

# Introduzione

**Laboratorio di Calcolo**

Corso di Laurea in  
Fisica

Università degli Studi di Roma  
“La Sapienza”



# WARNING

- Questo canale è solo per studenti di Fisica il cui cognome inizia con le lettere **L-O**
- TUTTAVIA... prima di un'assegnazione definitiva dobbiamo sapere con esattezza il numero di studenti iscritti
- Per la frequenza del laboratorio (vedi dopo) **sarà data priorità agli iscritti del I anno**
- **COMPILATE** la form su **<https://goo.gl/YL8Cr4>**

# Introduzione al Corso (1)

---

- Scopo del Corso
    - programmazione e formalizzazione
    - strumenti: C, Linux
  - Sintassi del linguaggio C
- Problemi di matematica, logica, statistica

# Introduzione al Corso (2)

---

- Formato:
  - 3 Lezioni la settimana (~ 10 settimane)  
in due giornate, Martedì 11-12 e Mercoledì 11-13
  - 1 Laboratorio (3 ore) la settimana divisi in due sottogruppi  
**Venerdì e Lunedì 14-17 al Fermi, Il piano**
  - Ultimi 2 Laboratori valutati con voto
  - Gruppi di due studenti

# Introduzione al Corso (3)

---

- Pagina Web del canale
  - [www.roma1.infn.it/people/barone/labcalc](http://www.roma1.infn.it/people/barone/labcalc)
- Logistica del Laboratorio
  - aula Informatica (Via Tiburtina 205, aula 17)
  - aula Informatica N.E. Fisica II piano
  - PC con Linux
  - l'aula di Fisica e' disponibile per laboratorio libero in alcune ore da definirsi successivamente

# Introduzione al Corso (4)

---

- Testo:

L.M.Barone, E.Marinari, G.Organtini, F.Ricci-Tersenghi

Programmazione Scientifica, ed. Pearson  
35 €, copre corsi dei tre anni

- Materiali utili sul sito

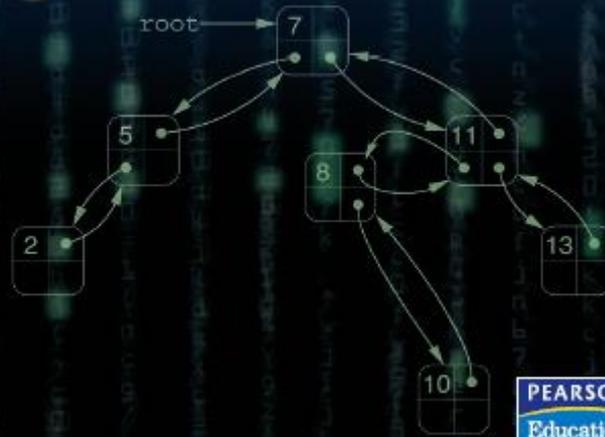
<http://www.programmazione scientifica.org/>

- un buon manuale del linguaggio C è  
Kelley & Pohl: C-Didattica e Programmazione  
ed.Addison & Wesley (~ 30 Euro)

Luciano M. Barone • Enzo Marinari  
Giovanni Organtini • Federico Ricci-Tersenghi

# PROGRAMMAZIONE SCIENTIFICA

Linguaggio C, algoritmi e modelli nella scienza



PEARSON  
Education

# Introduzione al Corso (5)

---

- Il laboratorio NON basta

Lavoro a casa:

- Su Windows macchina virtuale (VM) scaricabile via rete; vedere istruzioni sulla pagina web del corso
- Su Mac basta aprire una Terminal Window

# Introduzione al Corso (6)

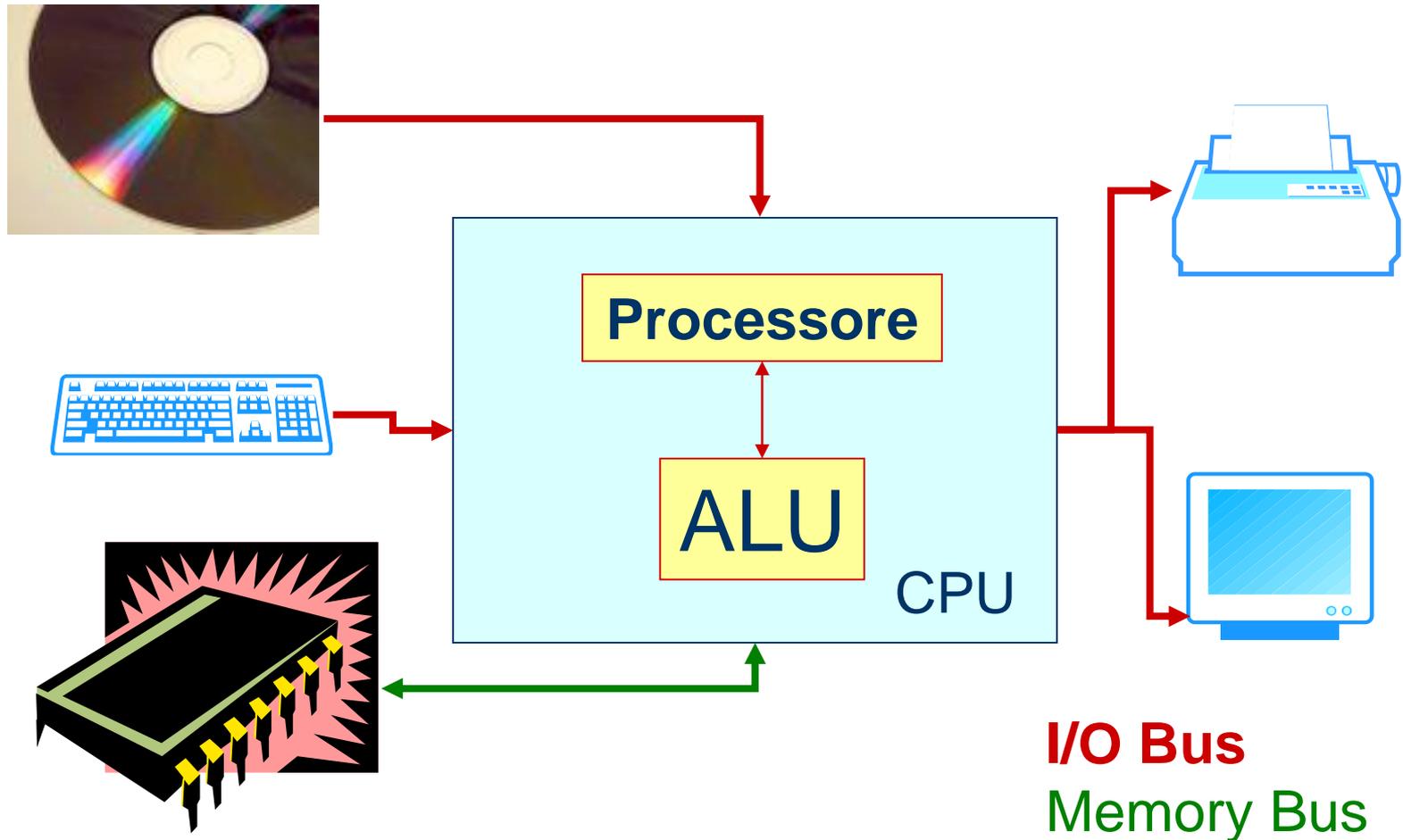
---

- Mio indirizzo e-mail:
  - `luciano.barone@roma1.infn.it`
- Ricevimento il Mercoledì o previa mail
- Esami: prova pratica (+ bonus da es.valutate se l'esame è sostenuto alla fine del semestre)
- Date esami: fine Gennaio e fine Febbraio (vedere pagina web)

# Calcolatori

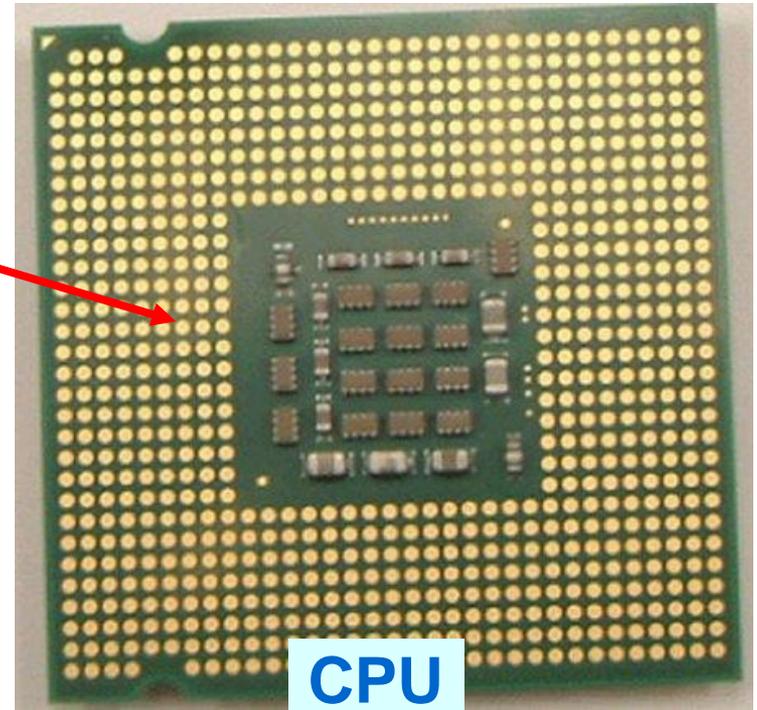
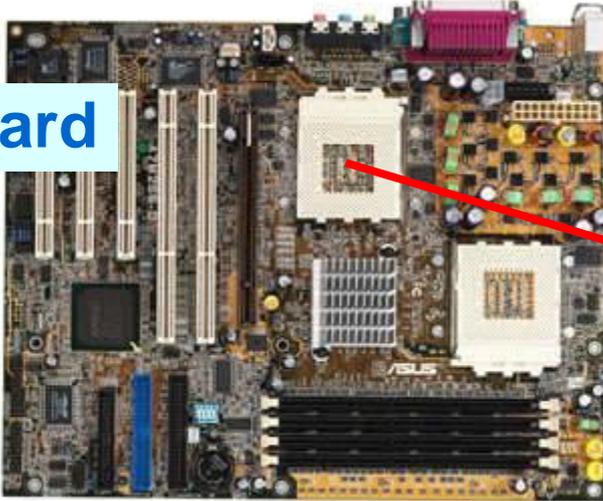
- Calcolatori Digitali
  - Informazione codificata utilizzando numeri (digits)
  - 2 numeri  $\leftrightarrow$  sistema binario (bit = **b**inary **digi**t)
    - 1/0
    - Interruttore acceso/spento
    - Tensione  $V=0/V=-0.8$  (NIM)
    - Condensatore Carico/Scarico
  - byte: sequenza di 8 bit
- Compiti
  - Eseguire operazioni logiche e/o numeriche
  - Operazioni complesse “spezzate” in operazioni elementari

# Architettura HW di base



# Architettura HW di base

Motherboard



CPU



Hard Disk



# Software

- Insieme di istruzioni e dati trattati dal computer
- Le istruzioni possono essere a diversi livelli
  - Istruzioni macchina: insieme limitato di funzioni specifiche della CPU
    - Leggi un byte da una locazione di memoria
    - Somma tra loro il contenuto di due registri
    - Esegui le istruzioni a partire dalla locazione m
  - Istruzioni di alto livello: funzioni che vengono tradotte in sequenze di istruzioni in linguaggio macchina prima di essere eseguite

# Il sistema operativo

---

- Il sistema operativo è l'insieme di programmi che gestisce il computer:
  - permette l'interazione con il mondo esterno (Input/Output)
  - permette di eseguire programmi dell'utente
  - bilancia le risorse del sistema
- Sullo stesso PC possono convivere o alternarsi diversi sistemi (es. Linux e Windows o MacOS)

# Linguaggi di alto livello

---

- Linguaggi sintatticamente complessi **facili** da usare dagli umani: la traduzione in linguaggio macchina viene fatta “dietro le scene”
- Si possono classificare in procedurali (C, FORTRAN, PASCAL) e Object Oriented (C++,JAVA)

# Compilatori

- Traduttore da linguaggio di alto livello a linguaggio macchina
- Linguaggi evoluti
  - Versatilità, Semplicità, Flessibilità
  - **Indipendenti** dalla macchina



# Linker

---

- Uno o più files oggetto devono essere *collegati* alle *librerie*
- Librerie: insieme di files oggetto preparati dal produttore del compilatore con le istruzioni per l'esecuzione di compiti comuni
  - Librerie matematiche
  - Librerie grafiche
  - Servizi di I/O
- L'operazione di *link* può essere esplicita o implicita

# Linguaggio Macchina

---

- Ad ogni istruzione è associato un codice di N bit
- La CPU inizia caricando il byte presente nella locazione 0 e lo interpreta come istruzione
- A seconda dell'istruzione si interpretano i bytes successivi

# Esempio

- Supponiamo che i seguenti codici binari corrispondano alle seguenti istruzioni

001	ldA, X	Carica il contenuto della memoria X nel registro A
010	ldB, X	Carica il contenuto della memoria X nel registro B
011	sumAB	Somma i contenuti dei registri A e B e metti il risultato in A
100	setA, X	Scrivi nella memoria X il contenuto del registro A

# Esempio

- Ed ora supponiamo che la memoria del nostro computer sia così riempita:

Address	Content
001	001
010	101
011	010
100	011
101	011
110	100
111	000

# Esempio

- Ed ora supponiamo che la memoria del nostro computer sia così riempita:

Address	Content
001	001
010	101
011	010
100	011
101	011
110	100
111	000

001 viene interpretato come una istruzione e quindi come `ldA,X`.

Di conseguenza il byte successivo contiene il valore di X.

# Esempio

- Ed ora supponiamo che la memoria del nostro computer sia così riempita:

Address	Content
001	001
010	101
011	010
100	011
101	011
110	100
111	000

101 (5) viene dunque caricato nel registro A – è un DATO  
Il registro A non è visibile

# Esempio

- Ed ora supponiamo che la memoria del nostro computer sia così riempita:

Address	Content
001	001
010	101
011	010
100	011
101	011
110	100
111	000

010 viene nuovamente interpretato come una istruzione e dunque come `ldB, X`. Nel registro B viene caricato il numero 011 (3)

# Esempio

- Ed ora supponiamo che la memoria del nostro computer sia così riempita:

Address	Content
001	001
010	101
011	010
100	011
101	011
110	100
111	000

011 rappresenta l'istruzione di somma del contenuto dei registri A e B. In A finisce il valore 8 (5+3)

# Esempio

- Ed ora supponiamo che la memoria del nostro computer sia così riempita:

Address	Content
001	001
010	101
011	010
100	011
101	011
110	100
111	000

l'istruzione di somma non ha parametri perciò il byte successivo è ancora un'istruzione: setA, X dove X=0