

# Introduzione

Per persone attive a tempo pieno nel campo della ricerca scientifica la scelta di scrivere un libro di testo è sempre una scelta impegnativa e difficile: occorre dedicare molto tempo all'impresa, le decisioni sul taglio da dare al testo sono difficili e la garanzia che il testo risulti, a conti fatti, realmente innovativo, non c'è.

Per nostra fortuna abbiamo cominciato, pochi anni fa, a tenere, insieme ad altri colleghi, i corsi in cui si insegna l'uso del calcolatore agli studenti del Corso di Laurea in Fisica presso la Facoltà di Scienze Matematiche Fisiche e Naturali, dell'Università *La Sapienza* di Roma. Si tratta di corsi nuovi, nati con la recente riforma italiana ed europea, cosiddetta del *tre piú due*, che cercano di fornire agli studenti le competenze necessarie per operare in modo professionale nelle discipline scientifiche e tecnologiche.

L'insegnamento in questi corsi è basato sulla nostra esperienza di programmazione su calcolatore in diversi campi della ricerca scientifica: durante i corsi ci siamo resi conto che molte nozioni fondamentali nell'uso quotidiano del computer in campo scientifico non sono mai discusse nei tipici manuali usati come libri di testo. La ricerca di un libro di testo adeguato a corsi di questo tipo ha dato un esito negativo. I testi in circolazione, in effetti, si possono considerare come appartenenti a due categorie:

1. i manuali dei linguaggi di programmazione;
2. i testi di calcolo numerico.

I primi tendono spesso a illustrare l'uso di un linguaggio senza un fine preciso. Sono ricchi di esempi ed esercizi, ma per lo piú rivolti ad applicazioni di carattere gestionale o ludico; i pochi esempi a carattere matematico sono problemi troppo elementari per essere interessanti per uno studente universitario e quindi non sono adatti a studenti di facoltà scientifiche (ingegneria, matematica, fisica, chimica, biologia). I testi di calcolo numerico, viceversa, sono quasi sempre pensati per studenti degli anni finali degli studi universitari, e presuppongono di solito la conoscenza avanzata di un linguaggio di programmazione e di nozioni di matematica e di fisica evolute. Un docente che cerchi di adottare due testi, uno per ognuna di queste due categorie, non risolve il problema, perché gli argomenti importanti appaiono scorrelati, senza una traccia né un percorso comune.

Ci siamo infine convinti che non esistono, nel panorama attuale dei libri di testo, manuali d'introduzione alla programmazione pensati per la formazione universitaria di base a carattere scientifico.

Svincolandoci dai libri di testo in circolazione, abbiamo quindi elaborato uno schema di lavoro che ci sembra innovativo e adeguato al contesto dei nuovi corsi universitari. In questa situazione la possibilità di scrivere un nuovo libro di testo è apparsa un'idea interessante; abbiamo trovato anche stimolante il fatto che il progetto che via via abbiamo elaborato ha, a nostro avviso, un carattere internazionale. I curricula degli Atenei stranieri prevedono spesso corsi analoghi, senza tuttavia disporre di testi adeguati.

Nel percorso che abbiamo pensato e costruito, l'apprendimento del linguaggio di programmazione non è un fine, ma il mezzo attraverso il quale lo studente impara a costruire modelli per la soluzione di problemi di carattere scientifico. Nel nostro modo di insegnare, il trascorrere del tempo e l'evolversi della formazione dello studente contano, e gli esempi scelti sono sempre accessibili allo studente, perché sono legati al suo sviluppo formativo: i primi esempi sono semplici riferimenti a problemi matematici di calcolo infinitesimale, poi, nel secondo anno, si affrontano questioni fondamentali di fisica e matematica, come lo studio di equazioni differenziali e la loro applicazione a problemi di dinamica, sino ad arrivare, al terzo anno, a problemi di ottimizzazione anche di una certa complessità.

Queste considerazioni ci hanno convinti, e ci siamo decisi a realizzare il progetto di questo libro, diviso in tre parti, che riflettono le tre fasi dell'insegnamento della programmazione e che corrispondono a circa tre corsi universitari. Il percorso è strettamente legato allo sviluppo di un curriculum ingegneristico o scientifico visto nella sua interezza. I tre corsi che proponiamo sono legati in modo forte, e portano lo studente da un possibile "analfabetismo" informatico a una competenza avanzata e consapevole. Al tempo stesso le tre parti hanno un certo grado di indipendenza che permette a un docente di usarle anche parzialmente per corsi particolari.

Il titolo del libro è breve, ma vuole comunicare varie idee forti. Il sostantivo *Programmazione* esprime la nostra pretesa di insegnare le tecniche attraverso cui costruire modelli e algoritmi generali per la soluzione di un problema, insieme a quelle per realizzare un buon codice eseguibile. L'aggettivo *scientifica* ha una duplice accezione: in parte si riferisce al fatto che i problemi trattati sono di carattere scientifico, e in parte al fatto che il modo in cui introduciamo i metodi della programmazione ricalca quello usato nei corsi di Scienze per illustrare i concetti nuovi. Metodi, istruzioni e modelli non sono forniti come *ricette*, ma di ognuno si esaminano le ragioni di fondo, i meccanismi che li rendono utili, le possibili alternative, i pro e i contro.

## **Cosa non è questo libro**

Per quanto detto questo testo non è un manuale del linguaggio C: in libreria o in Internet si trovano molti di questi manuali. In *Programmazione Scientifica* il linguaggio C è introdotto come mezzo per imparare a programmare: un processo del tutto analogo a quello necessario per imparare a parlare. Così come per imparare a parlare occorre una lingua, per imparare a programmare occorre un linguaggio. Tuttavia lo scopo di questo testo non è quello di approfondire il linguaggio C in tutti i suoi aspetti, per i quali rimandiamo alla

bibliografia. Nonostante ciò, nel testo sono presenti informazioni sufficienti a introdurre il linguaggio senza l'ausilio di un manuale.

Infine, questo testo non vuole nemmeno essere un manuale di programmazione a carattere generale: non sono discusse nei dettagli applicazioni a carattere gestionale, grafico o multimediale. Le applicazioni scientifiche consistono essenzialmente nel calcolo e nell'organizzazione dei dati. È vero che i risultati si presentano spesso in modo grafico, ma il più delle volte ciò si ottiene attraverso l'uso di programmi specializzati. Questi programmi sono facilmente reperibili, in commercio o gratuitamente in Internet, e non discutiamo come vengano messi a punto.

Questo libro non è nemmeno un manuale di calcolo numerico. Sono presenti, è vero, numerosi algoritmi di calcolo, anche molto evoluti, ma sempre presentati attraverso un linguaggio semplice e intuitivo. Non ci interessa discutere tutti gli aspetti del calcolo numerico dal punto di vista matematico: per la dimostrazione dei teoremi, la trattazione dettagliata delle implicazioni di certe scelte e gli aspetti più teorici, si rimanda alla bibliografia. Tuttavia lo studente diligente arriverà a possedere tutti gli strumenti necessari per scrivere programmi di calcolo completi, accurati ed efficienti.

## **Come usare questo libro**

Il libro è diviso in tre parti. Ogni parte rispecchia i contenuti dei corsi che teniamo per i nostri studenti nei rispettivi anni di corso. Ogni parte è dunque pensata per contenere il materiale necessario per un corso universitario di primo livello di circa cinque crediti.

Per la prima parte non è necessario alcun prerequisito. Lo studente è accompagnato in un percorso di conoscenza degli strumenti (il computer e i linguaggi di programmazione), indispensabile per la comprensione delle tecniche di progettazione delle soluzioni. L'obiettivo della prima parte è quello di consentire allo studente di familiarizzare con le tecniche, imparando a usarle attraverso esempi semplici e interessanti. Qui si imparano i costrutti elementari e fondamentali del linguaggio C e li si applica alle tecniche numeriche di base (calcolo di derivate, integrali, ordinamento).

Nella seconda parte si comincia a privilegiare l'aspetto della progettazione. In ogni capitolo si affronta un problema diverso e lo si risolve. Questa è l'occasione per illustrare, attraverso gli esempi, le tecniche generali della soluzione dei problemi, per approfondire gli aspetti evoluti del linguaggio e per imparare a considerare questioni spesso trascurate, come la robustezza e l'efficienza dei programmi. I problemi proposti sono originali e stimolanti.

Nella terza parte si assume che lo studente domini completamente gli strumenti (e in particolare il linguaggio di programmazione) e s'illustrano i problemi complessi di calcolo più comuni nella vita di un ingegnere o di uno scienziato, insieme a tecniche di ottimizzazione dei codici. Anche in questa parte gli argomenti sono stati scelti con l'intento di offrire un panorama generale delle tecniche adottate, ma esaminandone spesso di nuove e poco conosciute. A nostro avviso, il materiale contenuto nella terza parte è

piú corposo di quello che si può ragionevolmente trattare in un corso normale: il docente potrà scegliere gli argomenti da affrontare.

Se questo testo susciterà negli studenti almeno la metà dell'entusiasmo che è cresciuto in noi nello scriverlo, la sua utilità sarà, non c'è dubbio, molto grande.

## Ringraziamenti

Al termine di questo lavoro vogliamo prima di tutto ringraziare i nostri colleghi, i nostri collaboratori, i nostri studenti di dottorato e tutti quelli che, in un modo o nell'altro, contano su di noi.

Un ringraziamento particolare va al Direttore e al personale del Dipartimento di Fisica della nostra Università, e ai Direttori e al personale della Sezione dell'Istituto Nazionale di Fisica Nucleare e della Unità dell'Istituto Nazionale per la Fisica della Materia (ora assurdamamente disciolto) di Roma 1: senza il grande aiuto di tutti il nostro lavoro sarebbe stato impossibile.

Grazie ai nostri colleghi Zhen'an Liu di Pechino e Wu Hao di Shanghai, per averci aiutato a scrivere la citazione del Capitolo 3. Rosa Avolio ha creato parte di un tautogramma proprio per noi: abbiamo trovato delizioso questo fatto.

Abbiamo avuto un importante aiuto da tutti i colleghi che hanno insegnato in questi anni e insegnano nei molti corsi di programmazione del Corso di Laurea in Fisica della *Sapienza*. Gli studenti che hanno seguito i nostri corsi ci hanno stimolato e incalzato con domande e discussioni spesso molto importanti per lo sviluppo del testo: sono troppi per citarli tutti, ma li ringraziamo molto caldamente.

Alcuni dottorandi e giovani ricercatori borsisti hanno, in vari momenti, contribuito all'insegnamento nei nostri corsi, e hanno corretto molti errori, piú o meno gravi: a Tommaso Castellani, Tommaso Chiarusi, Jovanka Lukic e Valery Van Kerrebroeck va un riconoscimento pieno di sincera gratitudine.

L'implacabile rigore filologico, estetico, sintattico e grammaticale di Carlo Piano ci ha molto aiutato: la scorrevolezza e la fluidità di questo testo devono tanto al suo competente lavoro.

A Lisa Ferranti, splendido *editor* di Pearson Education, dobbiamo tutto: lei ha creduto, dall'inizio, nel nostro progetto, lo ha trovato, cullato e fatto crescere. Durante il nostro lavoro ci ha seguito, stimolato, aiutato senza risparmiarsi nemmeno per un momento: le sue competenze tecniche sono state determinanti per arrivare al testo che ora diamo alle stampe e hanno segnato in modo indimenticabile questo progetto. Lisa è stata per noi una guida e un'amica preziosa e, davvero, senza di lei niente avrebbe potuto essere. Grazie Lisa.

Roma, 4 novembre 2005.

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