

# Corso di Laurea Magistrale in Fisica

## Piano di Studi Fisica Teorica Generale

### Corso di Laboratorio di Fisica Computazionale (CFU 4+2)

**Dott. Gianluca Lattanzi**  
**e-mail: [gianluca.lattanzi@ba.infn.it](mailto:gianluca.lattanzi@ba.infn.it)**

**Descrizione del corso.** Il corso si pone l'obiettivo di fornire agli studenti un'introduzione alla soluzione dei problemi della fisica mediante il calcolatore elettronico. Si assume che lo studente non abbia alcuna esperienza di programmazione per costruire, fin dalle prime lezioni, un quadro di riferimento nell'ambito del linguaggio di programmazione python, liberamente disponibile su ogni piattaforma (windows, linux, MacOSX) e facilmente adattabile alle esigenze del fisico teorico, incluse le istruzioni per la visualizzazione dei risultati ottenuti. Il materiale teorico presuppone le usuali conoscenze di matematica avanzata che caratterizzano i laureati triennali in fisica ed è particolarmente curato per le esigenze del fisico teorico, allo scopo di formare studenti che siano in grado di risolvere problemi computazionali nei più diversi ambiti di applicazione. Le esercitazioni in laboratorio costituiscono parte integrante del corso: gli studenti svilupperanno porzioni di codice a partire da esempi discussi a lezione. Sono inoltre previste prove in itinere che consisteranno in problemi individuali assegnati agli studenti con relazioni finali che potranno essere valutate in sede di esame.

#### **Risultati di apprendimento.**

Conoscenza e capacità di comprensione:

- Acquisizione degli elementi fondamentali della programmazione in python per la fisica teorica.
- Acquisizione degli elementi teorici della fisica computazionale. Conoscenza dei principali algoritmi per la risoluzione dei problemi per via numerica.
- Acquisizione degli elementi fondamentali del calcolo numerico nell'ambito della fisica teorica con particolare riferimento alla determinazione dell'errore nelle soluzioni numeriche.

Capacità di applicare conoscenza e comprensione:

- Capacità di risolvere numericamente le equazioni della fisica teorica, con tecniche di programmazione avanzate adattate alle esigenze del fisico teorico.
- Capacità di analizzare criticamente le diverse possibilità per la risoluzione numerica dei problemi posti dalla fisica teorica.
- Capacità di visualizzare, interpretare e presentare i risultati del calcolo numerico.

## Programma del Corso

- 1. La programmazione in python per i fisici.** Istruzioni di base: variabili, assegnazioni, input/output, operazioni aritmetiche, funzioni, pacchetti, moduli. Cicli if/while, istruzioni di controllo. Liste, matrici. Stili di programmazione per la fisica computazionale.
- 2. Grafica e visualizzazione.** Grafici semplici. Grafici di dispersione (scatter plot). Grafici di densità (density plot). Grafici in tre dimensioni. Animazioni.
- 3. Accuratezza e velocità.** Variabili e intervalli, errore numerico, velocità di esecuzione dei programmi.
- 4. Integrali e derivate.** Metodi fondamentali per il calcolo numerico degli integrali: regola del trapezoide, regola di Simpson. Calcolo degli errori negli integrali. Integrazione di Romberg. Metodi di integrazione di ordine superiore. Quadratura gaussiana. Integrali su intervalli illimitati. Integrali multipli. Calcolo delle derivate. Differenze in avanti e all'indietro. Errori. Differenze centrali. Approssimazioni di ordine superiore per il calcolo delle derivate. Derivate seconde. Derivate parziali. Derivate di dati rumorosi. Interpolazione.
- 5. Risoluzione delle equazioni lineari e non lineari.** Equazioni lineari simultanee: eliminazione gaussiana, retrosostituzione (backsubstitution), pivoting, decomposizione LU, calcolo del reciproco di una matrice, matrici tridiagonali e a banda. Autovalori e autovettori. Equazioni non lineari: metodo del rilassamento, ricerca binaria, metodo di Newton, metodo della secante, metodo di Newton per due o più variabili. Massimi e minimi delle funzioni: ricerca del rapporto aureo, metodo di Gauss-Newton e del gradiente.
- 6. Trasformate di Fourier.** Serie di Fourier. Trasformata di Fourier discreta: posizione dei punti, trasformate in due dimensioni, interpretazione fisica. Applicazioni. Metodo FFT (Fast Fourier Transform).
- 7. Equazioni differenziali ordinarie.** Equazioni differenziali del primo ordine in una variabile: metodo di Eulero, metodo di Runge-Kutta, metodo di Runge-Kutta al quarto ordine. Equazioni differenziali in più di una variabile. Equazioni differenziali al secondo ordine. Passi di integrazione adattativi. Problemi al contorno: metodo dello shooting, del rilassamento, degli autovalori.
- 8. Equazioni differenziali alle derivate parziali.** I problemi al contorno e metodo del rilassamento. Metodi più rapidi per i problemi al contorno: ultrarilassamento, metodo Gauss-Seidel. Problemi di valore iniziale: metodo FTCS, stabilità numerica, metodo implicito e metodo Crank-Nicolson. Metodi spettrali.
- 9. Processi stocastici e metodo Monte Carlo.** Numeri casuali: generatori di numeri casuali, scelta del numero seme, probabilità, numeri casuali non uniformi, numeri casuali con distribuzione Gaussiana. Integrazione Monte Carlo: metodo del valor medio, integrali in più dimensioni, campionamento di importanza. Simulazioni Monte Carlo: campionamento di importanza in fisica statistica, il metodo delle catene di Markov. Simulated annealing.

Modalità di svolgimento delle lezioni: ogni lezione teorica avrà la durata di due ore. Le lezioni di carattere applicativo (esercitazioni) saranno svolte nel laboratorio di fisica computazionale.

Libro di testo:

- Mark Newman, Computational Physics, University of Michigan, 2012.

Altri testi consultati e consigliati:

- Luciano M. Barone, Enzo Marinari, Giovanni Organtini, Federico Ricci-Tersenghi, Programmazione scientifica, Pearson Education, 2006.
- Rubin H. Landau, Manuel J. Paez, Cristian C. Bordeianu, Computational Physics, Wiley 2007.
- Alejandro L. Garcia, Numerical Methods for Physics, 2nd Edition, Prentice Hall, 2000.