

COSA E' L'INFORMATICA?

- Informatica come scienza della *rappresentazione* e dell'*elaborazione* dell'informazione
- L'elaborazione dell'informazione avviene in maniera sistematica e rigorosa, pertanto può essere automatizzata

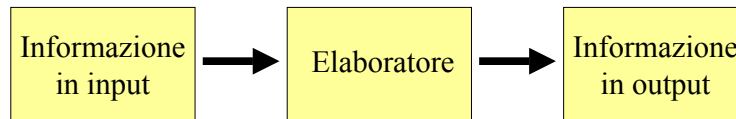
➔ *Informazione + automatica = informatica*

Il prodotto principale della tecnologia informatica e' l'informazione!

ELABORARE LE INFORMAZIONI

- L'informatica comprende lo *sviluppo* e l'*utilizzo* di strutture e di procedure per l'elaborazione delle informazioni
- L'informazione presentata al calcolatore viene trasformata in una forma diversa, generalmente di più facile utilizzo
 - la prima e' l'informazione in ingresso (*input*), o dato, ed e' definibile come la materia prima del processo
 - la seconda e' l'informazione in uscita (*output*), o prodotto finito

ELABORARE LE INFORMAZIONI



Unita' di informazione: il *bit* (0-1 sistema binario)

8 bit = 1 byte

1 Kbyte = 10^3 byte (1024 byte)

1 Mbyte = 10^6 byte (1.048.576 byte)

1 Gbyte = 10^9 byte (1.073.741.824 byte)

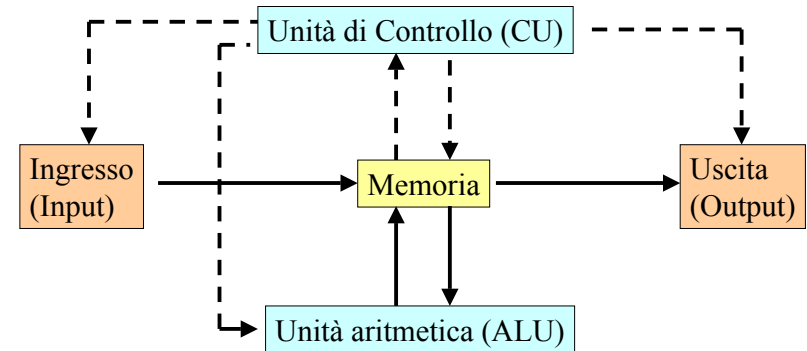
LA STRUTTURA INTERNA DI UN ELABORATORE

- Un elaboratore elettronico (calcolatore) e' un oggetto complesso, costituito da molte parti che interagiscono fra loro
- Due componenti fondamentali:
 - l'**hardware** (componenti fisici del sistema)
 - il **software** (programmi che vengono eseguiti dal sistema)
- Entrambe le componenti, mutuamente dipendenti, permettono l'elaborazione dei dati in ingresso in modo che possano essere generati dei dati in uscita

MACCHINA DI VON NEUMANN (1945)

- Costituisce ancor oggi un'ottima astrazione per descrivere lo schema architeturale di base della maggior parte dei calcolatori moderni
- E' una macchina astratta (non coincide esattamente con nessun calcolatore reale), ma descrive sinteticamente le caratteristiche essenziali di ognuno di essi prescindendo dalle peculiarità delle singole macchine fisiche

MACCHINA DI VON NEUMANN



UNITA' DI INGRESSO

- Comunica con il calcolatore (trasduttori + parte EO)
- Permette di:
 - introdurre dati
 - introdurre programmi
 - comunicare con dispositivi normalmente non compatibili con il calcolatore (funziona cioè da *interfaccia*)
- Esempi:
 - Tastiera, mouse, convertitori A/D, telecamere, microfoni, scanner, lettori di codici a barre, ...

UNITA' DI USCITA

- Sistema usato dal calcolatore per comunicare con il mondo esterno (parte EO + attuatori)
- Permette al calcolatore di:
 - comunicare con il mondo esterno attraverso dispositivi normalmente non compatibili con il calcolatore (funziona cioè da *interfaccia*)
- Esempi:
 - Stampante, monitor, plotter, altoparlanti, convertitori D/A, ...

UNITA' DI CONTROLLO

- E' responsabile del prelievo e della decodifica delle istruzioni nonché dell'invio dei segnali di controllo che provocano i trasferimenti o le elaborazioni necessarie per l'esecuzione dell'istruzione decodificata
 - Sequenzializzazione delle operazioni
 - L'orologio di sistema (**clock**) sincronizza le operazioni rispetto ad una data frequenza
- Si misura in cicli/secondo [Hz]
(400 MHz = 400 x 10⁶ Hz = 4 x 10⁸ Hz ⇒
1 ciclo e' eseguito in 2.5 x 10⁻⁹ s = 2.5ns)

UNITA' ARITMETICO-LOGICA

(Arithmetic Logic Unit, ALU)

- Comandata da segnali di controllo
- Esegue le operazioni aritmetiche e logiche eventualmente richieste per l'esecuzione dell'istruzione

LA MEMORIA CENTRALE

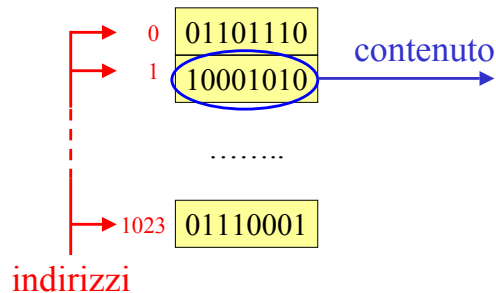
- Contiene le informazioni necessarie all'esecuzione di un programma, cioè **istruzioni e dati**
- Diversi tipi in dipendenza della capacita` di conservare le informazioni in caso di mancanza di alimentazione:
 - RAM (*random access memory*)
 - ROM (*read only memory*)
 - PROM (scritte una sola volta)
 - EPROM (scritte piu` volte)

LA MEMORIA CENTRALE

- E' un insieme di elementi di memoria di un bit aggregati a gruppi di otto (**celle o locazioni**) in grado di memorizzare un byte
- Ciascuna cella di memoria e' caratterizzata da:
 - **indirizzo fisico**: la sua posizione all'interno della memoria
 - **contenuto**: un numero (binario) fisicamente memorizzato al suo interno e che costituisce l'informazione

LA MEMORIA CENTRALE

- Il sistema (processore), attraverso l'indirizzo, può localizzare ciascuna cella e memorizzarvi o leggervi un dato/istruzione



- Essenziale la differenza tra contenuto e indirizzo

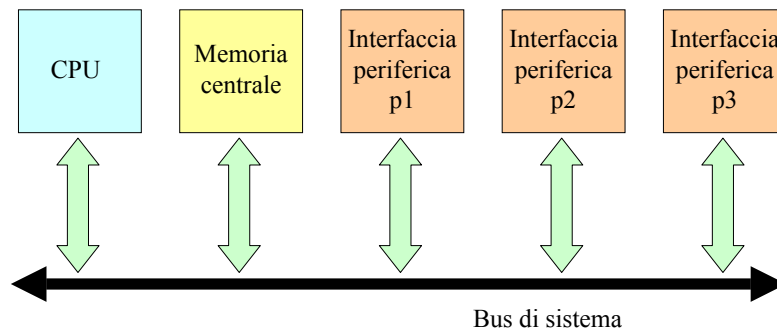
LA MEMORIA CENTRALE

Osservazioni:

- I moderni sistemi utilizzano celle di più byte contigui (**parole** o **word**) costituite in genere di 2, 4 o 8 byte, accedendo, in genere, ad un'intera word
- I dati sui quali devono essere fatti i calcoli vengono prelevati dalla memoria e collocati in particolari celle di memoria dette **registri** (residenti all'interno del processore)
 - nei registri le parti della CPU compiono i calcoli e scrivono i risultati

ORGANIZZAZIONE A BUS

- CPU (Central Processing Unit, CU+ALU)
- Il **bus di sistema** collega fra di loro la CPU, la memoria e le varie interfacce verso i dispositivi periferici

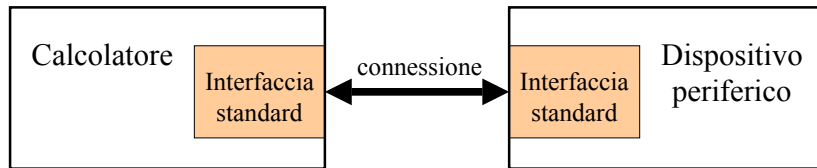


I DISPOSITIVI PERIFERICI

- Sono collegati al calcolatore tramite una circuiteria dedicata (*controller*)
- Generalmente contengono CPU dedicate, alloggiare fisicamente al loro interno
 - Se la CPU deve gestire anche i *dispositivi periferici* si ha un degrado delle prestazioni
- Nell'architettura di von Neumann, le periferiche includono anche le memorie di massa (dal punto di vista dell'interazione con il calcolatore si comportano come terminali, stampanti ecc.)

LE INTERFACCE

- I dispositivi periferici di diversi costruttori devono essere intercambiabili, senza modificare i programmi (blocchi indipendenti)
- Sono state definite alcune interfacce standard:



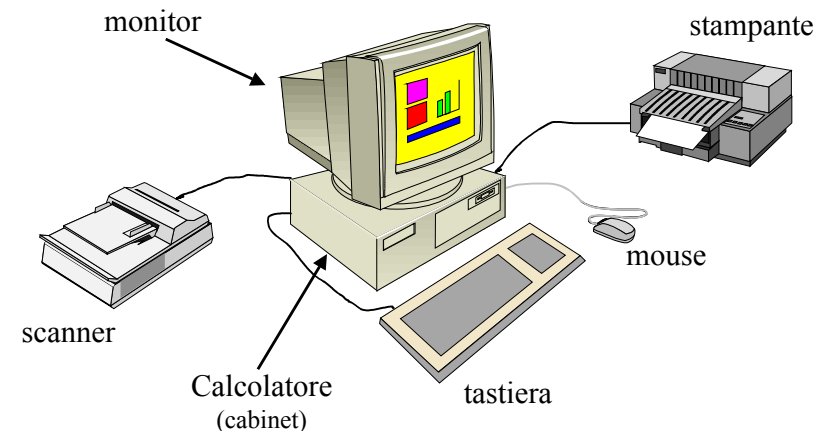
L'INTERFACCIA SERIALE

- E' l'interfaccia piu' comune e piu' semplice
- Il nome rispecchia la modalita' di comunicazione utilizzata (seriale)
- RS232, RS432, ... fissano gli standard (livelli di tensione, piedinatura, temporizzazione)
- L'intervallo tra due bit successivi e' costante (velocita' di trasmissione)
- Il parametro principale e' la *bit-rate*

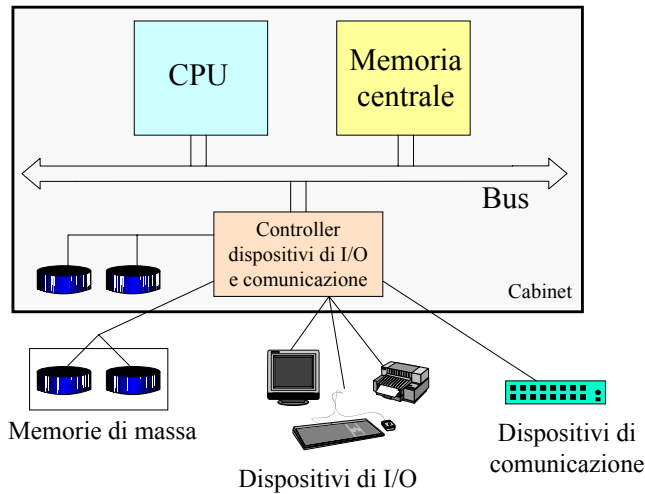
L'INTERFACCIA PARALLELA

- I bit vengono inviati tutti insieme, con l'aggiunta di segnali di controllo
- La comunicazione e' piu' veloce, ma occorrono piu' fili rispetto alla seriale
- Di conseguenza e' usata in modo *monodirezionale* (ad es. stampante, per distanze brevi)

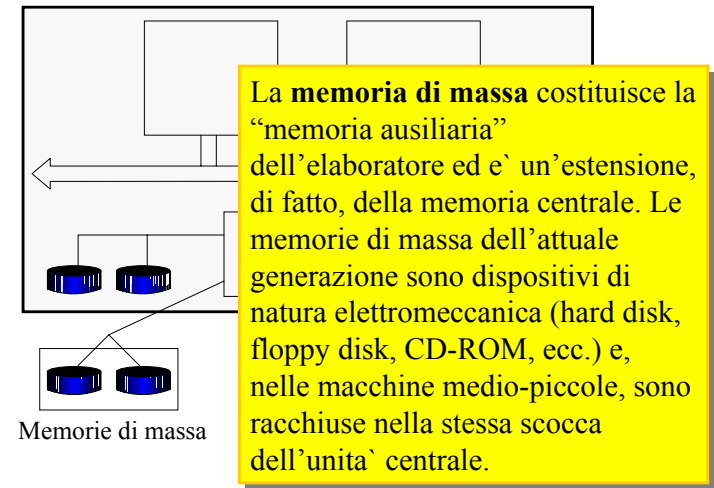
UN CASO REALE



UN CASO REALE

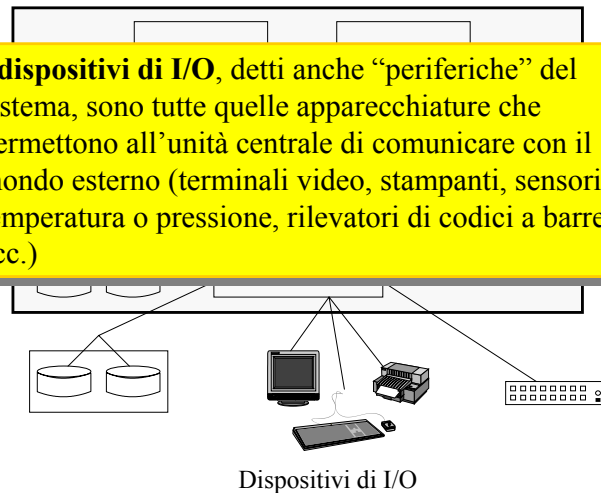


UN CASO REALE



UN CASO REALE

I **dispositivi di I/O**, detti anche "periferiche" del sistema, sono tutte quelle apparecchiature che permettono all'unità centrale di comunicare con il mondo esterno (terminali video, stampanti, sensori di temperatura o pressione, rilevatori di codici a barre, ecc.)



UN CASO REALE

I **dispositivi di comunicazione** consentono ai calcolatori lo scambio di informazioni con altri computer



SOFTWARE

Può essere classificato in due distinte categorie:

- **software di base** (o **software di sistema**):
 - dedicato alla gestione dell'elaboratore
 - fornisce un ambiente completo per la predisposizione e l'esecuzione delle elaborazioni
 - opera direttamente sull'hardware
- **software applicativo**:
 - comprende tutti i programmi utilizzati dagli *utenti* finali per la soluzione dei loro problemi specifici (battitura di testi, calcolo numerico, contabilità, disegno, ecc.)
 - opera al di sopra del software di base

SOFTWARE DI BASE

Comprende:

- **il sistema operativo (S.O.)**
 - = l'insieme dei moduli software per la gestione automatica del sistema
- **software di comunicazione**
 - = insieme di moduli software per la gestione delle trasmissioni dati e delle reti di calcolatori
- **strumenti di sviluppo**
 - = un insieme di moduli software per la preparazione, la verifica e l'esecuzione dei programmi

SISTEMA OPERATIVO

- Isola gli utenti dai dettagli dell'architettura hardware
- Fornisce un insieme di funzionalità di alto livello
- Esistono diversi tipi di S.O. classificati rispetto:
 - al numero di utenti
 - al numero di programmi in esecuzione, ovvero, più in generale, alle diverse tecniche di gestione delle risorse

GESTIONE UTENTI

- **Mono-utente**:
 - l'intero sistema (tipicamente un *Personal Computer*) e' dedicato ad un singolo utente
- **Multi-utente**:
 - diversi utenti condividono lo stesso sistema
 - il S.O. nasconde a ciascun utente la presenza degli altri, dando l'impressione che l'intero sistema (CPU, memoria, periferiche) gli sia interamente dedicato

GESTIONE DELLE RISORSE

• **Mono-programmazione**

- e' il modo più semplice per gestire un elaboratore
 - il S.O. gestisce in modo sequenziale i diversi programmi, uno per volta
 - poca efficienza dell'hw a causa della diversa velocità tra CPU e periferiche

• **Multi-programmazione**

- vengono gestiti simultaneamente più programmi indipendenti; il S.O. e' molto più complesso:
 - carica contemporaneamente più programmi
 - mentre uno e' in attesa, ne esegue un altro
 - la scelta dipende dalla priorità

ESEMPI

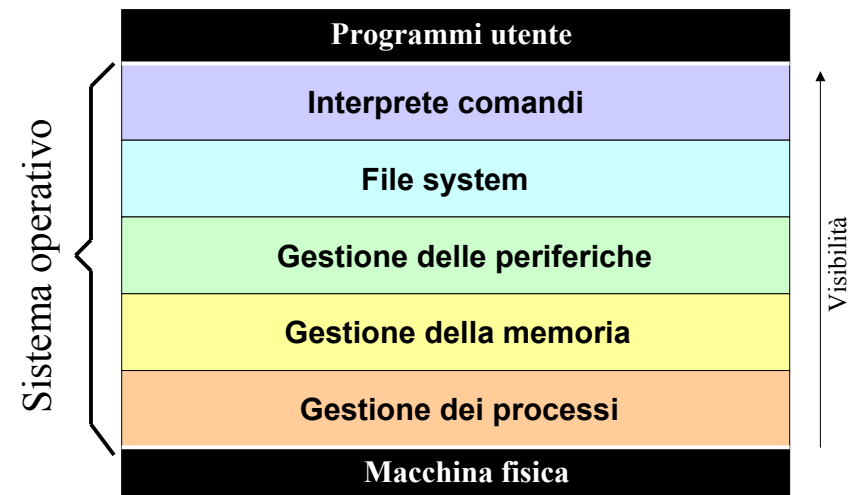
- **MS-DOS:** monoutente, monoprogrammato
- **Windows95, OS/2:** monoutente, multiprogrammato
- **Windows NT:** multiutente, multiprogrammato
- **UNIX:** multiutente, multiprogrammato

SISTEMA OPERATIVO

- Un utente "vede" l'elaboratore solo tramite il sistema operativo → **il S.O. realizza una "macchina virtuale"**
 - diversi S.O. possono realizzare diverse macchine virtuali *sullo stesso elaboratore fisico*
- Attraverso il S.O. il livello di interazione utente/elaboratore viene elevato:
 - senza S.O.: sequenze di bit
 - con S.O.: comandi, programmi, dati

I sistemi operativi si sono evoluti nel corso degli ultimi anni (interfacce grafiche, Macintosh, Windows, Linux, ...)

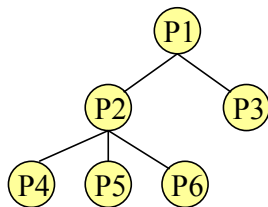
FUNZIONI DEL SISTEMA OPERATIVO



GESTIONE DEI PROCESSI

- Il *processo* è un concetto chiave in tutti i sistemi operativi
- E' essenzialmente un *programma in esecuzione*
 - si tratta di un oggetto dinamico, che evolve nel tempo, contrapposto a un programma, che e' un oggetto statico e invariante nel tempo

Un processo può creare uno o più processi figli e questi altri processi figli, ...



GESTIONE DELLA MEMORIA

- Stabilisce il complesso di regole che governano la ripartizione dello spazio disponibile tra piu' entita' che, in modo concorrente, ne richiedono l'utilizzo (ad es:
 - il sistema operativo
 - i buffer I/O
 - i processi generati
- I criteri di allocazione della memoria centrale hanno un notevole impatto sulle funzionalita' e sulle prestazioni globali dei sistemi, in particolare in quelli multitasking.

GESTIONE DELLE PERIFERICHE

- Offre all'utente una visione astratta delle periferiche in cui le loro caratteristiche hw vengono mascherate
- Mette a disposizione un insieme di procedure standard di alto livello, che leggono dati in ingresso e scrivono dati in uscita
 - ▶ I programmi di sistema che permettono di eseguire operazioni di I/O per uno specifico componente vengono detti **driver**

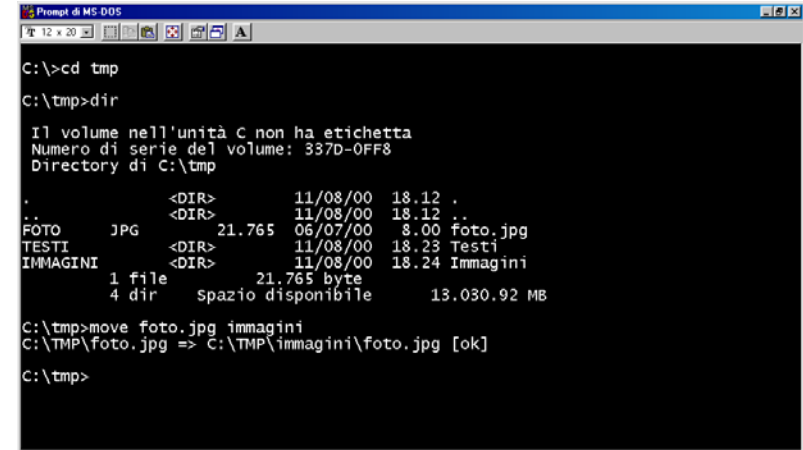
FILE SYSTEM

- L'informazione contenuta nella memoria di massa e' organizzata in **file**
 - un file e' dotato di un proprio nome
 - contiene dati in diversi formati
 - essenzialmente "testo" o "binario"
 - programmi eseguibili
 - immagini,
 - suoni,
- Il file system
 - consente l'accesso a file attraverso i nomi simbolici
 - gestisce i diritti di accesso e le protezioni (nei sistemi multi-utente)

INTERPRETE COMANDI

- L'interprete dei comandi (**shell**) è un particolare processo che consente all'utente di impartire comandi e mandare in esecuzione programmi: e' *l'interfaccia principale* tra l'utente ed il S.O.!!
- Il processo *shell* viene automaticamente creato all'atto della connessione dell'utente al sistema e rimane attivo fino al termine della sessione di lavoro
- Le *shell* possono essere: command oriented
 grafiche

SHELL - MS-DOS



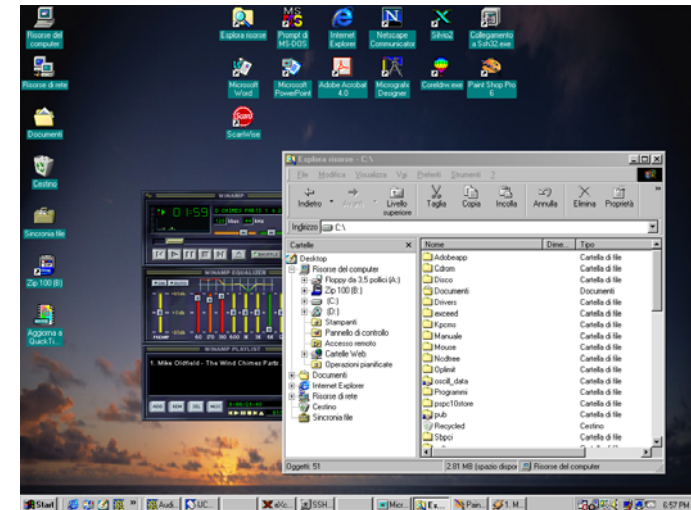
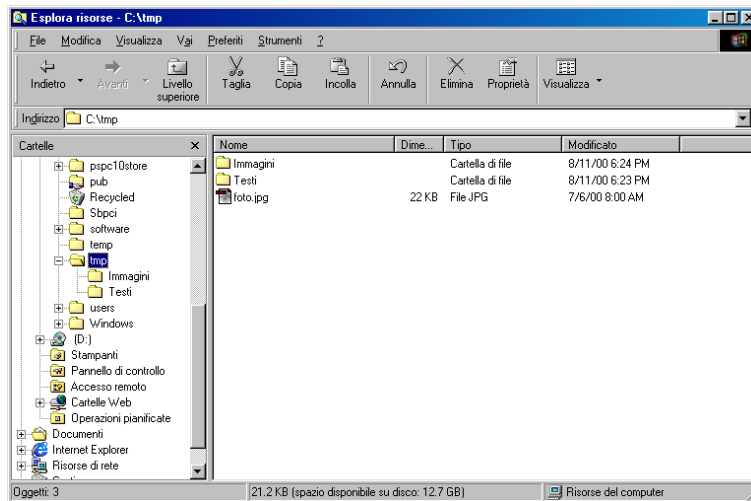
```
Prompt di MS-DOS
C:\>cd tmp
C:\tmp>dir

Il volume nell'unit  C non ha etichetta
Numero di serie del volume: 337D-0FF8
Directory di C:\tmp

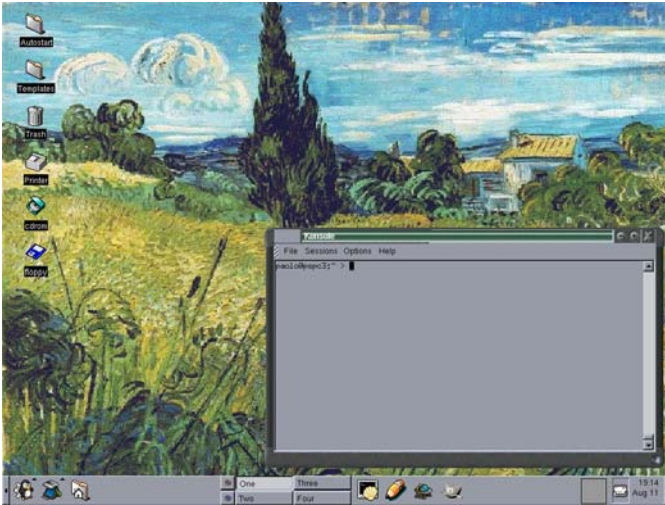
.                <DIR>          11/08/00   18.12 .
..               <DIR>          11/08/00   18.12 ..
FOTO             JPG           21.765   06/07/00   8.00 foto.jpg
TESTI            <DIR>          11/08/00   18.23 Testi
IMMAGINI         <DIR>          11/08/00   18.24 Immagini
1 file           21.765 byte
4 dir            spazio disponibile 13.030.92 MB

C:\tmp>move foto.jpg immagini
C:\TMP\foto.jpg => C:\TMP\immagini\foto.jpg [ok]
C:\tmp>
```

SHELL - Windows



SHELL - Linux



FASE DI "BOOT"

- Ad ogni accensione del sistema il S.O. deve essere ricaricato
- Il programma di **boot** copia in memoria dal disco fisso l'interprete dei comandi e il modulo di gestione dei dispositivi periferici
 - L'hw è progettato in modo tale che, ogni volta che la macchina viene accesa, il programma di boot venga automaticamente caricato ed eseguito
 - Il boot consta solo di poche istruzioni, tuttavia sufficienti per leggere il resto del S.O.

SOFTWARE DI COMUNICAZIONE

- Insieme dei programmi che supportano la comunicazione tra macchine collegate in rete
- La comunicazione avviene utilizzando protocolli che garantiscono un corretto scambio di dati e messaggi

Esempio:

- rete **Internet** (protocollo TCP/IP):
 - telnet:* apertura di un terminale remoto
 - ftp:* trasferimento di file
 - mail:* posta elettronica
 - http:* World Wide Web

STRUMENTI DI SVILUPPO

(premessa)

- Il calcolatore fornisce risorse (hw) di calcolo e, più in generale, di elaborazione dell'informazione
- il S.O. fornisce un'interfaccia verso tali risorse
 - rende trasparente il loro accesso
 - realizza una macchina virtuale
- L'utente è in condizioni di:
 - verificare che il calcolatore funzioni
 - utilizzare strumenti virtuali, rappresentati da applicativi commerciali
 - **utilizzare le risorse del calcolatore per la risoluzione di problemi specifici**

STRUMENTI DI SVILUPPO

(premessa)

- La specificazione delle procedure per la risoluzione del problema e la “comunicazione” di tali procedure al calcolatore rappresentano lo scopo della **programmazione**
- I **programmi** rappresentano una descrizione delle procedure necessarie alla risoluzione del problema in un linguaggio comprensibile al calcolatore:

- Linguaggio macchina
- Linguaggio d’alto livello

```
00101100  
1001...  
0010101...
```

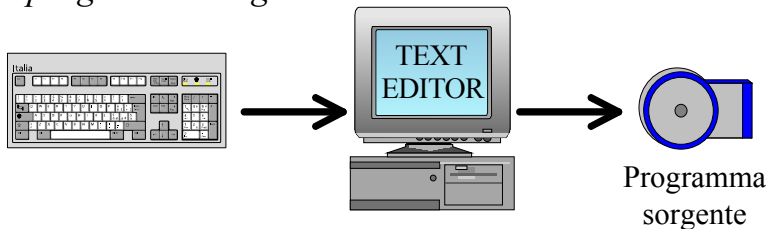
```
main(){  
int i;  
for(i=1;i<N;i++)  
A[i]=B[N-i];...
```

STRUMENTI DI SVILUPPO

- Insieme di programmi che consentono la scrittura, la verifica e l’esecuzione di nuovi programmi per applicazioni specifiche
 - **Editor**
 - **Compilatore**
 - **Interprete** } **Traduttori**
 - **Linker** (collegatore)
 - **Debugger** (controllore dell’esecuzione)

EDITOR

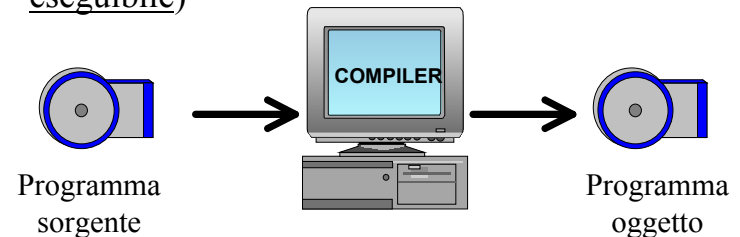
- Serve per creare file che contengono testi (sequenze di caratteri). In un ambiente di programmazione l’**editor** consente di scrivere il *programma sorgente*



```
edit nome.c → nome.c
```

COMPILATORE

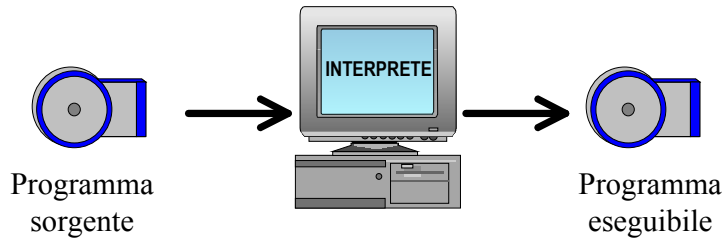
- Opera la traduzione di un programma sorgente scritto in un linguaggio ad alto livello in un *programma oggetto* (file binario non ancora eseguibile)



```
cl -c nome.c → nome.obj
```

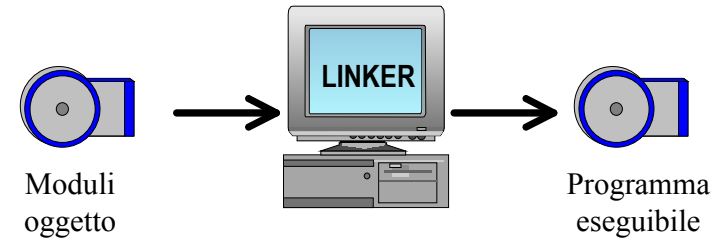
INTERPRETE

- Traduce ed esegue direttamente ciascuna istruzione del programma sorgente. Ad ogni attivazione dell'interprete su una particolare istruzione segue l'esecuzione dell'istruzione stessa
 - in alternativa al compilatore
 - esecuzione più lenta rispetto a quella di un programma compilato



LINKER

- Crea da un programma oggetto un *programma eseguibile*, eventualmente collegando (unendo) più moduli (compilati separatamente)



```
cl nome.obj -----> nome.exe
```

DEBUGGER

- Usando il debugger, il programma viene controllato nel corso della sua esecuzione
 - break-point
 - dump
 - trace
 - ...
- Questo controllo aiuta a scoprire ed eliminare errori non rilevati in fase di compilazione