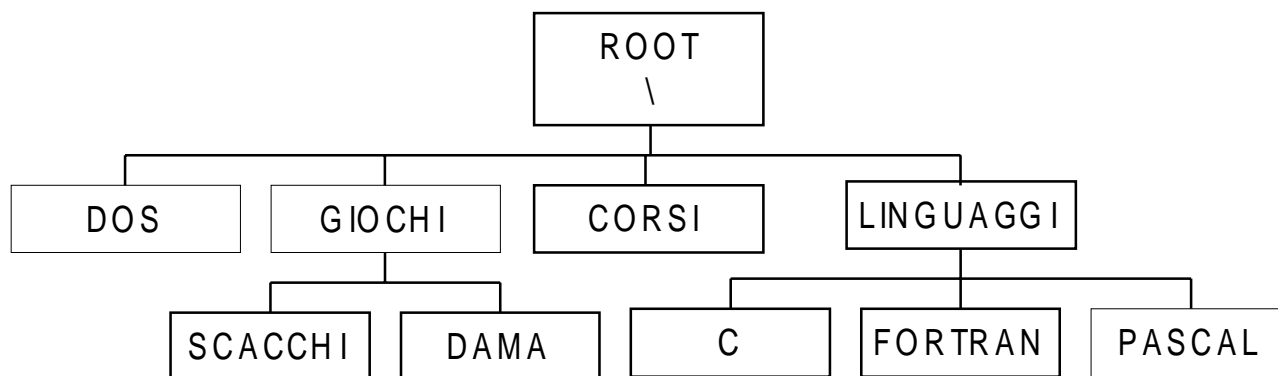
- Una directory è un insieme di file tra loro correlati da una relazione.
- Ogni unità fisica è organizzata in direttori.
- Un direttorio ha un nome con un massimo di 8 caratteri.
- Una directory può contenere altre directory che si chiamano sottodirectory.
- I direttori sono strutturati ad alberi.
- Tutti gli alberi di directory hanno una radice.

Il disco, da un punto di vista gerarchico è organizzato in direttori e sottodirettori. La directory principale di una unità a disco, ossia quella che contiene tutti gli altri direttori, è la directory *radice* o *root*. La root è rappresentata con un nome simbolico ossia “\” anche detto *barra rovescia* o

Breve guida al sistema operativo MS-DOS

backslash, proprio quel simbolo che abbiamo visto prima relativamente al prompt; ciò vuol dire che quando il sistema si avvia si attiva sulla radice dell'unità a disco fisso C:.

STRUTTURA AD ALBERO DELLE DIRECTORY



Nella figura l'analogia con gli scatoloni è evidente: ho quattro sottoarchivi, alcuni di essi contengono altri sottoarchivi. All'interno di ogni directory vi sono altre directory e/o file.

Il percorso dato dal prompt ha una notevole importanza. Supponiamo di volere cercare il file a.a all'interno del disco rigido. Se il sistema si è avviato su disco A: è chiaro che prima di usare il comando di visualizzazione del file devo accedere al disco C: (digitando al prompt C:). Il file quindi viene identificato non solo dal suo nome ma anche dal suo *percorso* o *pathname*. Ciò è equivalente all'indirizzo di casa: io sono Nicola Bianchi residente a Palermo in via Guglielmo Marconi n.11, Scala A, 3 piano, interno 22. Ognuna di queste specifiche da sola ha poco significato ma tutte insieme ci permettono non solo di inviare correttamente la nostra posta ma anche di evitare casi di omonimia. Similmente ciò vale anche per i file. A me non basta sapere che il file a.a è contenuto nel disco C:, ma devo anche sapere che è contenuto nella sottodirectory PASCAL della directory LINGUAGGI del disco C:. Solo ora saprò come raggiungere il file. Il pathname è la sequenza di directory che permette di raggiungere il file ossia: C:\LINGUAGGI\PASCAL.

2.5 I metacaratteri o caratteri jolly

Tramite dei caratteri particolari, definiti *metacaratteri*, ci si può riferire a più file contemporaneamente. I metacaratteri costituiscono uno strumento per specificare parzialmente un nome di file in modo tale che più file corrispondano alle specifiche enunciate.

Esistono due metacaratteri: l'asterisco (*) e il punto interrogativo (?). Entrambi possono essere usati in diversi modi nelle specifiche di file. "*" sostituisce intere parole o gruppi di caratteri (nel caso di nomi di file da un minimo di 1 lettera ad un massimo di 8). Ad es. DIR *.BAT; DIR LET*.DOC; DIR PIPPO.*; DIR *.* (equivalente a DIR). Il punto interrogativo sostituisce un singolo carattere nella posizione in cui viene inserito; è possibile inserire più caratteri "?": ad es. DIR ???.* (visualizza tutti i file con nome di 3 caratteri e qualunque estensione); DIR LETT?RA.TXT (visualizza ad es. i file LETTERA.TXT e LETTURA.TXT).

2.6 Gestione della memoria nel DOS

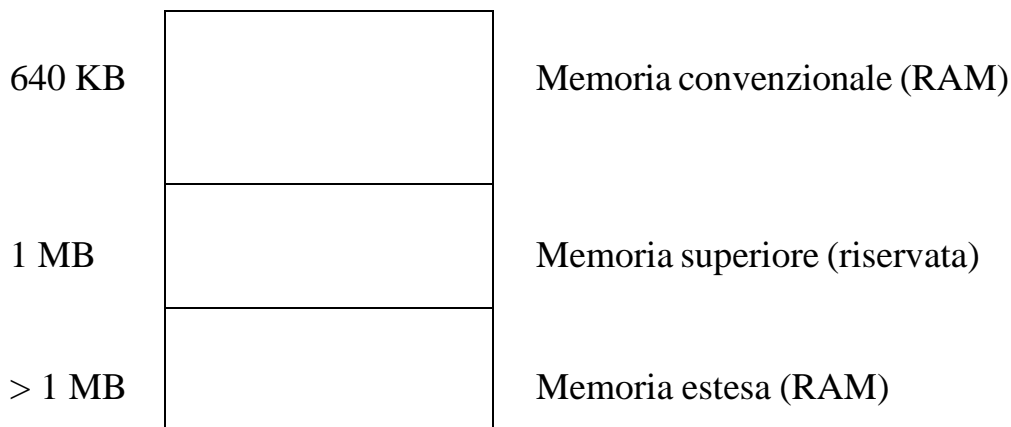
Come già noto, la memoria tecnologicamente è realizzata da dispositivi RAM.

La memoria del nostro PC è vista dal DOS in tre porzioni, come indicato nella figura:

quella che va da 0 a 640 KB (come indirizzi), identificata con il nome di *memoria convenzionale*, è la memoria RAM utilizzata dal DOS per memorizzare se stesso e per caricare ed eseguire i programmi. Vi è poi una seconda porzione che viene indicata con il termine di *memoria superiore* o *riservata* che

Breve guida al sistema operativo MS-DOS

non è a disposizione del DOS e neanche dell'utente: è uno spazio di indirizzi riservato a contenere delle funzioni speciali. I primi P.C. (8086, 8088) avevano a disposizione al massimo questi due primi blocchi di memoria, ossia la CPU poteva indirizzare soltanto un milione di byte e pertanto la disponibilità di memoria era limitata a non più dei 640 Kb di memoria convenzionale, essendo la restante parte fino ad un Mb dedicato a funzioni particolari, che potremmo chiamare *funzioni hardware*.



Andando ad analizzare la memoria convenzionale ci accorgiamo che essa è costituita da due zone differenti: una che contiene la parte residente del DOS ed una porzione libera ove il DOS carica i programmi da eseguire per conto dell'utente.

Se inizialmente il limite dei 640 Kb poteva considerarsi sufficiente per operare con gli applicativi disponibili, con l'evolversi del DOS e delle funzioni richieste ai programmi e quindi con il crescere della dimensione dei programmi, tale spazio è divenuto angusto. Di conseguenza è nata l'esigenza di ampliare la RAM disponibile. Con le vecchie CPU come l'8086 saremmo stati comunque limitati intrinsecamente ad 1 Mb di memoria dal *potere di indirizzamento* della CPU ma con l'avvento dei processori 486 e quindi della tecnologia Pentium, l'hardware ha avuto la possibilità di estendere lo spazio di indirizzi e quindi, pur rimanendo la porzione di memoria riservata si è abbattuta la soglia del Mb di memoria arrivando, ai giorni nostri, ad un potere di indirizzamento oltre i 250 Mb. La parte di memoria che supera il Mb viene denominata *memoria estesa*. Il DOS comunque non è in grado di gestire la memoria al di sopra del Mb, ma necessita di opportuni programmi strettamente legati alla gestione dell'hardware che rendono possibile l'utilizzo della memoria estesa. In Windows '95 non si hanno tali problemi e il potere di indirizzamento dipende soltanto dalla CPU.

2.7 Comandi principali

Un comando ha una sua *sintassi* ossia ha delle regole precise che permettono al DOS l'esatta interpretazione di quanto vogliamo fare. La sintassi viene indicata in tal modo:

<nome comando> [parametri] [opzioni]

si noti che si è usato < > e []; le prime indicano l'obbligatorietà di quello che contengono, le seconde invece ci informano che il campo è opzionale. Si noti anche che i campi sono separati da spazi. Tutti i comandi hanno un'opzione sempre disponibile "/ ?". Essa fornisce un aiuto in linea del comando stesso.

Breve guida al sistema operativo MS-DOS

2.7.1 DIR

Visualizza l'elenco dei file e delle sottodirectory in una directory.

Sintassi DIR [unità:][percorso][nomefile] [/P] [/W] [/O[:]ordinamento]] [/S]
 [/B] [/L] [/V]

Parametri

unità: Specifica l'unità di cui si desidera visualizzare l'elenco.
percorso Specifica la directory di cui si desidera visualizzare l'elenco.
nomefile Specifica il file o il gruppo di file di cui si desidera visualizzare l'elenco.

Opzioni

/P Visualizza l'elenco una schermata per volta. Per passare alla schermata successiva, è sufficiente premere un tasto.
/W Visualizza l'elenco in formato ampio, elencando fino a cinque file o directory per riga.
/O Elenca i file secondo un ordinamento specificato.
ordinamento N Per nome (alfabetico).
 S Per dimensione (dal più piccolo).
 E Per estensione (alfabetico).
 D Per data e ora (dal più recente).
 G Directory prima dei file. Prefisso “-” per invertire l'ordine.
/S Visualizza i file nella directory specificata e in tutte le sottodirectory.
/B Usa il formato semplice (solo nomi, senza intestazioni).
/L Usa le lettere minuscole.
/V Visualizzazione completa.

Le opzioni possono essere preimpostate nella variabile d'ambiente DIRCMD.

2.7.2 CD

Visualizza il nome della directory corrente, oppure cambia la directory corrente.

Sintassi CHDIR [unità:][percorso]
 CHDIR[..]
 CD [unità:][percorso]
 CD[..]

Parametri

unità: Specifica l'unità che si desidera accedere.
percorso Specifica la directory che si desidera accedere.
.. Indica che si desidera passare alla directory superiore.

Digitare CD senza parametri per visualizzare l'unità e la directory corrente.

Breve guida al sistema operativo MS-DOS

2.7.3 MD

Crea una directory.

Sintassi MKDIR [unità:]percorso
MD [unità:]percorso

Parametri

percorso Specifica la directory che si desidera creare.
unità: Indica l'unità su cui creare la directory.

2.7.4 RD

Rimuove (elimina) una directory.

Sintassi RMDIR [unità:]percorso
RD [unità:]percorso

Parametri

percorso Specifica la directory che si desidera eliminare.
unità: Indica l'unità su cui eliminare la directory specificata.

Il comando ha esito positivo solo se la directory specificata non contiene alcun file.

2.7.5 COPY

Copia uno o più file in un'altra posizione. Se vengono copiati più file, MS-DOS visualizza il nome di ciascun file man mano che viene copiato.

Sintassi COPY origine [destinazione] [/V]

Parametri

origine Specifica la posizione ed il nome di un file o di un gruppo di file che si desidera copiare. Il parametro origine può consistere di una lettera di unità seguita dai due punti, di un nome di directory, di un nomefile o di una loro combinazione.

destinazione Specifica la posizione ed il nome di un file o di un gruppo di file nei quali si desidera copiare. Il parametro destinazione può consistere di una lettera di unità seguita dai due punti, di un nome di directory, di un nome di file o di una loro combinazione.

Opzione

/V Controlla che i nuovi file siano stati copiati in maniera corretta.

Esempi

Il seguente comando copia un file cambiandone il nome: copy memo.doc letter.doc

Per copiare il file NOTE.TXT dall'unità e dalla directory corrente (C:\DOC) alla directory MIENOTE (sottodirectory di DOC) digitare il seguente comando:
copy note.txt mienote

Breve guida al sistema operativo MS-DOS

2.7.6 XCOPY

Copia file (ad eccezione dei file di sistema e nascosti) e directory, incluse le sottodirectory. Con questo comando è possibile copiare tutti i file di una directory, compresi i file nelle relative sottodirectory.

Sintassi XCOPY origine [destinazione] [/Y|/-Y] [/D:data] [/P] [/S] [/E] [/V]

Parametri

origine Specifica la posizione e i nomi dei file che si desidera copiare. L'origine deve includere o un'unità o un percorso.

destinazione Specifica la destinazione dei file che si desidera copiare. La destinazione può includere una lettera di unità seguita dai due punti, un nome di directory, un nome di file o una loro combinazione.

Opzioni

/Y Indica che XCOPY sostituisce i file esistenti senza chiedere conferma. Per impostazione predefinita, se si specifica un file come file di destinazione, XCOPY chiederà se si desidera sovrascrivere il file esistente.

/-Y Indica che XCOPY chiede la conferma quando sostituisce un file esistente.

/D:data Copia solo i file di origine che siano stati modificati nella data specificata o successivamente.

/P Chiede conferma alla creazione di ciascun file di destinazione con un messaggio.

/S Copia le directory e le sottodirectory, purché non siano vuote. Se viene omessa questa opzione, XCOPY ha effetto solo all'interno di una singola directory.

/E Copia qualsiasi sottodirectory, anche se vuota. Con questa opzione è necessario utilizzare l'opzione /S.

/V Verifica il modo in cui ciascun file è stato scritto sul file di destinazione per assicurarsi che i file di destinazione siano l'esatta copia del file di origine.

Come specificare se destinazione è un file o una directory

Se destinazione non contiene una directory esistente e non termina con una barra rovesciata (\), XCOPY visualizza un prompt in questo formato:

Destinazione è un nome di file o di directory?

(F = file, D = directory)

Premere F se si desidera che il file o i file vengano copiati su un file, D per copiarli su una directory.

Esempi

L'esempio seguente copia tutti i file e le sottodirectory (incluse anche le sottodirectory vuote) dal disco nell'unità A: al disco nell'unità B: xcopy a: b: /s /e

L'esempio seguente utilizza le opzioni /D: e /V: xcopy a: b: /d:04/11/90 /s /v

Breve guida al sistema operativo MS-DOS

In questo esempio solo i file nel disco dell'unità A: che sono stati scritti il 04/11/90 o successivamente, vengono copiati sul disco dell'unità B:. Dopo l'operazione di copia, il comando XCOPY confronta i file sui due dischi per assicurarsi che siano uguali.

2.7.7 DEL (Erase)

Elimina i file specificati.

Sintassi DEL [unità:][percorso] nomefile [/P]

Parametri

unità: Specifica l'unità in cui eliminare il/i file.
percorso Specifica la directory in cui eliminare il/i file.
nomefile Specifica il/i nomi dei file che si desidera eliminare.

Opzione

/P Chiede conferma prima di eliminare il file specificato. Il comando DEL visualizza il nome con una richiesta di conferma dell'eliminazione nel seguente formato: nomefile, Eliminare (S/N)?
Premere S per confermare l'eliminazione, N per annullarla e visualizzare il file successivo (se è stato specificato un gruppo di file), oppure premere CTRL+C per interrompere l'esecuzione del comando.

Eliminazione contemporanea di più file

E' possibile eliminare tutti i file di una directory con il comando DEL seguito da [unità:]percorso. E' inoltre possibile eliminare più file contemporaneamente utilizzando i caratteri jolly (* e ?). E' necessario comunque fare molta attenzione nell'utilizzare i caratteri jolly, in quanto si corre il rischio di eliminare accidentalmente alcuni file necessari. Si supponga, ad esempio, di digitare il seguente comando:

del *.*

DEL visualizzerà il seguente messaggio:

Tutti i file della directory verranno eliminati! Continuare? (S/N)

Premere S e quindi INVIO per cancellare i file nella directory corrente, premere invece N ed INVIO per annullare l'operazione.

Prima di eliminare un gruppo di file utilizzando i caratteri jolly, è meglio utilizzare gli stessi caratteri jolly con il comando DIR per visualizzare un elenco con i nomi di tutti i file compresi nel gruppo e assicurarsi di eliminare solo i file desiderati.

Esempi

Per eliminare tutti i file di una directory chiamata TEST sull'unità C, si possono utilizzare i seguenti comandi indifferentemente: del c:\test del c:\test*.*

2.7.8 DELTREE

Elimina una directory con tutti i file e sottodirectory relative.

Sintassi DELTREE [/Y] [unità:]percorso

Breve guida al sistema operativo MS-DOS

Parametri

unità: Specifica l'unità in cui eliminare la directory.
percorso Specifica il nome della directory che si desidera eliminare.

Il comando DELTREE eliminerà tutti i file contenuti nella directory specificata e tutte le sottodirectory con i relativi file.

Opzione

/Y Esegue il comando DELTREE senza chiedere conferma dell'eliminazione.

Esempio

Per eliminare la directory TEMP sull'unità C, includendo tutti i file e le sottodirectory di TEMP, digitare quanto segue al prompt dei comandi: `deltree c:\temp`

2.7.9 REN (Rename)

Cambia il nome di uno o più file. E' possibile assegnare un nuovo nome a tutti i file che corrispondono al nome specificato. Non è possibile utilizzare questo comando per assegnare un nuovo nome a file di unità diverse o per spostarli in nuove directory.

Sintassi REN [unità:][percorso]nomefile1 nomefile2

Parametri

unità: Specifica l'unità in cui vi sono dei file da rinominare.
percorso Specifica la directory in cui vi sono dei file da rinominare.
nomefile1 Specifica il/i file a cui si desidera assegnare un nuovo nome.
nomefile2 Specifica il nuovo nome del file o, se si utilizzano i caratteri jolly, i nuovi nomi dei file. Non è possibile specificare una nuova unità o un percorso diverso.

Utilizzo dei caratteri jolly con il comando RENAME

E' possibile utilizzare i caratteri jolly (* e ?) in tutti e due i parametri nomefile. Se i caratteri jolly vengono utilizzati nel parametro nomefile2, i caratteri che essi rappresentano risultano identici ai corrispondenti caratteri nel parametro nomefile1.

RENAME non ha effetto se nomefile2 è già esistente.

Se per nomefile2 viene specificato il nome di un file già esistente, RENAME visualizza il seguente messaggio: Nome di file esistente o file non trovato

Esempi

Si supponga di volere modificare l'estensione di tutti i file nella directory corrente con estensione TXT; ad esempio, si supponga di volere modificare l'estensione TXT in DOC. Digitare a questo fine il seguente comando: `ren *.txt *.doc`

2.7.10 MOVE

Sposta file e rinomina file o directory.

Sintassi

Per spostare uno o più file:

MOVE [/Y | /-Y] [unità:][percorso]nomefile1[...] destinazione

Per rinominare una directory:

MOVE [/Y | /-Y] [unità:][percorso]nomedir1 nomedir2

Breve guida al sistema operativo MS-DOS

Parametri

unità: Specifica l'unità che contiene il/i file che si vogliono spostare.
percorso Specifica il percorso che contiene il/i file che si vogliono spostare.
nomefile1 Specifica il/i nomi dei file che si vogliono spostare.
destinazione Specifica la nuova locazione dei file. La destinazione consiste della lettera di unità seguita dai due punti, il nome di directory o una combinazione dei due. Se si stanno spostando dei file, è possibile includere il nome di un file per rinominare il file durante lo spostamento.
nomedir1 Specifica la directory che si vuole rinominare.
nomedir2 Specifica il nuovo nome della directory.

Opzioni

/Y Non chiede conferma prima di creare una directory o di sovrascrivere la destinazione.
/-Y Chiede conferma prima di creare una directory o di sovrascrivere la destinazione.

Esempi

Il seguente comando sposta un file, cambiandone il nome, da una directory ad una altra:

```
MOVE C:\DOC\LETTERA.TXT C:\MIE\MICHELE.LET
```

Il seguente comando rinomina la directory CC in GCC: `MOVE C:\CC C:\GCC`

2.7.11 TYPE

Visualizza il contenuto di file di testo.

Sintassi TYPE [unità:][percorso]nomefile

Parametri

unità: Specifica l'unità che contiene il file da visualizzare.
percorso Specifica il percorso che contiene il file da visualizzare.
nomefile È il nome del file di testo da visualizzare.

Nota

Se il contenuto del file è più grande di una videata si può utilizzare il comando MORE da solo o con TYPE. In quest'ultimo caso il comando da specificare è:

TYPE [unità:][percorso]nomefile | MORE (dove “|” viene chiamato *pipe*)

2.7.12 MORE

Visualizza le informazioni una schermata alla volta.

Sintassi MORE [unità:][percorso]nomeFile

Parametri

unità: Specifica l'unità che contiene il file da visualizzare.
percorso Specifica il percorso che contiene il file da visualizzare.
nomefile È il nome del file di testo da visualizzare.

Breve guida al sistema operativo MS-DOS

Se vi sono comandi che emettono messaggi o output più grandi di una videata è possibile utilizzare la seguenti sintassi: <comando> [parametri] [opzioni] | MORE

dove “comando” specifica il comando di cui visualizzare l’output una schermata alla volta.

Esempi

MORE LETTERA.TXT - Visualizza il contenuto del file LETTERA.TXT una schermata alla volta.

TYPE LETTERA.TXT | MORE - Come sopra.

DIR | MORE - Visualizza il contenuto della directory corrente una schermata alla volta.

2.7.13 FORMAT

Formatta il disco nell’unità specificata per accettare i file di MS-DOS.

Il comando FORMAT crea una nuova directory principale e una tabella di assegnazione dei file per il disco. E’ anche possibile controllare le aree danneggiate sul disco ed eliminare tutti i dati sul disco. Per consentire a MS-DOS di utilizzare un disco nuovo, utilizzare innanzi tutto questo comando per formattare il disco.

Sintassi FORMAT unità: [/V[:etichetta]] [/Q] [/F:dimensione][[/S]]

Parametri

unità: Specifica l’unità che contiene il disco che si desidera formattare.

Opzioni

/V:etichetta Specifica l’etichetta del volume. Un’etichetta del volume identifica il disco e può contenere un massimo di 11 caratteri. Se si omette l’opzione /V o la si utilizza senza specificare l’etichetta del volume, MS-DOS richiede l’etichetta del volume dopo che ha terminato la formattazione. Se si formatta più di un disco utilizzando il comando FORMAT, verrà assegnata la stessa etichetta di volume a tutti i dischi.

/Q Specifica una formattazione veloce di un disco. Tramite questa opzione, FORMAT elimina la tabella di assegnazione dei file (FAT) e la directory principale di un disco formattato in precedenza, ma non esamina il disco per verificare la presenza di aree danneggiate. Si dovrebbe utilizzare l’opzione /Q per formattare solo dischi già formattati in precedenza che non sono sicuramente danneggiati.

/F: Specifica la dimensione del disco floppy da formattare. Utilizzare uno dei seguenti valori per “dimensione”:

720 indica dischi da 3,5 pollici di capacità 720Kb, a doppia faccia e doppia densità; 1440 indica dischi da 3,5 pollici di capacità 1,44 Mb, a doppia faccia e quadrupla densità (ormai i più diffusi).

Avvertenza: Non formattare un disco floppy in misura maggiore a quella stabilita.

/S Copia i file di sistema in uso IO.SYS, MSDOS.SYS e COMMAND.COM dall’unità disco di avvio del sistema su un disco formattato che può essere utilizzato come disco di avvio del sistema.

Formattazione di un disco rigido

Quando si utilizza il comando FORMAT per formattare un disco rigido, MS-DOS visualizza un messaggio di questo tipo prima di iniziare la formattazione del disco rigido:

Breve guida al sistema operativo MS-DOS

ATTENZIONE! TUTTI I DATI DELL'UNITÀ X: DEL DISCO RIGIDO
ANDRANNO PERSI!

Procedere con la formattazione? (S/N)

Per formattare il disco rigido, premere S, altrimenti premere N.

Si noti che non è necessario specificare la dimensione del disco fisso; questa verrà stabilita dal sistema operativo tramite un controllo effettuato sulla periferica.

Esempi

Per formattare un nuovo disco floppy nell'unità A: utilizzando la dimensione predefinita, digitare il seguente comando: `format a:`

Per eseguire una formattazione rapida su un disco già formattato nell'unità A:, digitare il seguente comando: `format a: /q`

Per formattare un disco floppy nell'unità A: cancellandone completamente i dati, digitare il seguente comando: `format a: /u`

Per formattare un disco floppy da 720Kb nell'unità A e copiarvi i file del sistema operativo per creare un disco di sistema, digitare i seguenti comandi:
`format a: /f:720 /s`

Per formattare un disco floppy nell'unità A e assegnargli l'etichetta di volume "DATI", digitare il seguente comando: `format a: /v:DATI`

2.7.14 LABEL

Crea, modifica o elimina l'etichetta di volume di un disco.

Sintassi `LABEL [unità:][etichetta]`

Parametri

unità: Indica l'unità disco in cui va dato un nome al dischetto

etichetta È il nome da dare al dischetto (max 8 caratteri)

Se non viene specificata l'etichetta viene visualizzato il seguente messaggio:

Il volume nell'unità C è MASTERBLADE

Numero di serie del volume: 1B6E-14DA

Etichetta di volume (11 caratteri, INVIO per non darne)?

Digitare il nome nuovo oppure invio per eliminare l'etichetta corrente.

2.7.15 DISKCOPY

Copia il contenuto di un disco floppy su un altro disco floppy. Effettua una copia fisica, copiando byte per byte il primo disco sul secondo.

Sintassi `DISKCOPY [unità1: [unità2:]]`

Breve guida al sistema operativo MS-DOS

Parametri

unità1: Indica il disco origine.

unità2: Indica il disco destinazione.

Specificando la stessa unità come origine e destinazione (ad es. DISKCOPY A: A:) il sistema avviserà quando sostituire i dischetti alternativamente per leggere o scrivere.

I due dischi floppy devono essere dello stesso tipo.

2.7.16 CHKDSK

Controlla il disco e ne riassume i dati riguardanti lo stato.

Sintassi CHKDSK [unità:][[percorso]nomefile] [/F] [/V]

Parametri

unità: Specifica l'unità da controllare.

percorso Specifica la directory da controllare.

nomefile Il/i file di cui controllare la frammentazione.

Opzioni

/F Corregge gli errori trovati nel disco.

/V Visualizza il nome e il percorso completo di ogni file controllato.

Note

Digitare CHKDSK senza parametri per controllare il disco corrente.

2.7.17 MEM

Visualizza la quantità di memoria utilizzata e la quantità di memoria libera nel sistema. E' possibile utilizzare il comando MEM per visualizzare le informazioni relative alle aree di memoria utilizzate, alle aree di memoria libere ed ai programmi caricati in memoria.

Sintassi MEM [/C/F] [/P]

Per visualizzare lo stato della memoria utilizzata e della memoria libera del sistema, utilizzare la seguente sintassi: MEM

Opzioni

/C Elenca i programmi che sono stati caricati effettivamente nella memoria e visualizza la quantità di memoria convenzionale e di memoria superiore utilizzata da ogni programma. MEM /C fornisce anche un sommario dell'utilizzo di memoria ed elenca i blocchi di memoria più grandi a disposizione. E' possibile utilizzare l'opzione /C con /P, ma non con altre opzioni di MEM.

/F Elenca le aree di memoria convenzionale e superiore libere. MEM /F visualizza l'indirizzo del segmento, la dimensione di ogni area di memoria convenzionale libera e il blocco più grande libero in ogni area di memoria superiore. E' possibile utilizzare l'opzione /F con /P, ma non con altre opzioni di MEM.

/P Fa una pausa dopo ogni schermata di output.

Breve guida al sistema operativo MS-DOS

2.7.18 DATE

Visualizza la data o consente di reimpostarla se necessario.

MS-DOS registra la data di creazione o di modifica di ciascun file. La data viene visualizzata accanto al nome del file all'interno della directory.

Sintassi DATE [gg-mm-aa]

Parametro

gg-mm-aa Imposta la data specificata. I valori per giorno, mese ed anno devono essere separati tra loro da un punto (.), da un trattino (-), da una sbarra (/) o da uno spazio. I valori validi per le voci giorno, mese ed anno del formato gg-mm-aa sono i seguenti:

gg da 1 a 31
mm da 1 a 12
aa da 80 a 99 o 1980 a 2099

Esempio

Per impostare la data del sistema al 05/05/1998 digitare il comando: DATE 05-05-98

2.7.19 TIME

Visualizza l'ora di sistema oppure imposta l'orologio interno del computer.

MS-DOS utilizza le informazioni del comando TIME per aggiornare la directory tutte le volte che viene creato o modificato un file.

Sintassi TIME [ore:[minuti[:secondi[.centesimi]]]

Per visualizzare l'ora corrente utilizzare la sintassi seguente: TIME

Parametri

ore Specifica l'ora. I valori validi vanno da 0 a 23.
minuti Specifica i minuti. I valori validi vanno da 0 a 59.
secondi Specifica i secondi. I valori validi vanno da 0 a 59.
centesimi Specifica i centesimi di secondo. I valori validi vanno da 0 a 99.

Esempio

Per impostare l'orologio del sistema all'ora 13:36, digitare il comando: TIME 13.36

2.7.20 PATH

Visualizza o imposta il percorso di ricerca per i file eseguibili.

Sintassi PATH [[unità:]percorso[;...]]
PATH;

Parametri

unità: Specifica l'unità.
percorso Specifica il nome della directory che si desidera aggiungere come ulteriore percorso di ricerca per i file eseguibili

Breve guida al sistema operativo MS-DOS

Esempio

PATH C:\;C:\DOS;C:\NUOVO;C:\LINGUAGGI\BASIC

Digitare PATH; per cancellare le impostazioni del percorso di ricerca e indicare al DOS di cercare solo nella directory corrente.

Digitare PATH senza parametri per visualizzare il percorso di ricerca corrente.

2.7.21 CLS

Cancella lo schermo

Sintassi CLS

2.8 File batch

- Un programma batch è un file di testo, ASCII, che contiene una sequenza di comandi DOS
- I file batch hanno estensione .BAT
- I comandi DOS, all'interno dei programmi batch, sono eseguiti sequenzialmente

Quindi un programma batch è un file ASCII con una corrispondenza 1 a 1 fra riga e comando. Ciò vuol dire che ogni comando deve stare in una riga e che i comandi verranno eseguiti nell'ordine in cui sono stati inseriti. Il DOS riconosce come programma batch i file con estensione .BAT. Per eseguire un programma batch basta digitare il nome del file (anche senza estensione). Una volta eseguito il programma batch il controllo ritorna all'utente. Per creare il file batch si può usare l'EDIT del DOS. Nel nostro caso ad esempio potremo digitare.

C:\> EDIT SALVA.BAT

A questo punto basterà scrivere le linee:

COPY A:.* C:\LAVORO\OLD

COPY C:\LAVORO*.* A:\

e salvare il file.

Se al prompt digitiamo SALVA o SALVA.BAT verranno eseguiti i due comandi nella esatta sequenza con cui li abbiamo scritti nel programma. Si faccia presente che la sequenza può essere ripetuta tutte le volte che si vuole digitando solo un unico "comando" ed evitando errori dovuti a stanchezza o disattenzione. Si fa pure notare che il programma batch viene utilizzato allo stesso modo di un comando DOS non residente in memoria e come tale, per poter essere eseguito deve essere indicato il percorso o bisogna trovarsi nel direttorio che lo contiene.

In alcuni casi per l'esecuzione del programma batch risulta necessario interagire con l'operatore, permettendo la visualizzazione di messaggi. A tal fine il DOS ha introdotto tre comandi finalizzati all'interattività nella scrittura ed esecuzione di file batch:

REM inserisce dei commenti in un programma batch

PAUSE sospende temporaneamente (fino alla pressione di un tasto) l'esecuzione batch

ECHO visualizza messaggi sullo schermo

Breve guida al sistema operativo MS-DOS

Il comando REM si scrive all'inizio di una riga. In tal modo tale riga diventa di commento, il che vuol dire che non verrà utilizzata durante l'esecuzione. In questo modo è possibile inserire dei commenti utili a comprendere meglio il programma.

Il comando PAUSE viene utilizzato per sospendere momentaneamente l'esecuzione di un programma batch per permettere ad esempio di sostituire un disco. Per far ripartire il programma dalla riga immediatamente successiva al comando PAUSE basta premere un tasto.

Il comando ECHO invia a schermo alcuni messaggi atti a specificare azioni o ad informare l'utente durante l'esecuzione, ad esempio per la sostituzione di un disco prima del PAUSE specificando:

ECHO Inserire il disco

consente all'utente di evitare un errore. Si noti che ECHO va specificato a inizio riga, come REM. ECHO può anche essere utilizzato insieme ad OFF per disabilitare la visualizzazione dei comandi, ossia ECHO OFF evita che durante l'esecuzione batch vengano emessi a video i comandi scritti. Vediamo come potrebbe essere fatto il nostro programma batch SALVA.BAT:

ECHO OFF

REM Puliamo la schermata prima dell'esecuzione

CLS

REM Il programma SALVA.BAT consente il salvataggio del contenuto del floppy di
REM sicurezza nella directory C:\LAVORO\OLD e dei file della directory
REM C:\LAVORO in A:\

ECHO Il programma SALVA.BAT consente il salvataggio del contenuto del disco di
ECHO sicurezza nella directory C:\LAVORO\OLD e dei file della directory
ECHO C:\LAVORO in A:\

ECHO Per interrompere l'esecuzione digitare CTRL+C

ECHO Inserire il floppy di sicurezza

PAUSE

COPY A:.* C:\LAVORO\OLD

COPY C:\LAVORO*.* A:\

ECHO Fine lavoro

REM Fine programma

ECHO ON

Si noti le righe:

ECHO Per interrompere l'esecuzione digitare CTRL+C

PAUSE

In effetti per interrompere un programma batch bisogna digitare CTRL+C. Il sistema chiederà se continuare o interrompere la procedura batch. Digitando S seguito da INVIO il controllo ritornerà all'utente.

Come si può vedere l'ultima riga è ECHO ON il cui compito è quello di riabilitare l'echo dei comandi a video.

Si noti che quanto abbiamo fatto rende automatica la procedura ma non flessibile. Infatti in ogni caso copierà tutti i file da floppy a OLD e da LAVORO al floppy. Supponiamo che in realtà nel lavoro cambi solamente alcuni dei file e che tali file non siano sempre gli stessi. Per ottenere questa flessibilità si possono usare i *parametri sostituibili* o *variabili*.

Breve guida al sistema operativo MS-DOS

2.8.1 Parametri

- Un parametro è una sequenza di caratteri e/o cifre introdotte nella riga di comando dopo il nome del comando, ad esempio: COPY TONI VALERIO
- Un parametro sostituibile è uno dei simboli %0, %1, ..., %9 che viene gestito in un programma batch, ad esempio : COPY %1 %2
- Il numero che segue il carattere % specifica il numero d'ordine nella riga di comando : %0 è il file batch stesso, %1 è la prima opzione, %2 la seconda, ... e così via.

Abbiamo già parlato di parametro parlando dei comandi DOS. Abbiamo detto che alcuni comandi prevedono l'utilizzo di parametri. Ad esempio il comando COPY obbliga l'utilizzo di parametri, infatti è necessario specificare almeno un file di origine e/o di destinazione. Nell'ambito della programmazione batch si parla di parametri sostituibili utilizzando dei nomi simbolici che avranno significato in base al contesto. Ossia il valore che assumerà il parametro %2 non è definito a priori ma può essere scelto o variato in base ad un preciso meccanismo: la corrispondenza fra valore numerico del parametro sostituibile e parametro inputtato per eseguire il file batch. Supponiamo ad esempio che il programma di prima sia stato generalizzato usando le variabili e che in questo modo si possa non copiare necessariamente l'intera directory. Si potrà specificare anche un singolo file, ad esempio:

SALVA WORK.TXT

utilizzando una vera e propria sintassi che questa volta, a differenza dei comandi DOS, è gestita interamente da noi. Durante l'esecuzione del file batch il parametro %0 varrà ossia sarà equivalente a SALVA e %1 a WORK.TXT.

Quindi basterà riscrivere le righe :

```
COPY A:*. * C:\LAVORO\OLD
COPY C:\LAVORO\*.* A:\
```

con

```
COPY A:%1 C:\LAVORO\OLD
COPY C:\LAVORO\%1 A:\
```

Digitando SALVA WORK.TXT verrà salvato il file WORK.TXT.

Naturalmente non si possono utilizzare più di nove parametri. È anche possibile utilizzare parametri utilizzando caratteri jolly, ad esempio SALVA P*. ??T o ancora SALVA *.*.

In pratica durante l'esecuzione, tutte le volte che comparirà %1 verrà sostituito da, rispetto al primo caso, WORK.TXT e così via per gli altri parametri.

Supponiamo che anziché volere depositare nel direttorio OLD i vecchi file, li voglia scrivere su LAVORO ma con un estensione differente (ad esempio le prime tre lettere di ogni mese), basterà scrivere:

```
COPY A:%1 C:\LAVORO\*.*%2
COPY C:\LAVORO\%1 A:\
```

e digitare ad esempio SALVA WO*.TXT GEN

I file WO*.TXT contenuti in A: verranno salvati in LAVORO con estensione GEN e verranno sostituiti con quelli nuovi.

Breve guida al sistema operativo MS-DOS

Indice

1.	Cos'è un sistema operativo	1
2.	L'MS-DOS	2
2.1	Le componenti dell'MS-DOS	2
2.2	Il prompt e i dischi	3
2.3	File e nomi di file	3
2.4	Alberi di directory	4
2.5	I metacaratteri o caratteri jolly	5
2.6	Gestione della memoria nel DOS	5
2.7	Comandi principali	6
2.7.1	DIR	7
2.7.2	CD	7
2.7.3	MD	8
2.7.4	RD	8
2.7.5	COPY	8
2.7.6	XCOPY	9
2.7.7	DEL (Erase)	10
2.7.8	DELTREE	10
2.7.9	REN (Rename)	11
2.7.10	MOVE	11
2.7.11	TYPE	12
2.7.12	MORE	12
2.7.13	FORMAT	13
2.7.14	LABEL	14
2.7.15	DISKCOPY	14
2.7.16	CHKDSK	15
2.7.17	MEM	15
2.7.18	DATE	16
2.7.19	TIME	16
2.7.20	PATH	16
2.7.21	CLS	17
2.8	File batch	17
2.8.1	Parametri	19