

Laboratorio di Calcolo

Corso di Laurea in Fisica — Università di Trieste

Lezione 1 — Introduzione al corso

A.A. 2025/2026

Chi sono



Prof. Alex Saro

Professore Associato

Dipartimento di Fisica, Università di Trieste

asaro@units.it



La mia ricerca

Astrofisica e Cosmologia osservativa

Ammassi di galassie, simulazioni cosmologiche,
missione ESA Euclid, South Pole Telescope

Affiliazioni

INAF — Oss. Astronomico di Trieste

IFPU — Inst. for Fundamental Physics

Percorso

Laurea e Dottorato — UniTS

PostDoc e Asst. Prof. — LMU München

Dal 2019: UniTS

Collaboratori del corso

Prof.ssa Milena Valentini

Dott. Fontana, Dott. Salvalaggio, Dott. Lacopo

Dott.ssa Damiano, Dott.ssa Esposito

Partecipa a questo evento Wooclap



1

Vai a wooclap.com

2

Immettere il codice dell'evento nel
banner superiore

Codice evento

CYREQY

Obiettivi del corso

1

Sviluppare la capacità di scrivere con metodo programmi funzionanti che implementino semplici algoritmi numerici

2

Introdurre semplici algoritmi numerici, la loro derivazione e il controllo della precisione

3

Mettere in grado di valutare l'affidabilità dei risultati numerici

4

Acquisire strumenti di lavoro (Python, Bash, LaTeX) che vi accompagneranno per tutta la carriera

Strumenti e Laboratori

STRUMENTI DEL CORSO

Python

Linguaggio di programmazione principale.
Versatile, usato in fisica, data science, AI.

Bash

Interfaccia a riga di comando.
Automazione e gestione del sistema.

LaTeX

Sistema di scrittura scientifica. Equazioni,
report, tesi.

COME FUNZIONANO I LABORATORI

G

Gruppi assegnati

Postazioni omogenee in aula, non portate il PC. Istruzioni online per replicare a casa.

T

2.5 ore flessibili

Se finite prima potete andare. Team: Prof.ssa Valentini, Dott. Fontana, Salvalaggio, Lacopo, Dott.sse Damiano e Esposito.

C

Consegna

Ogni lab produce una consegna. Parte integrante del percorso.

R

Recupero assenze

Nella settimana successiva con un altro turno. Mandate una mail prima.

Novità: il corso passa da Fortran a Python.

Come funziona il laboratorio



Lavoro in laboratorio

Lavoro individuale

al proprio computer

Gruppi di 3 studenti

per confronto e discussione

Docenti di didattica integrativa

presenti in aula per supporto

Quando hai un problema:

1

Prova a risolverlo da solo/a

2

Confrontati con il tuo gruppo

3

Chiedi ai docenti in aula

AI e LLM nel corso: buone norme

Usi consentiti



Chiedere spiegazioni su concetti
("Cosa fa questa riga di codice?")

Capire messaggi di errore
("Cosa significa questo traceback?")

Ripassare o approfondire dopo il lab

Esplorare alternative e documentazione

Usi da evitare



Copiare codice senza capirlo
("Scrivi un programma che...")

Far scrivere la soluzione all'AI
al posto vostro

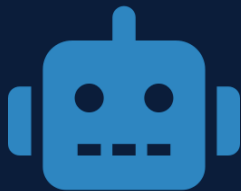
Usarla durante le verifiche/esami

Fidarsi ciecamente dell'output

L'obiettivo è imparare a programmare, non a delegare. L'AI è un utensile, non un sostituto.



Il principio guida



*Se non sai spiegare cosa fa il tuo codice
riga per riga, non l'hai scritto tu.*

L'AI può accelerare l'apprendimento se usata con consapevolezza.
Ma il vostro obiettivo è costruire competenze, non consegnare risultati.

Programma di oggi



Organizzazione pratica e struttura del corso

Struttura, laboratori, logistica, esame



Perché questo corso

Il computer come strumento del fisico moderno



Perché Python?

Differenza rispetto agli anni scorsi



Una breve storia della fisica computazionale

Dagli abachi alla fisica computazionale

L'esame

Formato

- Esame al computer (TBC)
- Sia parte concettuale sia numerica
- Problema da risolvere con uno script o notebook
- I temi ricalcano i lab svolti durante il corso

Criteri di valutazione

1

Correttezza

del risultato

2

Precisione

controllo e discussione dell'errore

3

Qualità codice

commenti, funzioni, nomi chiari

4

Output e leggibilità

grafici, tabelle, messaggi a schermo

Un risultato fisicamente sbagliato con codice pulito non basta — e viceversa.

Perché Python? (e non per esempio Fortran)

NOVITÀ RISPETTO ALL'ANNO SCORSO

Fortran (ma anche C, C++,..)

✓ Punti di forza

- Ancora usato per calcolo HPC intensivo
- Ottimizzazione eccellente
- Plasma, meteorologia, fluidodinamica

✗ Svantaggi per iniziare

- Molte righe per cose semplici
- Dichiarazioni di tipo obbligatorie
- Ecosistema dati/ML limitato

Python

✓ Perché lo scegliamo

- Leggibile e conciso
- NumPy, SciPy, matplotlib, pandas
- Standard in fisica, AI, data science
- Compila C sotto: veloce dove conta

```
# Python: tracciare  $\sin(x)/x$  in <10 righe
```

```
x = np.linspace(0.01, 4*np.pi, 1000); y = np.sin(x)/x; plt.plot(x, y)
```

In Fortran: ~80 righe + compilazione + gestione array manuale

Il principio ricorrente del corso

PRIMA L'ALGORITMO A MANO — POI LA LIBRERIA



Esempi: media → `numpy.mean` · trapezi → `scipy.integrate.quad` · bisezione → `scipy.optimize` · LCG → `numpy.random`

Perché un laboratorio di calcolo?

IL COMPUTER PER UN FISICO



Simulare

Sistemi fisici non risolvibili analiticamente: pendolo con attrito, tre corpi, gas reali



Analizzare

Grandi quantità di dati sperimentali: fit, istogrammi, propagazione errori



Automatizzare

Calcoli ripetitivi che farebbero impazzire con carta e penna



Comunicare

Grafici e relazioni in LaTeX: il linguaggio della scienza moderna

Il computer è il terzo pilastro della fisica moderna, insieme a teoria e sperimentazione.

Mini-storia degli strumenti di calcolo

...e la nascita della fisica computazionale

Abaco
~3000

Pascaline
1642
Jacquard
1801
Babbage
1837
Enigma
1918
Colossus
1918
Z3
1941
Enigma
1918
Colossus
1918
Z3
1941

Gli albori: calcolo manuale



Abaco romano

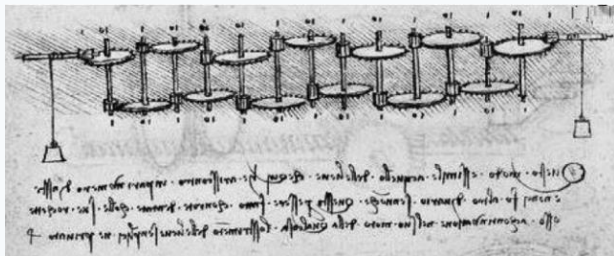
~3000 a.C. — Abaco

Il primo strumento di calcolo. Usato da Sumeri, Romani, Cinesi. In alcune forme ancora oggi.



Abaco moderno (video)

Gli albori: calcolo manuale



~1500 — Leonardo da Vinci

Progetta una calcolatrice meccanica (Codice Atlantico). Non fu mai costruita in vita.



1642 — Blaise Pascal

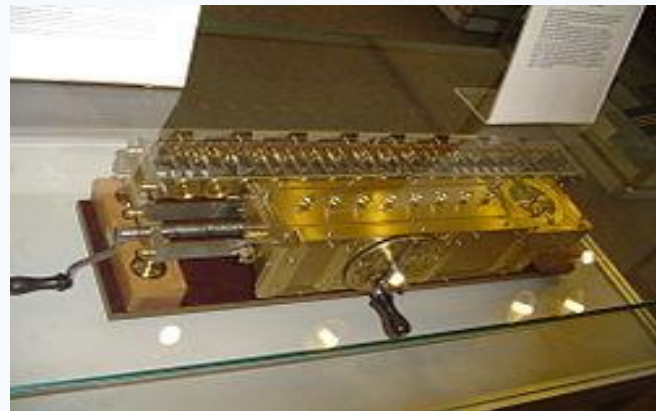
La Pascalina: prima calcolatrice meccanica funzionante. Addizioni e sottrazioni con ruote dentate.

Gli albori: calcolo manuale

Leibnitz

1646–1716

Calcolatrice binaria. Intuisce che il sistema binario è la base naturale per il calcolo meccanico.



1632–1970 — Regolo calcolatore

Basato sui logaritmi. Strumento standard di ingegneri e scienziati per secoli — fino agli anni '70!

Verso le macchine programmabili

Jacquard

1752–1834

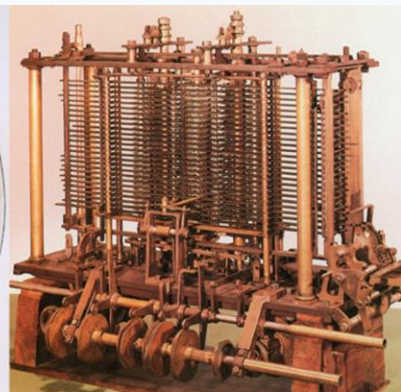
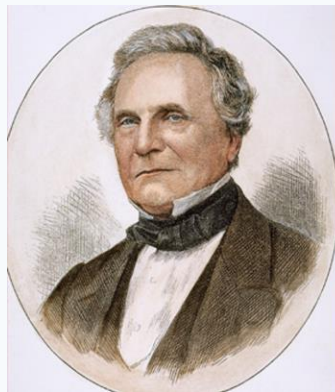
Telaio programmabile con schede perforate. L'idea rivoluzionaria: "istruzioni codificate" che guidano una macchina.



Babbage

1791–1871

Macchina alle differenze e macchina analitica: il primo progetto di un computer general-purpose. Ada Lovelace scrive il primo "programma".



L'idea chiave di Jacquard e Babbage: separare le istruzioni (il programma) dai dati. Questo principio è alla base di ogni computer moderno.

Verso le macchine programmabili

Hollerith

1860–1929

Sistemi di analisi su scheda perforata. Usati per il censimento USA del 1890. Fonda l'azienda che diventerà IBM.



Schema di una scheda perforata

I primi computer

IBM Mark I — 1943

Howard Hathaway Aiken
Elettromeccanico

3 addizioni/sottrazioni al secondo

1 moltiplicazione in 6 secondi

1 divisione in 15.3 secondi

1 logaritmo in più di un minuto

Lungo 15 m, peso 5 t, 750.000 componenti

Confronto con oggi:

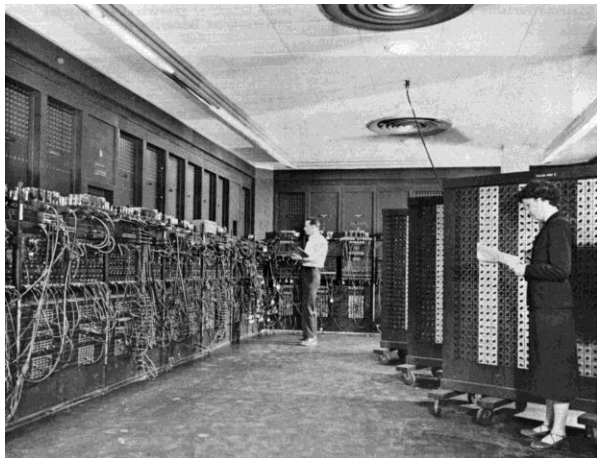
Un moderno smartphone esegue miliardi di operazioni al secondo contro le 3 operazioni/s del Mark I.

10⁹× più veloce!



I primi computer

ENIAC — 1946



Eckert e Mauchly

Primo computer elettronico programmabile

Somma 10 cifre in $1/5000$ s
357 moltiplicazioni/s, 35 divisioni/s
17.468 valvole, consumo: 150 kW

CDC 7700 — 1973



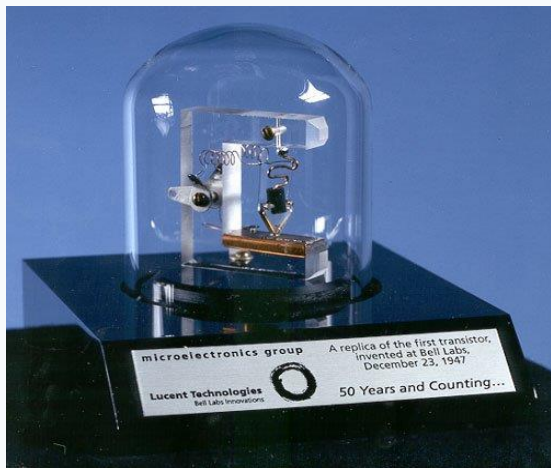
Supercomputer CDC 7700 — Control Data Corporation

La potenza di calcolo cresce...

1947

Transistor

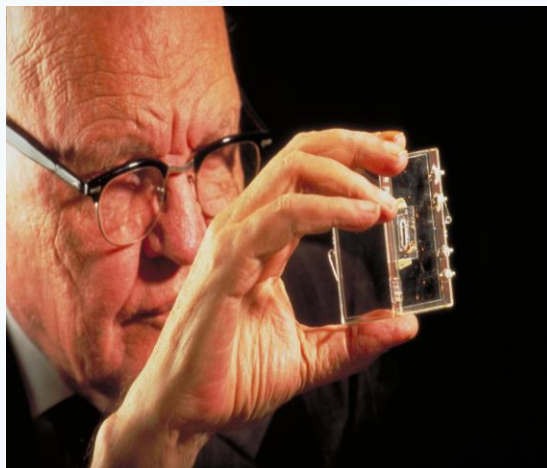
Bardeen, Brattain, Shockley
Sostituisce le valvole:
più piccolo, veloce, affidabile



1959

Circuito integrato

Jack Kilby (Nobel Fisica 2000)
Milioni di transistor su $\sim 1 \text{ cm}^2$
Nasce la microelettronica



1971

Microprocessore

Federico Faggin (fisico)
Intel 4004: primo μ processore
Il computer diventa personale



La potenza di calcolo cresce...

Legge di Moore (1965)

Il numero di transistor per chip raddoppia ogni ~2 anni.

Da più di 60 anni!

C'è un limite?

Scala atomica? Computer quantistici? Chip neuromorfici?

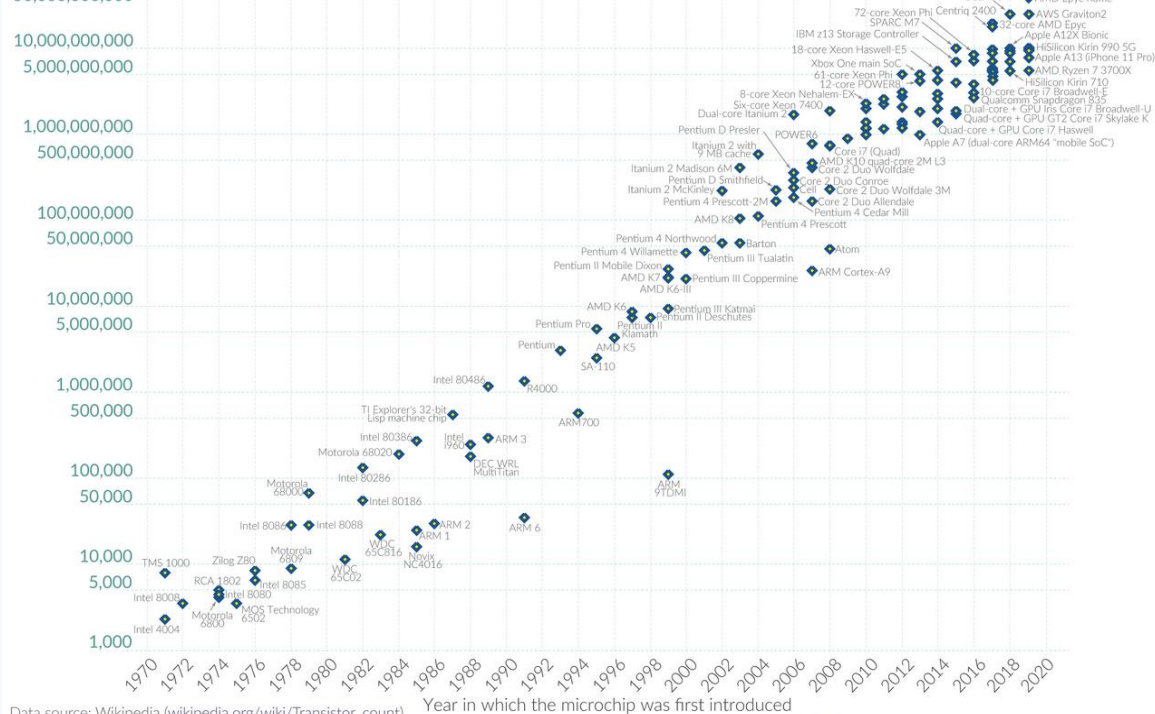
Moore's Law: The number of transistors on microchips doubles every two years

Our World
in Data

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.

Transistor count

50.000.000.000



Data source: Wikipedia (wikipedia.org/wiki/Transistor_count)

OurWorldinData.org – Research and data to make progress against the world's largest problems.

Licensed under CC-BY by the authors Hannah Ritchie and Max Roser.

CINECA Leonardo & il TOP500

Leonardo — Pre-Exascale

Dove: CINECA, Tecnopolo di Bologna, Italia

Inaugurato: Novembre 2022

Prestazioni: ~241 PetaFLOPS (HPL)

GPU: 13.824 NVIDIA A100 (3.456 nodi)

Rete: NVIDIA InfiniBand HDR 200 Gbit/s

Architettura: Atos BullSequana XH2000

Modulo Data Centric: 1.536 nodi x 2 processori Intel Xeon Platinum 8480 x 56 core ciascuno

Finanziatore: EuroHPC JU + MUR (Ministero Università)



Il TOP500

Cos'è il TOP500?

Classifica semestrale dei 500 supercomputer più potenti al mondo, misurati con il benchmark LINPACK (HPL). Pubblicata da Jun. e Nov.

Leonardo in classifica:

2022 (Nov): **#4 mondiale**

2023 (Giu): **#4 mondiale**

2023 (Nov): **#6 mondiale**

2024 (Nov): **#9 mondiale**

241

PetaFLOPS
(HPL)

13.824

GPU A100
NVIDIA

#9

Posizione TOP500
al mondo (2024)

#2

In Europa
2° in Europa

BORA — Cluster HPC del Dipartimento di Fisica, UniTS

Cos'è BORA?

Cluster HPC ibrido ed estensibile del Dip. Fisica UniTS. Il nome richiama il vento locale — simbolo di potenza e velocità.

Finanziamento: Dipartimento di Eccellenza 2023–2027

Scheduler: SLURM

Accesso: Ricercatori e studenti del Dip. Fisica

Software: Python, MPI, CUDA, GCC, BLAS/LAPACK, moduli ambiente



<https://sites.units.it/bora/index.html>

Hardware

Utilizzi: Simulazioni, ML, calcolo parallelo MPI/GPU, tesi

Computing nodes

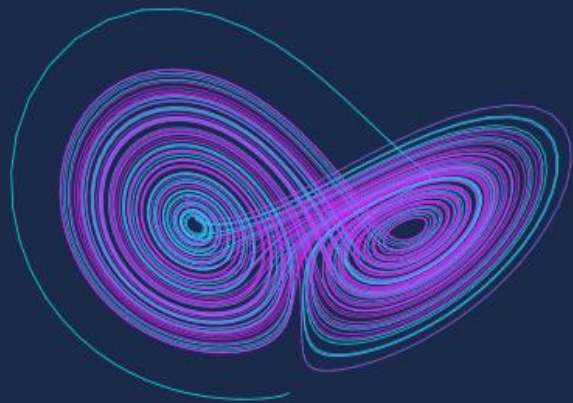
Here are the specs of the computing nodes:

Node type	Description	Nodes count	CPU	Cores	Frequency	RAM	GPU
Login	Entry point to the cluster	1	Xeon Silver 4410Y	24	2.0 GHz	128 Gb	—
CPU	General-purpose CPU nodes	4	Xeon Gold 6354	36	3.0 GHz	256 Gb	—
GPU	Nodes equipped with NVIDIA graphic cards	1	Xeon Gold 6354	36	3.0 GHz	256 Gb	2 x Nvidia A30 24 Gb
FAT	Nodes for memory-intensive jobs	1	Xeon Gold 6354	36	3.0 GHz	1024 Gb	—

...e nasce la fisica computazionale

A un certo punto il computer non è più solo
uno strumento di supporto.
Diventa uno strumento di scoperta.

*Risultati che non vengono fuori da un conto a mano.
Vengono fuori dal computer.*



Fermi, Pasta, Ulam (1955)

La domanda

Come viene raggiunto l'equilibrio termico in un sistema di oscillatori accoppiati con interazione non lineare? Una debole anarmonicità nelle interazioni tra atomi di un cristallo è sufficiente a far redistribuire l'energia tra tutti i modi vibrazionali fino a raggiungere l'equilibrio termico?

266.

STUDIES OF NON LINEAR PROBLEMS

E. FERMI, J. PASTA, and S. ULAM
Document LA-1940 (May 1955).

La sorpresa!

L'energia, invece di distribuirsi tra i modi (equipartizione), ritorna quasi completamente al modo iniziale!

Non tutti i sistemi interagenti riescono ad equilibrare. Occorre superare una soglia di interazione.

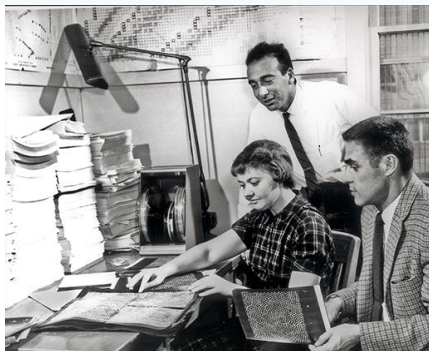
Questo risultato inaspettato aprì la strada allo studio dei solitoni e alla teoria del caos.

Alder, Wainwright e Mansigh (1957)

La domanda

Che succede a un gas di palle da biliardo (sfere rigide) quando la densità cresce?

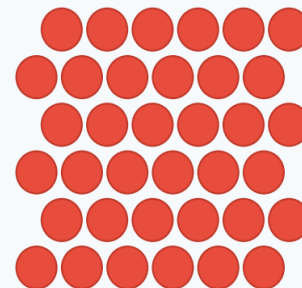
Simulazione di dinamica molecolare al Lawrence Livermore National Laboratory: si seguono le traiettorie di N particelle classiche.



Fluido (bassa densità)



Solido (alta densità)



La sorpresa!

Dalle traiettorie generate sul computer appare una transizione di fase fluido-solido!

Sfere dure, senza alcuna attrazione, si ordinano spontaneamente ad alta densità. Un risultato che nessun conto analitico aveva previsto.

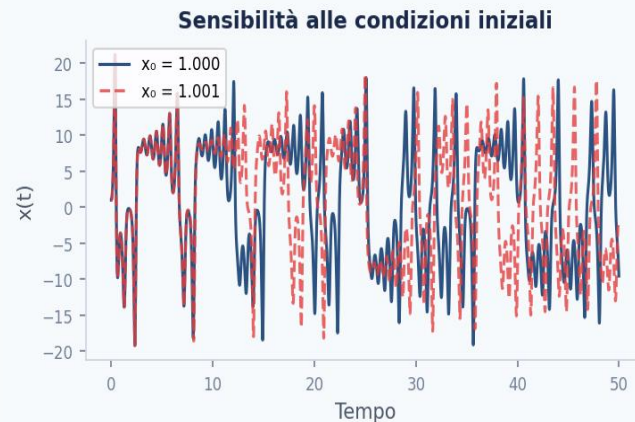
Nasce la dinamica molecolare: oggi una delle tecniche più usate in fisica, chimica, biologia e scienza dei materiali.

Lorenz e la nascita della teoria del caos (1963)

Edward Lorenz — meteorologo al MIT

Simula un modello semplificato dell'atmosfera. Per risparmiare tempo, riparte da metà simulazione arrotondando i dati da 6 a 3 cifre decimali.

La traiettoria diverge completamente! Una differenza minuscola (0.001) nelle condizioni iniziali porta a risultati totalmente diversi.



L'effetto farfalla

"Il battito d'ali di una farfalla in Brasile può provocare un tornado in Texas?"

Anche sistemi deterministici possono essere imprevedibili: sensibilità alle condizioni iniziali.

La lezione per noi

La precisione numerica conta!
Gli errori di arrotondamento non sono solo un dettaglio tecnico: possono cambiare completamente il risultato.

Questo è uno dei temi centrali del corso.

The Nobel Prize in Physics 2024

THE NOBEL PRIZE
IN PHYSICS 2024

Illustrations: Niklas Elmehed



John J. Hopfield

Geoffrey E. Hinton

"for foundational discoveries and inventions
that enable machine learning
with artificial neural networks"

THE ROYAL SWEDISH ACADEMY OF SCIENCES

Geoffrey Hinton & John Hopfield

Premio Nobel per la Fisica assegnato per le scoperte fondamentali sulle reti neurali artificiali, che hanno gettato le basi dell'intelligenza artificiale moderna.

Il riconoscimento sancisce il ruolo centrale del machine learning nella scienza contemporanea.

Oggi: il computer è ovunque nella fisica

Onde gravitazionali

LIGO/Virgo: i segnali delle fusioni di buchi neri sono estratti dal rumore grazie ad algoritmi e template numerici

Bosone di Higgs

LHC al CERN: miliardi di collisioni analizzate computazionalmente per trovare il segnale tra il fondo

Simulazioni cosmologiche

Universi interi simulati al computer per capire la formazione di galassie e la struttura su grande scala

Clima e meteo

Previsioni meteorologiche e modelli climatici: milioni di equazioni risolte su supercomputer

Protein folding

AlphaFold: l'IA ha rivoluzionato la predizione della struttura delle proteine (Nobel Chimica 2024)

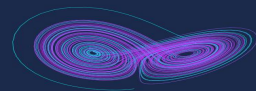
Fusione nucleare

Simulazioni MHD del plasma nei tokamak: fondamentali per ITER e il futuro dell'energia

Il messaggio

**Il computer non è solo una calcolatrice veloce.
È uno strumento di scoperta scientifica.
Ma i suoi risultati non sono automaticamente "giusti".**

Capire come funziona, sapere cosa può andare storto,
e sviluppare senso critico sui risultati numerici:
questo è quello che faremo in questo corso.



Fisica con il computer

Ingredienti di questo Corso (Trailer)

01

Architettura del computer

Poche idee corrette sul funzionamento di massima del computer (CPU, memoria, I/O)

02

Algoritmi numerici

Algoritmi per trasformare modelli fisici in procedure computazionali

03

Aritmetica floating point

Differenze tra aritmetica usuale e la sua rappresentazione su computer (errori, precisione)

04

Linguaggio di programmazione

Trasformare algoritmi in applicazioni funzionanti per risolvere problemi fisici