Università degli Studi di Palermo
Dipartimento di Ingegneria Informatica

C.I. 3 – Modulo "Informatica"
2 c.f.u.

Anno Accademico 2009/2010

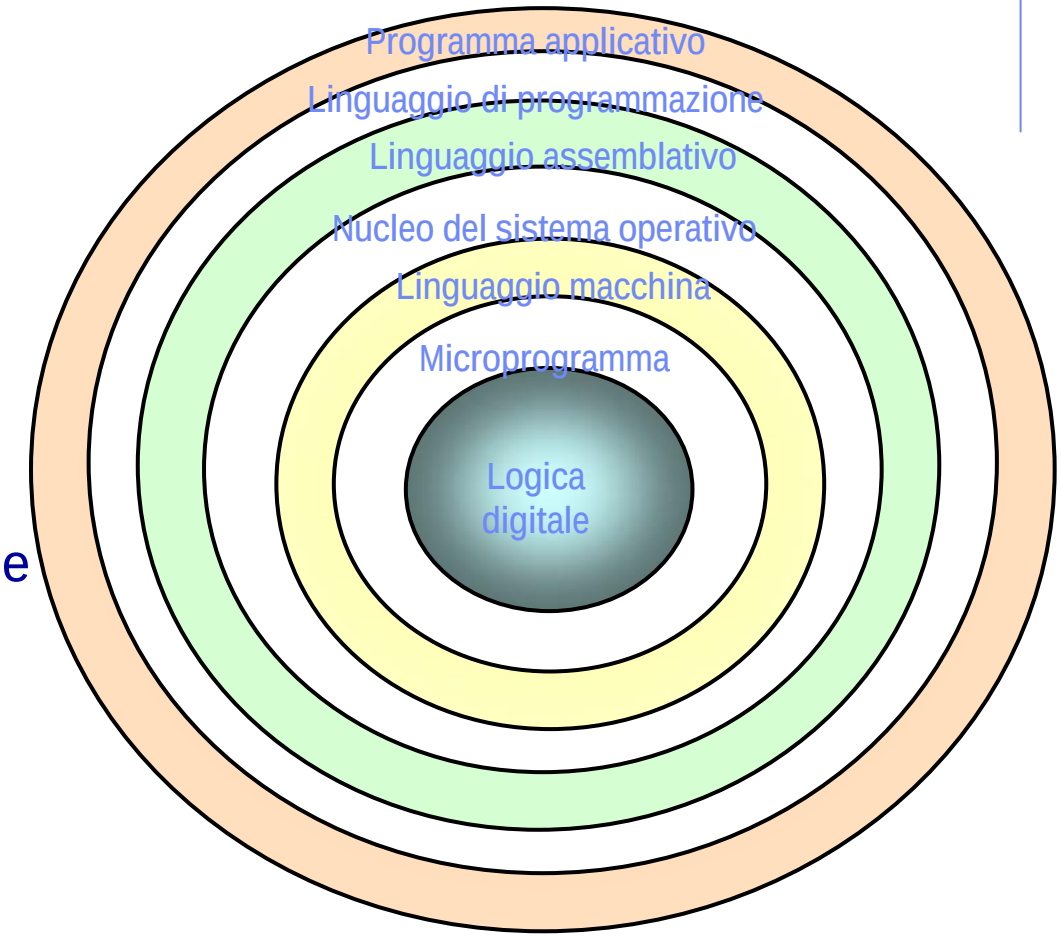
Docente: ing. Salvatore Sorce

Il Sistema Operativo

Gerarchia del software

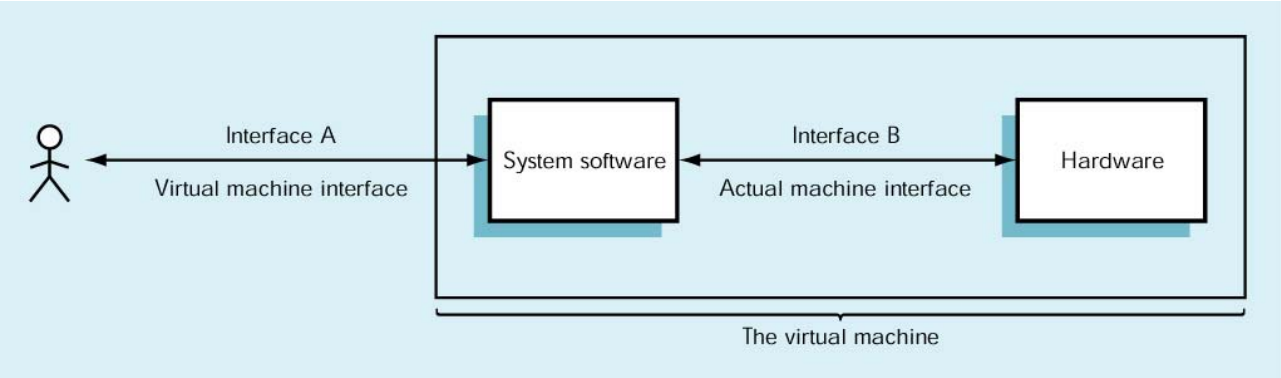
- Sei livelli di astrazione separano l'utente dall'hardware sottostante

- Microprogramma
- Linguaggio macchina
- Sistema operativo
- Linguaggio assemblativo
- Linguaggio di programmazione
- Programma applicativo



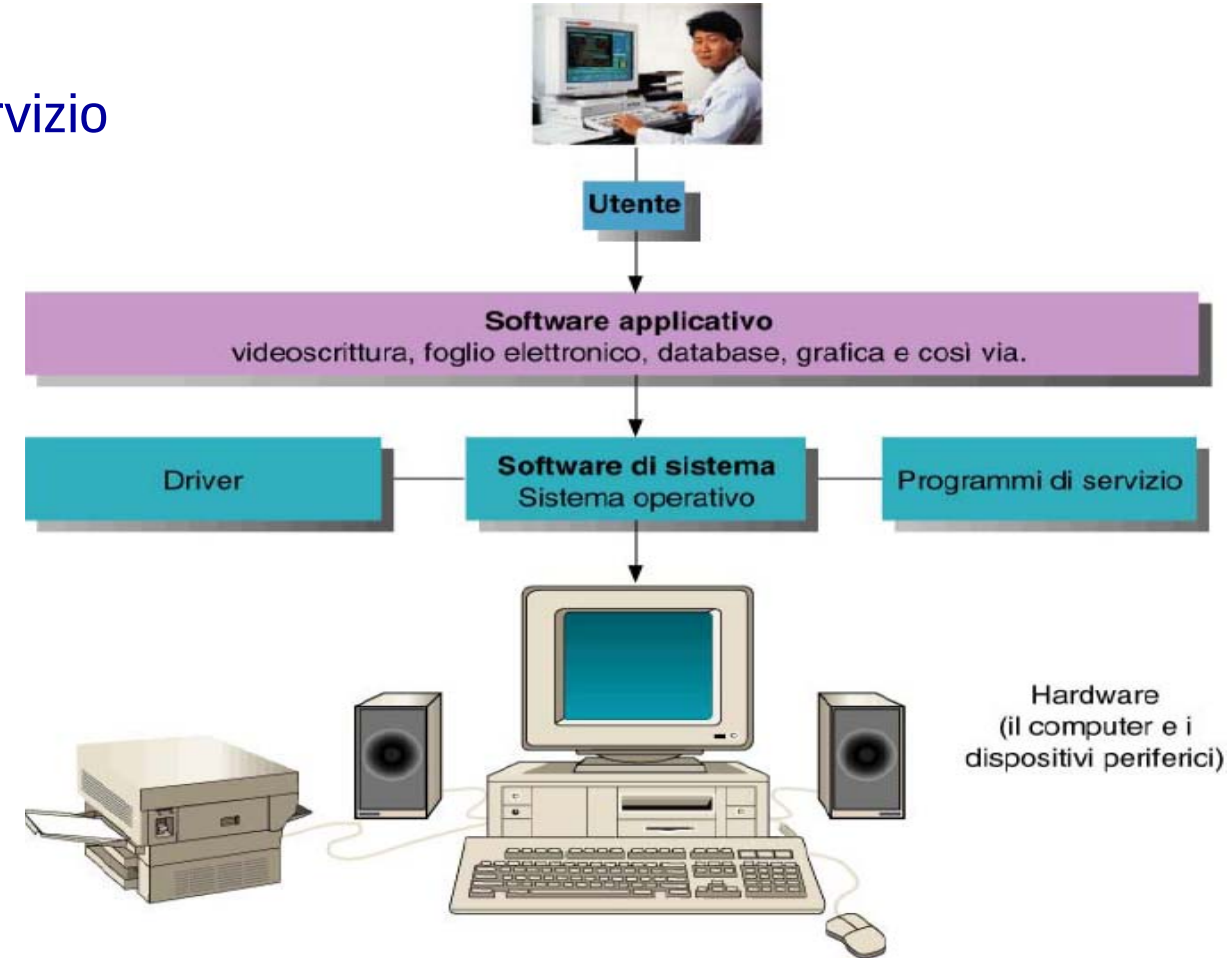
Software di sistema

- Software di sistema
  - Raccolta di programmi per la gestione delle risorse di un calcolatore e della loro accessibilità
  - Agisce da intermediario tra utente e hardware
- Macchina virtuale
  - Insieme dei servizi e delle risorse generate dal sw di sistema
- Il software di sistema è l'analogo del cruscotto per una macchina di Von Neumann



Software di sistema

- Sistemi operativi
  - Windows, DOS, Unix/Linux, Mac OS
- Driver
- Programmi di servizio



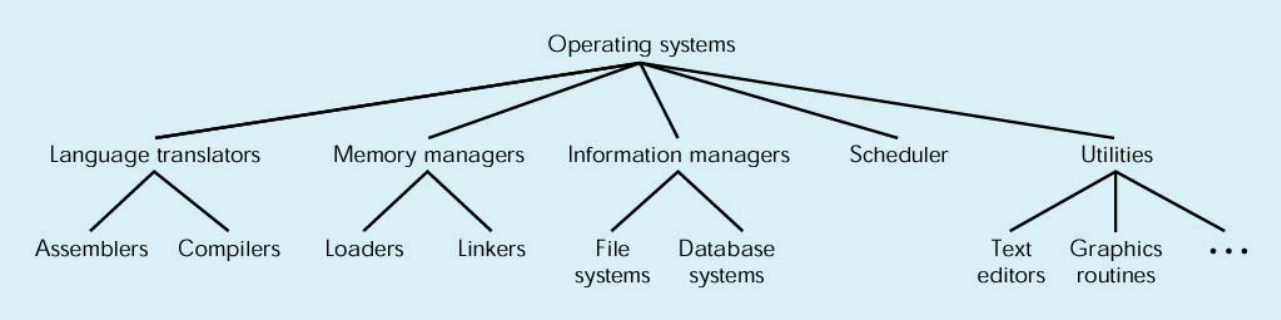
Università degli Studi di Palermo

Interfaccia tra hardware e software

- Nascondere all'utente i dettagli non necessari dell'hardware
- Presentare le informazioni
- Consentire all'utente un facile accesso alle risorse macchina disponibili
- Prevenire danni accidentali o intenzionali ad hardware, programmi e/o dati

Classificazione del software di sistema

- Sistema operativo
  - Programma che supervisiona tutte le operazioni di un calcolatore
  - Comunica con l'ambiente esterno, gestisce l'attivazione di periferiche e altre componenti sw
- Classi di programmi di sistema
  - Traduttori
  - Gestori della memoria
  - File system
  - Schedulatore
  - Programmi di utilità



Classificazione del software di sistema

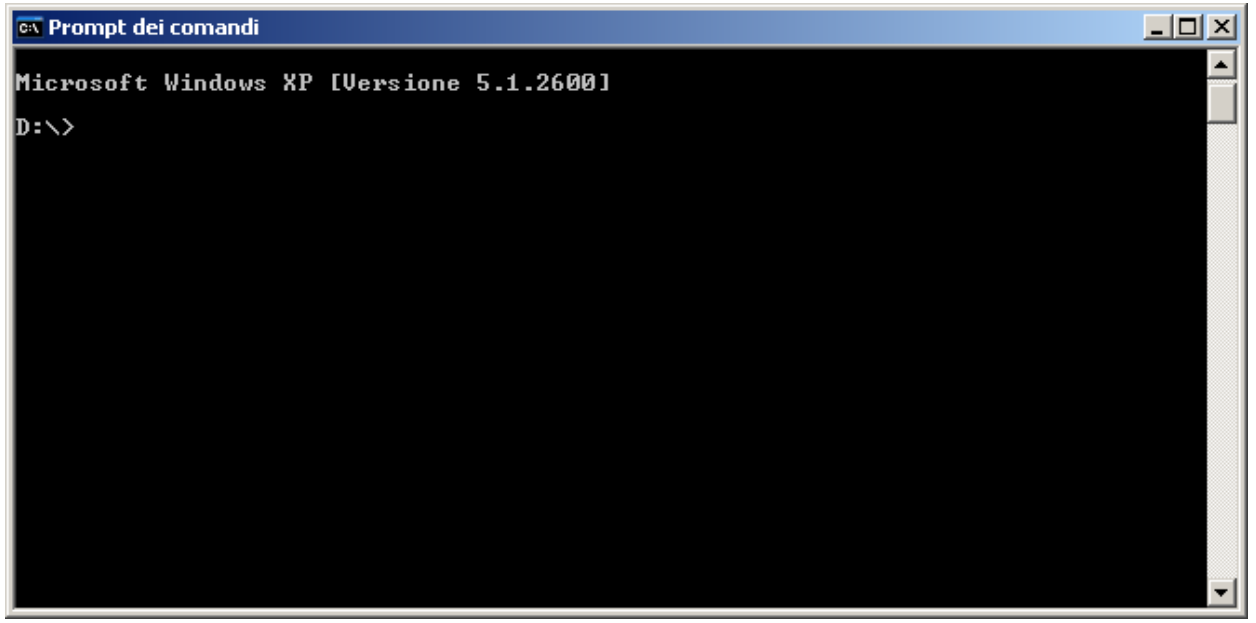
- Traduttori
  - Assemblatori, compilatori ed interpreti
  - Consentono di descrivere algoritmi in un linguaggio orientato all'utente
- Gestori della memoria
  - Riservano spazio in memoria per dati e programmi
  - Caricano in memoria i programmi prima dell'esecuzione
- File system
  - Gestiscono la memorizzazione e il recupero di informazioni sui dispositivi di memoria di massa
- Schedulatore
  - Gestisce l'elenco con priorità dei programmi pronti per l'esecuzione
  - Seleziona il programma prossimo da eseguire (prioritarizzazione)
- Programmi di utilità
  - Librerie di programmi che forniscono servizi sia all'utente che ad altri programmi

Interfacce utenti

- Implementano i comandi del SO
- Interfacce testuali
  - Uso di un linguaggio di comandi immessi come testo da tastiera

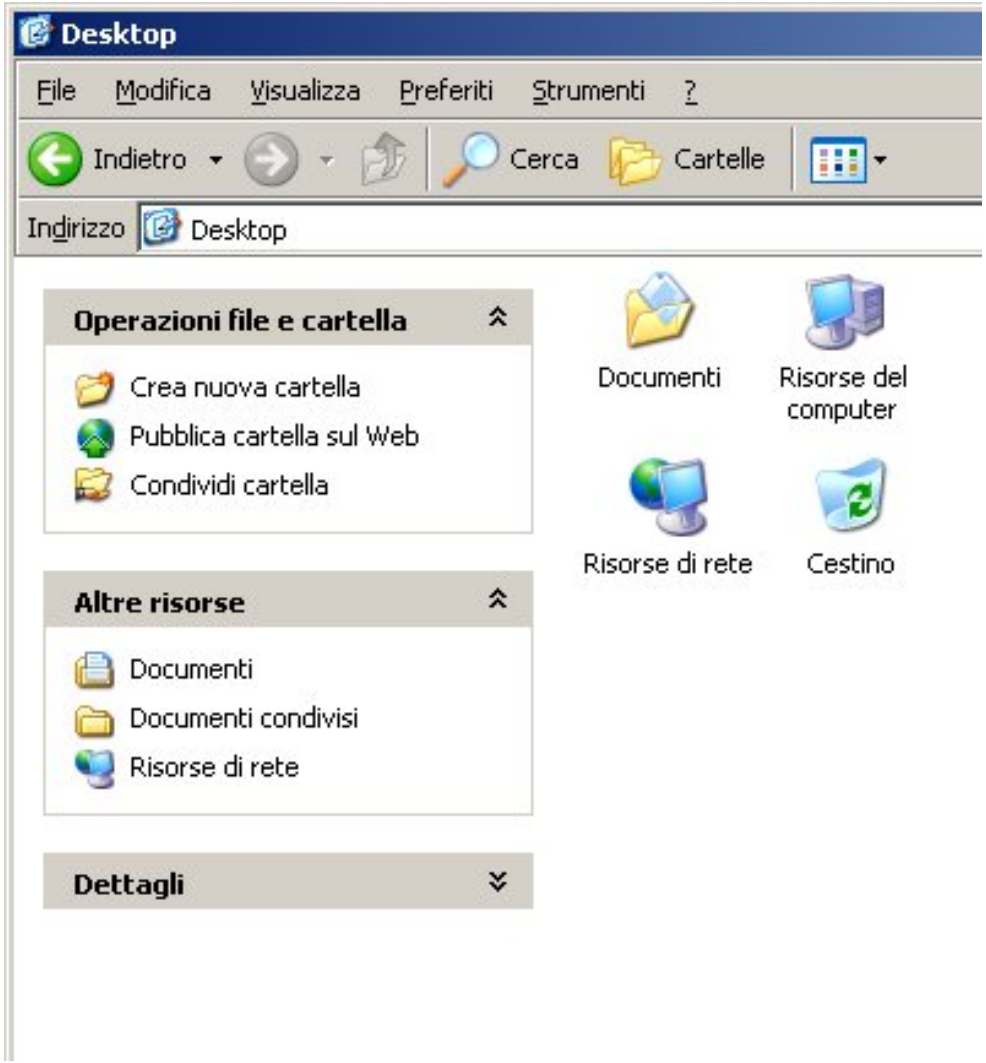
```
* * * * * * * * * * * * * * * * * * * * * * * * *
*                                                 *
*                                                 *
*  Welcome to Apollo ...                          *
*                                                 *
*  Macalester College AlphaServer 2000            *
*                                                 *
*                                                 *
* * * * * * * * * * * * * * * * * * * * * * * * *

User Name:  Schneider
Password:  XXXXX     (Blocked out for security reasons)
$     ($ is the prompt. The system is now waiting for a user request.)
```



Interfacce utenti

- Interfacce grafiche (GUI)
  - Uso di una metafora (desktop)
  - Ai comandi testuali sono sostituiti icone, menu, finestre e le azioni che possono essere eseguite su di essi



File system

- Esistono diversi tipi di supporti per la memorizzazione permanente delle informazioni: dischi magnetici (floppy disk, hard disk), dischi ottici (cd), nastri magnetici
- Un *file* è un insieme di byte che rappresentano una certa entità logica (testo, immagine, suono, programma, etc), organizzati secondo un certo formato, memorizzati su supporti di memoria secondaria.

Università degli Studi di Palermo

File system

- Il **File System** è quella parte del S.O. che si occupa di gestire e strutturare le informazioni memorizzate su supporti permanenti
- Il sistema operativo deve fornire una visione **astratta** (semplificata) dei file su disco e l'utente deve avere la possibilità di:
  - identificare ogni file con un nome (**filename**) astraendo completamente dalla sua memorizzazione fisica (blocchi su disco rigido e localizzazione dei blocchi)
  - avere un insieme di operazioni per lavorare sui file: creare o rimuovere un file, copiarlo, cambiargli nome, inserire informazioni in un file
  - effettuare l'accesso alle informazioni mediante operazioni ad alto livello, che non tengono conto del tipo di memorizzazione (accedere ad un file memorizzato sul disco rigido oppure su un CD-ROM allo stesso modo)

(segue ...)

File system

(... segue)

- avere la possibilità di strutturare un insieme di file, organizzandoli in sottoinsiemi secondo le loro caratteristiche, per avere una visione ordinata e strutturata delle informazioni sul disco
- in un sistema multi-utente, inoltre l'utente deve avere meccanismi per proteggere i propri file, ossia per impedire ad altri di leggerli, scriverli o cancellarli

➢ i moderni sistemi operativi forniscono supporto per queste attività

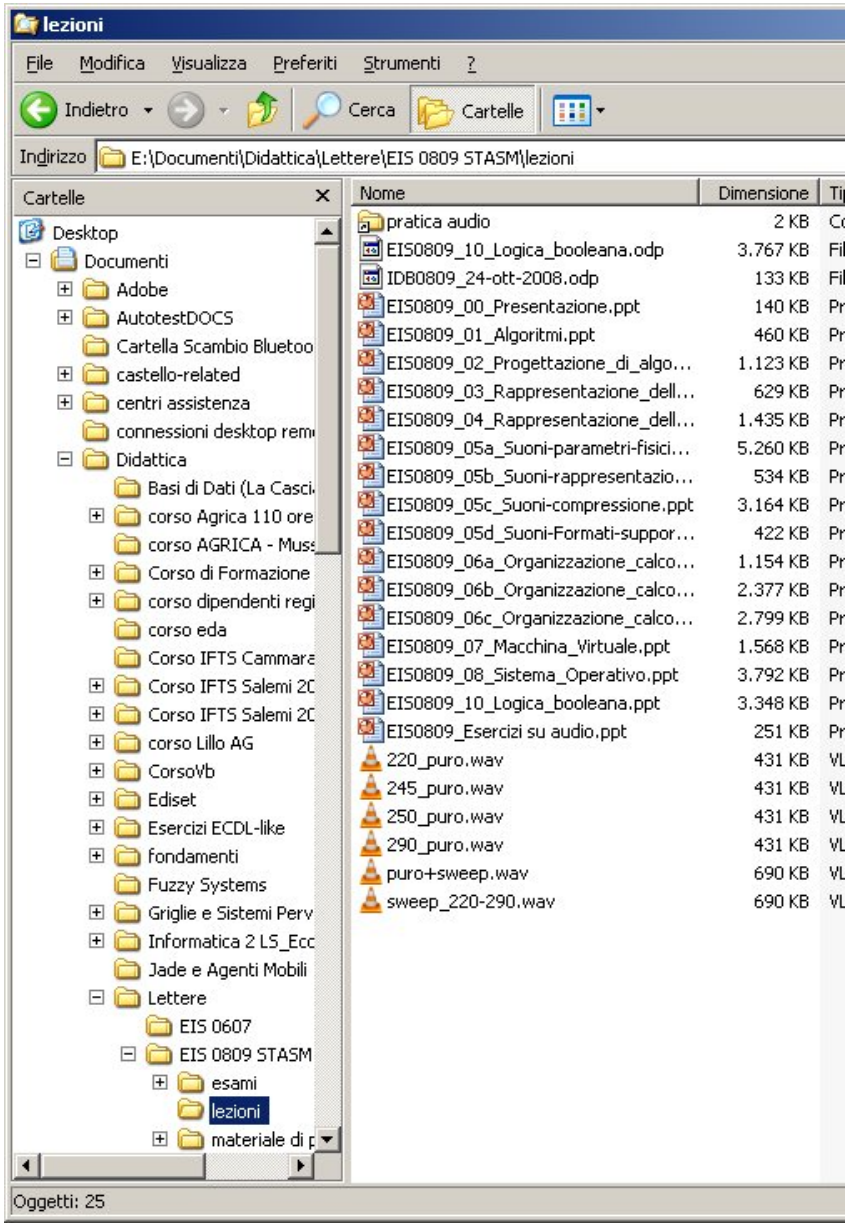
Università degli Studi di Palermo

File system

- Il file system deve tenere traccia di tutte le caratteristiche di file e sottoinsiemi di file (il nome, la dimensione, quali sono gli indirizzi dei blocchi sui quali sono memorizzati, etc.)
- Dove sono memorizzate queste informazioni?
- Una parte del disco rigido (un sottoinsieme di tracce) è riservato al sistema operativo per questi (ed altri) scopi
- Esempio: FAT (File Allocation Table)
  - Contiene le corrispondenze <nome file> → <blocco di inizio>
  - Settori concatenati: <1° blocco file> → <2° blocco file> → ...
- N.B. Anche una parte della memoria centrale (RAM) è riservata alla memorizzazione del sistema operativo

File system

- Presentazione dei file all'utente
  - Directory (cartelle)
  - Organizzazione gerarchica ad albero
  - Nomi dei files
  - Percorsi

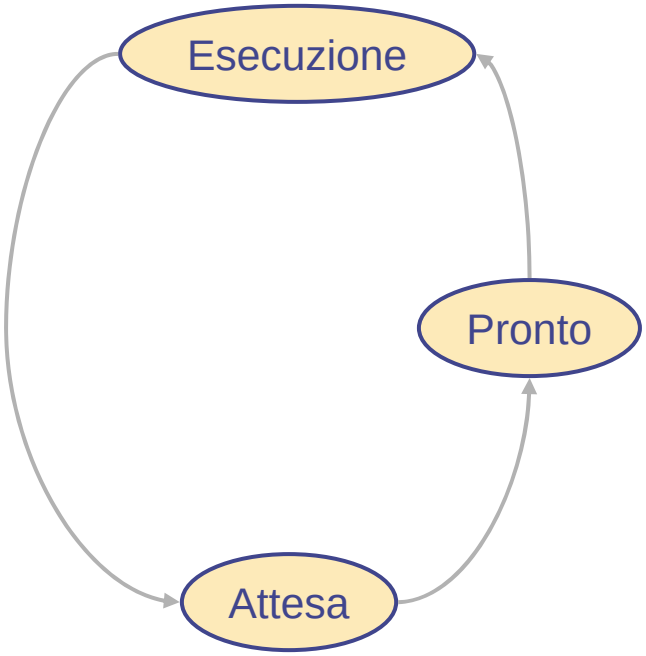


Allocazione efficiente delle risorse

- Esiste una notevole differenza nella velocità di esecuzione di operazioni tra processore e unità di I/O
- Il SO deve assicurare che il processore rimanga inutilizzato il minor tempo possibile
- Tanti programmi in esecuzione, ma un solo processore: quasi parallelismo
- Il SO mantiene una coda di programmi in esecuzione dei quali solo uno è attivo per ogni istante di tempo

Allocazione efficiente delle risorse

- stati di un programma
  - In esecuzione
    programma attualmente in esecuzione
  - Pronto
    programmi in memoria e pronti per l'esecuzione, ordinati per priorità
  - Attesa
    programmi che non possono essere eseguiti perché in attesa del completamento di una operazione di I/O



Allocazione efficiente delle risorse

- Quattro programmi, A, B, C, D
  - A in esecuzione
  - B, C, D pronti per passare in esecuzione

Attesa	Pronto	Esecuzione
	B	A
	C	
	D	

Università degli Studi di Palermo

Allocazione efficiente delle risorse

- Quattro programmi, A, B, C, D
  - A in esecuzione
  - B, C, D pronti per passare in esecuzione
- A inizia una operazione di I/O
  - A passa in attesa e B va in esecuzione

Attesa	Pronto	Esecuzione
A	C	B
	D	

Allocazione efficiente delle risorse

- Quattro programmi, A, B, C, D
  - A in esecuzione
  - B, C, D pronti per passare in esecuzione
- A inizia una operazione di I/O
  - A passa in attesa e B va in esecuzione
- B inizia una operazione di I/O
  - B passa in attesa e C va in esecuzione

Attesa	Pronto	Esecuzione
A	D	C
B		

Allocazione efficiente delle risorse

- Quattro programmi, A, B, C, D
  - A in esecuzione
  - B, C, D pronti per passare in esecuzione
- A inizia una operazione di I/O
  - A passa in attesa e B va in esecuzione
- B inizia una operazione di I/O
  - B passa in attesa e C va in esecuzione
- A completa l'operazione
  - Passa in pronto. Se ha priorità superiore a D, potrebbe scavalcarlo

Attesa	Pronto	Esecuzione
B	D	C
	A	

Allocazione efficiente delle risorse

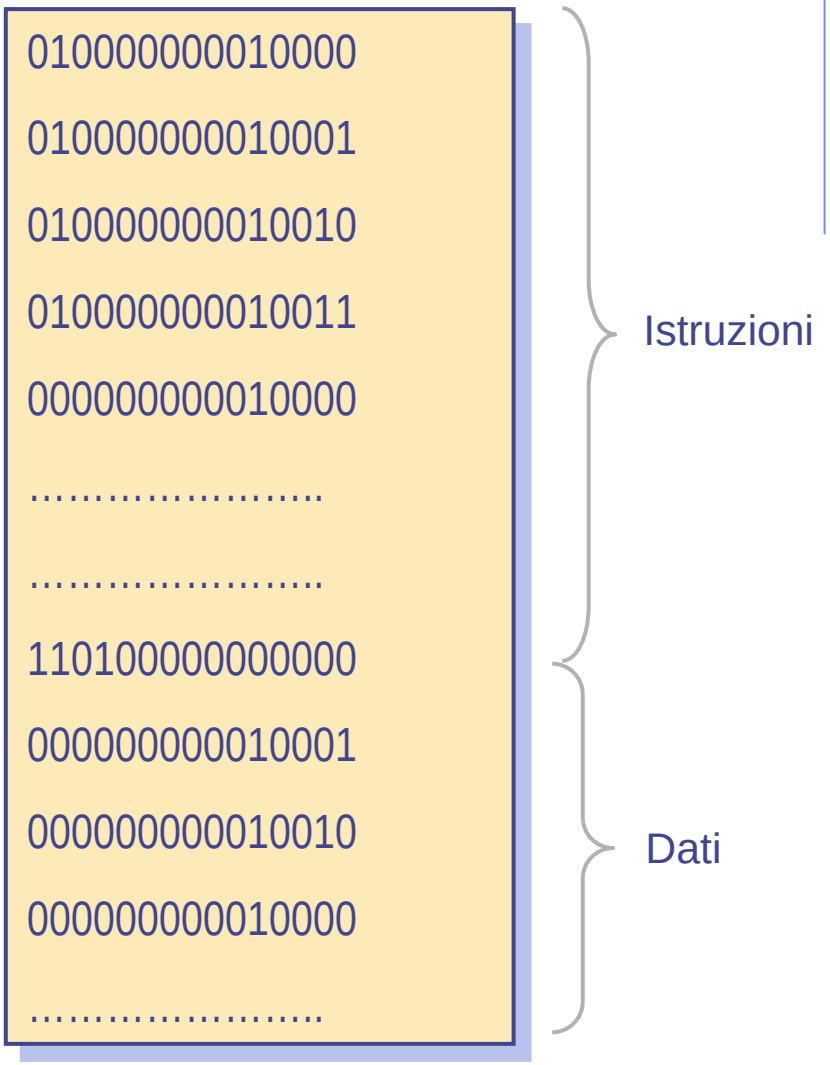
- Quattro programmi, A, B, C, D
  - A in esecuzione
  - B, C, D pronti per passare in esecuzione
- A inizia una operazione di I/O
  - A passa in attesa e B va in esecuzione
- B inizia una operazione di I/O
  - B passa in attesa e C va in esecuzione
- A completa l'operazione
  - Passa in pronto. Se ha priorità superiore a D, potrebbe scavalcarlo

Attesa	Pronto	Esecuzione
B	A	C
	D	

Il programma

- Il programma è costituito da una sequenza di istruzioni caricate nella memoria centrale sotto forma di sequenze di bit

- Esecuzione di un'istruzione:
  - Fase di acquisizione (Fetch)
  - Interpretazione (Decode)
  - Esecuzione (Execute)



Linguaggio macchina

- Linguaggio macchina
  - Formato binario. Le istruzioni sono indistinguibili dai dati su cui operano
  - Non consente l'uso di etichette o simboli per indicare locazioni di memoria o istruzioni adibite a compiti specifici
  - Difficile da modificare. Gli indirizzi delle istruzioni si susseguono sequenzialmente a partire dalla prima.
  - Difficile creare dati. I dati possono solo essere rappresentati nel loro formato interno
- I calcolatori della prima generazione potevano essere programmati soltanto in linguaggio macchina!

Università degli Studi di Palermo

Linguaggio assembler

- Linguaggio assembler
  - Orientato sia alla macchina che all'utente
  - Linguaggio di seconda generazione, contrapposto al linguaggio macchina o di prima generazione
- Le istruzioni sono indicate con etichette comprensibili che vengono tradotte nel codice binario corrispondente dal traduttore
- Codici mnemonici
  - ADD – addizione
  - SUB – sottrazione
  - LOAD, STORE – carica da memoria, memorizza in memoria
  - JUMP – salta ad istruzione successiva
- Rapporto 1:1 con il linguaggio macchina
  - Ogni istruzione in linguaggio assembler è tradotta esattamente nella sua corrispondente in linguaggio macchina
  - Specifico per una particolare classe di microprocessori

Università degli Studi di Palermo

Caratteristiche del linguaggio assembler

- Vantaggi rispetto al linguaggio macchina
  - Uso di codici operativi simbolici (mnemonici) anziché numerici
  - Uso di indirizzi di memoria simbolici anziché numerici
  - Pseudo-operazioni che forniscono servizi all'utente, come la generazione di dati
- Formato tipico di una istruzione

  etichetta: mnemonico campo_indirizzo -- commento
- Caratteristiche aggiuntive
  - Chiarezza dei programmi
  - Manutenibilità

Esempio di file oggetto

Indirizzo	Opcode data	Significato
0000	1101 000000001001	IN X
0001	1101 000000001010	IN Y
0010	0000 000000001001	LOAD X
0011	0111 000000001010	COMPARE Y
0100	1001 000000000111	JUMPGT DONE
0101	1110 000000001001	OUT X
0110	1000 000000000000	JUMP LOOP
0111	1110 000000001010	OUT Y
1000	1111 000000000000	HALT
1001	0000 000000000000	CONST 0
1010	0000 000000000000	CONST 0

Linguaggio macchina

Linguaggio assembler

Esempio di file oggetto

Indirizzo	Opcode data	Significato
0000	1101 000000001001	IN X
0001	1101 000000001010	IN Y
0010	0000 000000001001	LOAD X
0011	0111 000000001010	COMPARE Y
0100	1001 000000000111	JUMPGT DONE
0101	1110 000000001001	OUT X
0110	1000 000000000000	JUMP LOOP
0111	1110 000000001010	OUT Y
1000	1111 000000000000	HALT
1001	0000 000000000000	CONST 0
1010	0000 000000000000	CONST 0

Il primo indirizzo di un file oggetto è sempre 0

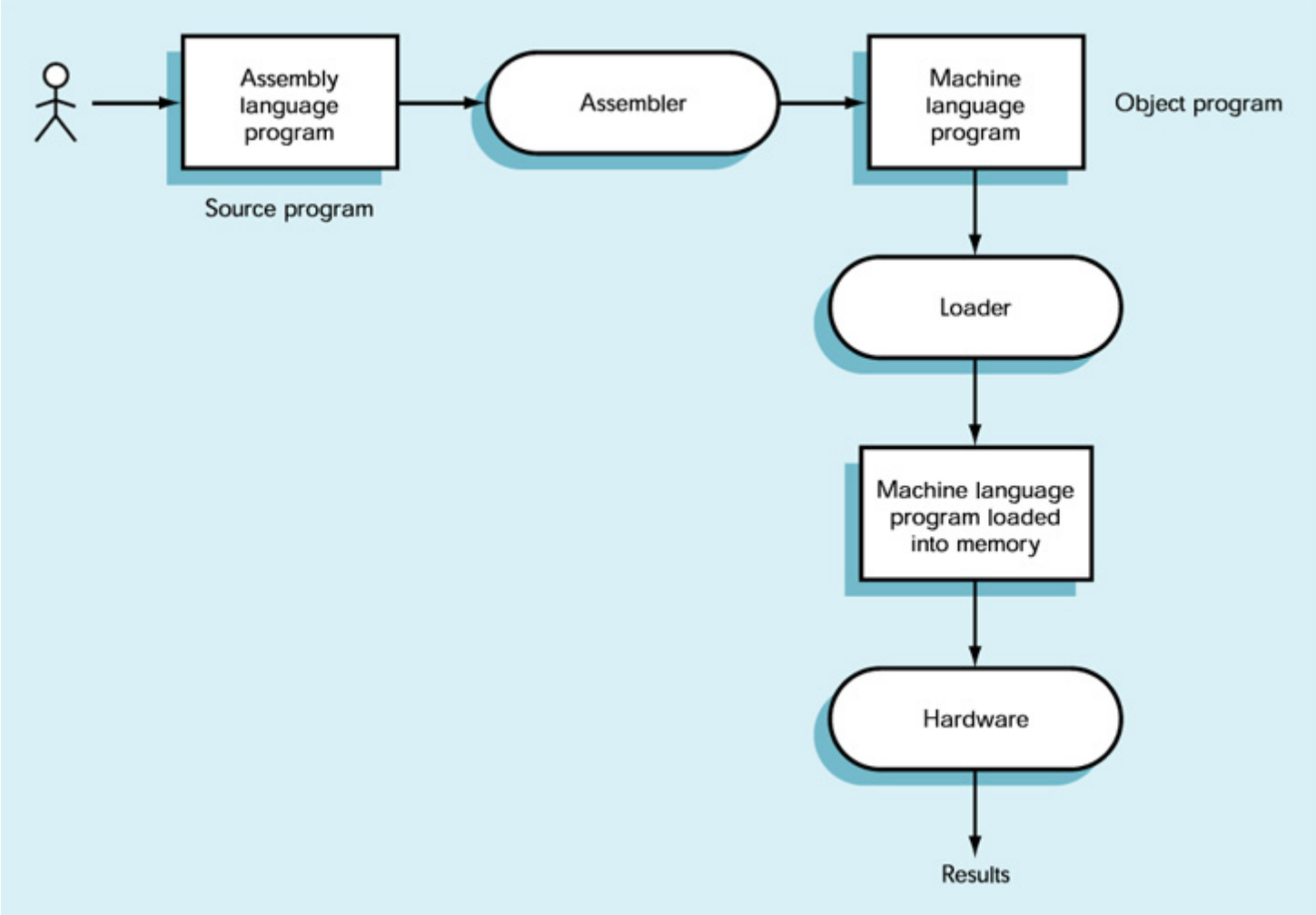
Linguaggio macchina

Linguaggio assembler

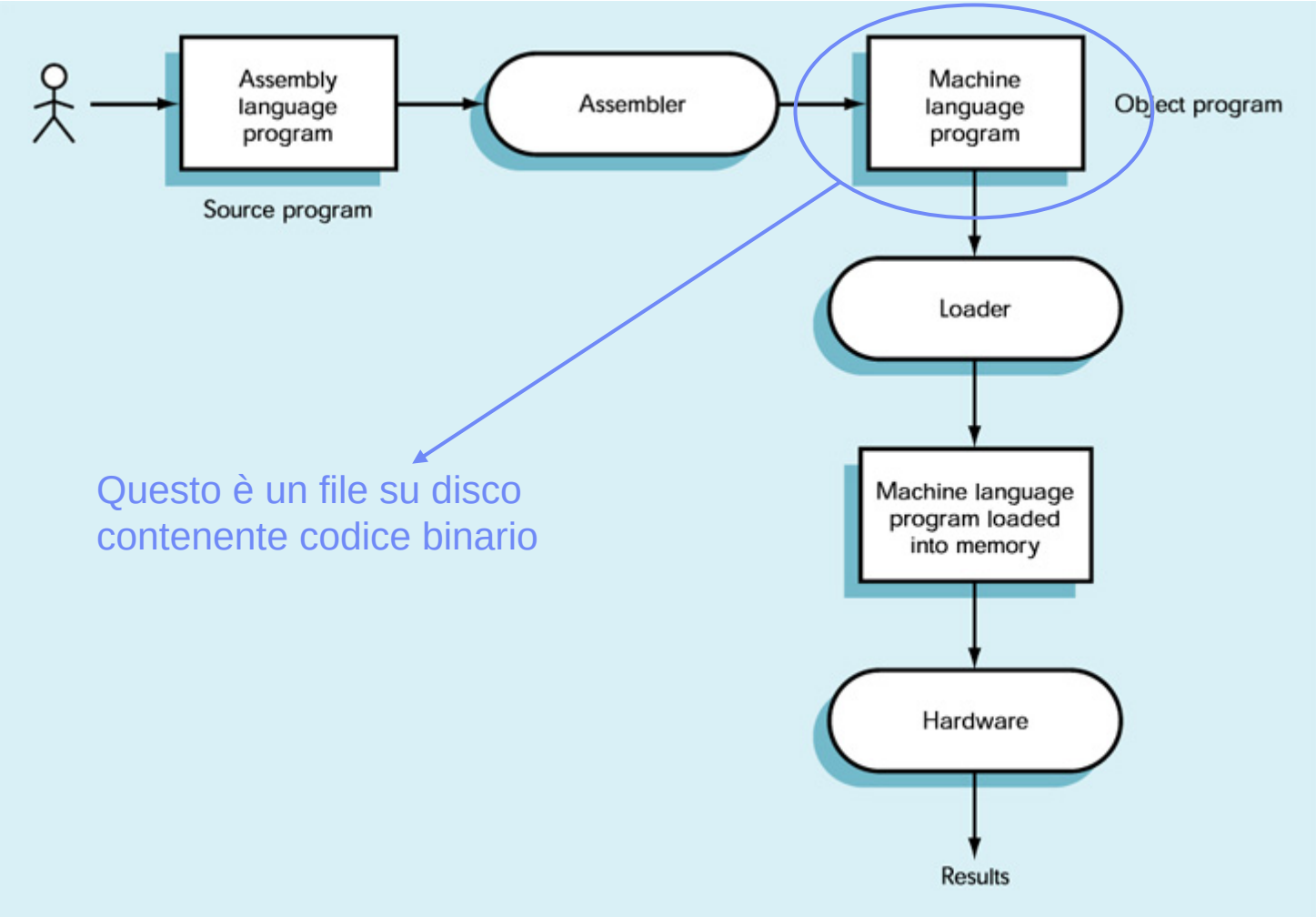
Applicazione programmi di sistema

- Scrivere un programma, eseguirlo e salvare i risultati
  1. Editor di testi
     Scrittura programma in linguaggio ad alto livello
  2. File system
     Memorizzare il programma come file di testo su disco fisso
  3. Traduttore
     Trasformare il programma dal linguaggio ad alto livello in linguaggio macchina
  4. Caricatore (loader)
     Riserva spazio in memoria per il programma, e caricare istruzioni per l'esecuzione
  5. Schedulatore
     Esegue il programma ogni qualvolta è il suo turno
  6. File system
     Memorizza i dati generati
  7. Debugger
     In caso di errori, esegue il programma passo-passo e tracciare l'errore

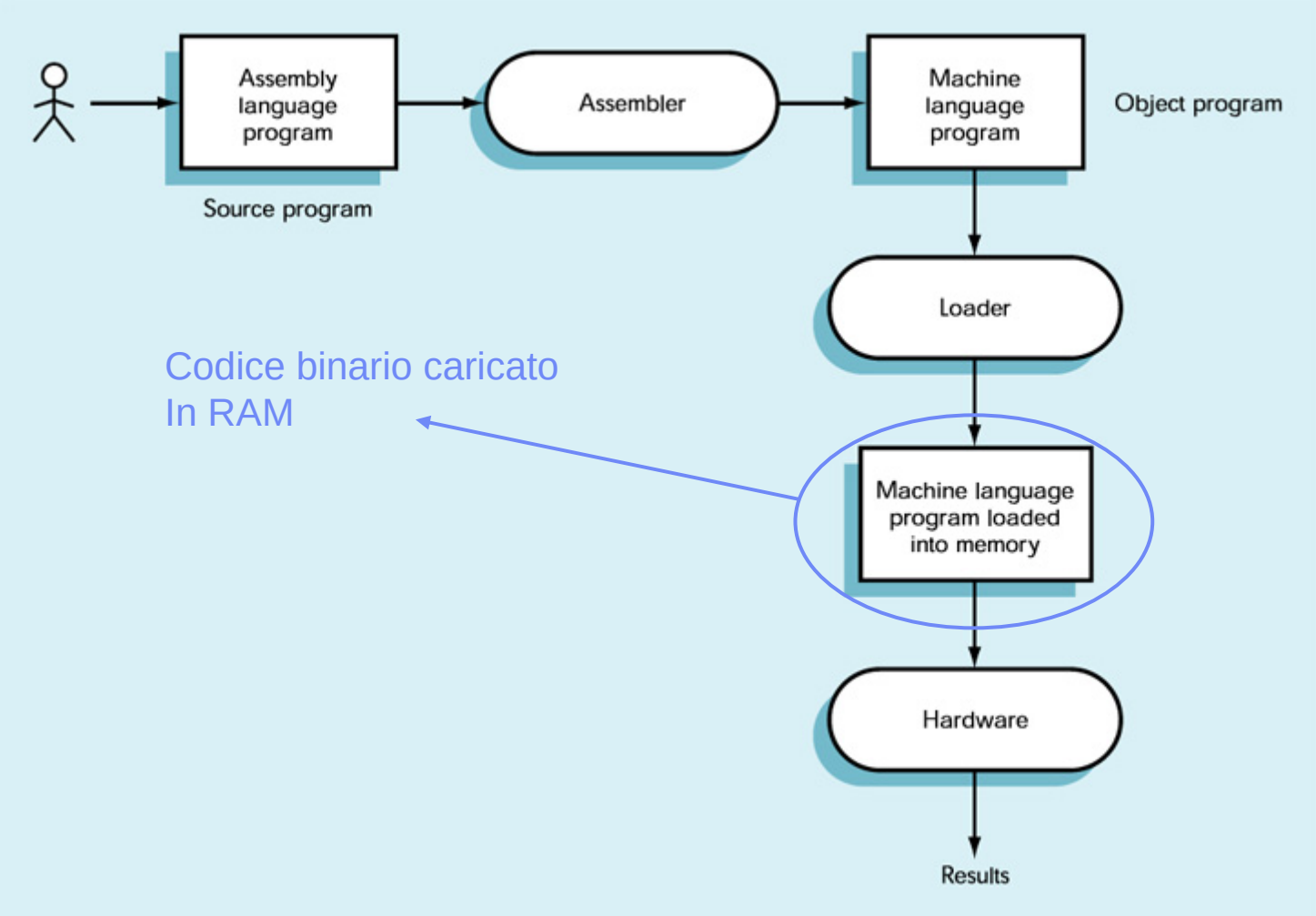
Traduzione / caricamento / esecuzione



Traduzione / caricamento / esecuzione



Traduzione / caricamento / esecuzione



Evoluzione dei linguaggi di programmazione

- Nascita dei linguaggi ad alto livello (BASIC, Pascal, C++, Java)
  - Orientati all'utente (più vicini al linguaggio naturale)
  - Indipendenti dalla particolare macchina
  - Rapporto 1:N con il linguaggio macchina: una istruzione ad alto livello richiede tipicamente N istruzioni in linguaggio macchina