

Informatica di Base¹ — Linea 1

Jianyi Lin

Dipp. di Matematica e Scienze dell'Informazione
Università degli Studi di Milano, Italia

jianyi.lin@unimi.it

a.a. 2011/12

¹© 2011 J.Lin, M. Monga. Creative Commons Attribuzione-Condividi allo stesso modo 2.5 Italia License.
<http://creativecommons.org/licenses/by-sa/2.5/it/>. Rielaborazione del materiale © 2009/10 © S. Mascetti.

Lezione I: Introduzione

Informazioni sul corso

- Lezione frontale: martedì 10:30–12:30 @ Aula 301
- Laboratorio: mercoledì pomeriggio @ Aula 307
 - Turno A1 (cognomi A–B): 13.00–14.30
 - Turno B1 (cognomi C–E): 14.30–16.00
- Esame:
 - Scritto con domande
 - Prova pratica per la parte di laboratorio (esercizi da risolvere al PC)
 - Eventuale prova orale per casi limite
- Libro: *Fluency. Conoscere e usare l'informatica*, di L. Snyder, Pearson Education.
- Per approfondimento e diletto: 1) *Informatica. Una panoramica generale*, di Brookshear, Pearson Education (9a ed.) 2) *Gödel, Escher, Bach. Un'eterna ghirlanda brillante. Una fuga metaforica su menti e macchine nello spirito di Lewis Carroll*, di Hofstadter, Adelphi
- Pagina del corso: <http://www.mat.unimi.it/users/lin>

Ricevimento

- Su appuntamento: jianyil.lin@unimi.it
- Ufficio 303, Via Comelico 39
(<http://maps.google.com/maps?q=via+comelico+39+milan&ie=UTF8&hq=&hnear=Via+Comelico,+39,+20135+Milano,+Lombardia,+Italia&ei=Zx7RTLf7F87s0Ye29MoM&ved=0CBUQ8gEwAA&z=16>)
- Scrivere da indirizzo universitario (mailstudenti.unimi.it)
- Suggerimenti per scrivere una buona email
(<http://vigna.dsi.unimi.it/comescrivere.php>)
- Corso a.a. precedente:
<https://mameli.docenti.dico.unimi.it/biotecno>
- Segreteria Didattica corsi Biotec.:
<http://users.unimi.it/biotecnologie>

Informatica

ACM — Association for Computing Machinery

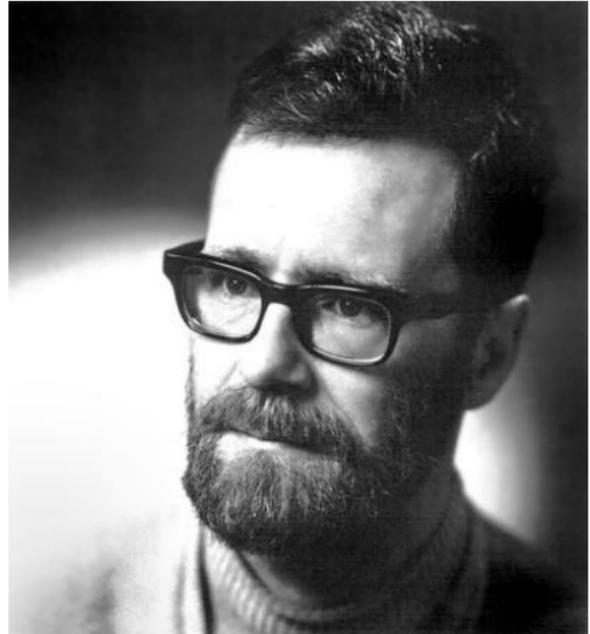
The discipline of computing is the systematic study of algorithmic processes that describe and transform information: their theory, analysis, design, efficiency, implementation, and application. The fundamental question underlying all computing is 'What can be (efficiently) automated?'

- **Non** è l'uso delle applicazioni (applitmatica?)

Scienza dei calcolatori?

Edsger Dijkstra

L'informatica non è la scienza dei calcolatori. Non più di quanto l'astronomia sia la scienza dei telescopi o la chirurgia la scienza dei bisturi.



Perché serve in un corso di biotecnologie

- capire cosa c'è dietro alle applicazioni informatiche per non farsene dominare
- comprendere l'importanza dell'informatica nel campo delle ricerche biotecnologiche
- utilizzare un approccio informatico nella risoluzione di problemi in tutti i campi
 - Es.: Chimica \rightsquigarrow hardware; DNA \rightsquigarrow software

Esempio di problem solving informatico

Il signor Sbevazzo ha a disposizione due recipienti vuoti (una damigiana e un bottiglione), in grado di contenere rispettivamente 9 litri e 4 litri. Vuole riempire la damigiana con esattamente 6 litri di acqua, ma le uniche operazioni che sa fare sono le seguenti:

RIEMPI D riempire la damigiana,

SVUOTA B svuotare il bottiglione,

VERSA versare il contenuto della damigiana nel bottiglione, fino a riempire il bottiglione o a svuotare la damigiana nel bottiglione.

Qual è la *piú breve* sequenza di queste operazioni che gli permette di raggiungere il suo obiettivo?

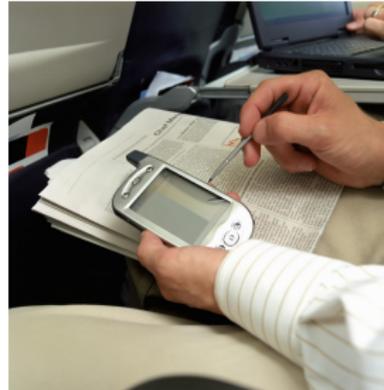
Computer



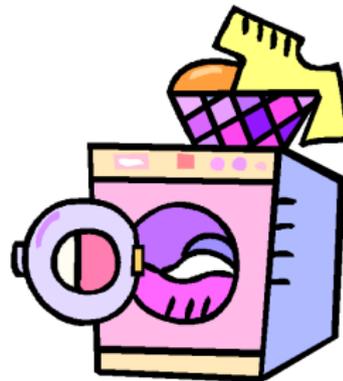
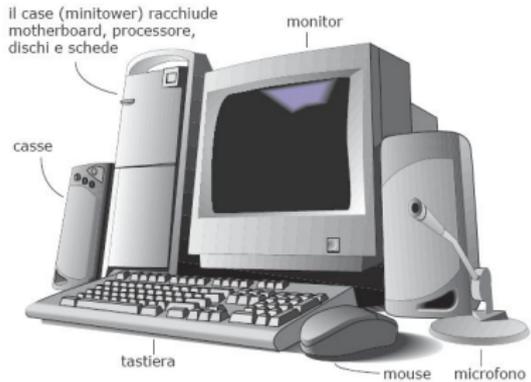
Computer



Computer

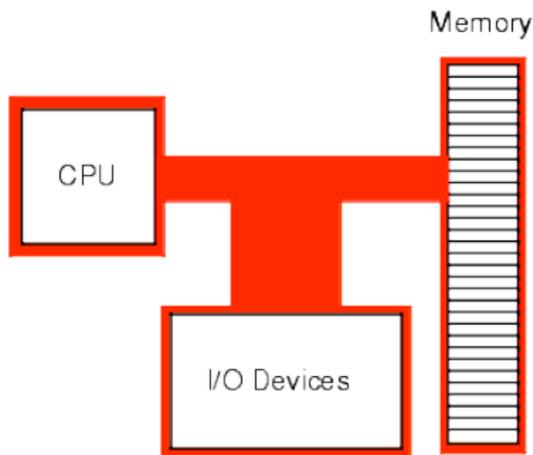


Computer



Un modello: la macchina di Von Neumann

Questi sistemi **molto** diversi fra loro possono essere *modellati* con uno schema molto semplice **la macchina di Von Neumann**

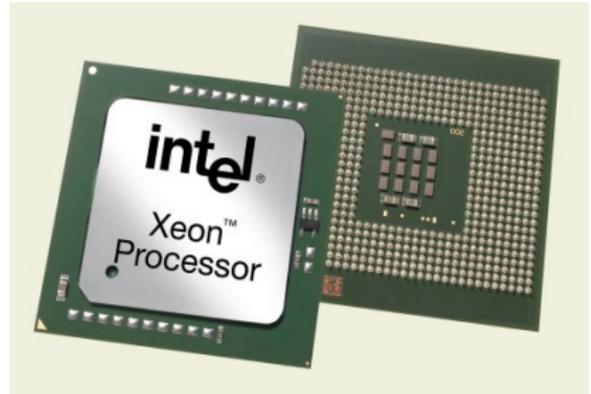


- un **processore** (CPU), in grado di **operare elaborazioni** di informazioni (segnali elettrici)
- una **memoria**, in grado di **conservare** informazioni
- **dispositivi periferici** per **trasdurre e attuare** segnali elettrici verso il mondo esterno
- collegamenti diretti fra CPU e memoria e fra CPU e dispositivi (e in alcuni casi anche fra memoria e dispositivi) (**bus**)

CPU

È in grado di eseguire istruzioni, in genere piuttosto semplici:

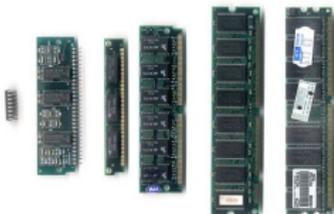
- p.es: attiva il segnale elettrico che simboleggia uno 0 in un certo componente elettronico
- piú comodo (e comprensibile) parlare metaforicamente: mette uno 0 in un *registro*
- somma il contenuto di due registri e mette il risultato nel terzo



Memoria

Memorizza segnali elettrici in componenti elettronici.

- È **direttamente** accessibile dal processore: ossia esistono istruzioni che possono riferirsi direttamente ai componenti della memoria (p.es. somma il contenuto di una *cella di memoria* con quello di un *registro*)
 - memoria centrale o primaria (RAM)
- Il segnale viene conservato solo fintanto che c'è corrente elettrica
 - memoria volatile (cioè non permanente)



Periferiche

Sono i dispositivi che permettono di interagire con il calcolatore

input forniscono dati da elaborare (trasdurre)

- tastiera
- mouse
- ...

output forniscono dati da interpretare nel mondo esterno (attuare)

- video
- speaker
- ...

input/output entrambe le cose

- touchscreen
- memorie di massa (permanenti!)
- schede di rete
-

Bus

Il bus sono collegamenti elettrici fra i componenti funzionali: sono ospitati sulla *scheda madre*

- Occorre stabilire le modalità con le quali si trasmette corrente elettrica sui *fili*
- p.es. PCI, SCSI, USB, Firewire, ...

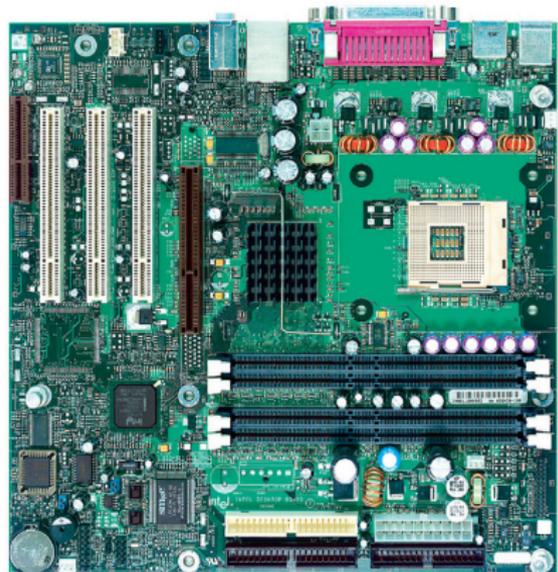


Figure 1.10. A computer motherboard.

I circuiti elettronici del processore

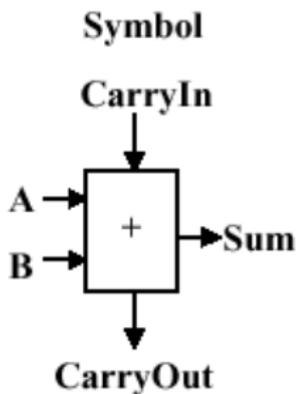
- Circuiti elettronici **bistabili**: due stati possibili, che convenzionalmente indichiamo con i simboli 0 e 1
- Non è una limitazione: qualsiasi informazione può essere rappresentata con una sequenza abbastanza lunga di simboli binari (**rappresentazione digitale binaria**)
- È molto facile definire le tabelle delle operazioni

	0	1
0	0	1
1	1	0*

I circuiti elettronici del processore

- Circuiti elettronici **bistabili**: due stati possibili, che convenzionalmente indichiamo con i simboli 0 e 1
- Non è una limitazione: qualsiasi informazione può essere rappresentata con una sequenza abbastanza lunga di simboli binari (**rappresentazione digitale binaria**)
- È molto facile definire le tabelle delle operazioni

	0	1
0	0	1
1	1	0*



Definition

A	B	CarryIn	CarryOut	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Ciclo di esecuzione

Le macchine del tipo Von Neumann funzionano semplicemente ripetendo ciclicamente queste tre operazioni

1. **Fetch** preleva una **parola** dalla memoria
2. **Decode** ogni sequenza di bit (parola) rappresenta un'operazione (e probabilmente i suoi operandi), quindi il processore decide quale istruzione deve essere attivata
3. **Execute** attiva l'istruzione, ottenendo i segnali elettrici risultanti

Queste operazioni vengono eseguite miliardi di volte al secondo (1GHz = 10^9 cicli al secondo).

L'insieme delle *parole di memoria* che vengono prelevate dal processore, costituiscono la parte variabile che determina il funzionamento di alto livello del sistema (**software**)